

BAB II

LANDASAN TEORI

2.1 SISTEM INFORMASI

2.1.1 PENGERTIAN SISTEM

Sistem adalah kumpulan dari elemen – elemen yang saling berkaitan dan tersusun untuk mencapai tujuan tertentu (Jogiyanto, Hal 2,1995). Dalam bekerja sistem memperoleh masukan (Data) yang menghasilkan keluaran (Informasi).

2.1.2 PENGERTIAN INFORMASI

Informasi adalah sesuatu yang menambah pengetahuan bagi yang menerima dan dapat digunakan untuk membantu mengurangi keragu – raguan penerimaannya dalam pengambilan keputusan untuk menentukan pilihan tindakan dan informasi itu sendiri bersifat subyektif (Jogiyanto, Hal 8, 1995).

2.1.3 PENGERTIAN SISTEM INFORMASI

Sistem informasi adalah aplikasi komputer untuk mendukung operasi dari suatu organisasi: operasi, instalasi, dan perawatan komputer, perangkat lunak, dan data (http://id.wikipedia.org/wiki/Sistem_informasi).

. **Sistem Informasi** adalah sekumpulan hardware, software, brainware, prosedur dan atau aturan yang diorganisasikan secara integral untuk mengolah data menjadi informasi yang bermanfaat guna memecahkan masalah dan pengambilan keputusan (http://id.wikipedia.org/wiki/Sistem_informasi).

Sistem Informasi adalah satu Kesatuan data olahan yang terintegrasi dan saling melengkapi yang menghasilkan output baik dalam bentuk gambar, suara maupun tulisan (http://id.wikipedia.org/wiki/Sistem_informasi).

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 (Jogiyanto,1995,hal:2).

Sistem Informasi Manajemen adalah kunci dari bidang yang menekankan finansial dan personal manajemen. Sistem Informasi Penjualan adalah suatu sistem informasi yang mengorganisasikan serangkaian prosedur dan metode yang dirancang untuk menghasilkan, menganalisa, menyebarkan dan memperoleh informasi guna mendukung pengambilan keputusan mengenai penjualan.

2.1.4 PENGERTIAN PENJUALAN LANGSUNG

Direct Selling (Penjualan Langsung) adalah : Metode penjualan barang dan/atau jasa tertentu kepada konsumen dengan cara tatap muka di luar lokasi eceran tetap oleh jaringan pemasaran yang dikembangkan oleh Mitra Usaha dan bekerja berdasarkan komisi penjualan, bonus penjualan dan iuran keanggotaan yang wajar. yang termasuk dalam penjualan langsung (Direct Selling), yaitu :

- a. **Single Level Marketing (Pemasaran Satu Tingkat)** maksudnya adalah : Metode pemasaran barang dan/atau jasa dari sistem Penjualan Langsung melalui program pemasaran berbentuk satu tingkat, dimana Mitra Usaha mendapatkan komisi penjualan dan bonus penjualan dari hasil penjualan barang dan/atau jasa yang dilakukannya sendiri.

- b. Multi Level Marketing (Pemasaran Multi Tingkat)** maksudnya adalah : Metode pemasaran barang dan/atau jasa dari sistem Penjualan Langsung melalui program pemasaran berbentuk lebih dari satu tingkat, dimana mitra usaha mendapatkan komisi penjualan dan bonus penjualan dari hasil penjualan barang dan/atau jasa yang dilakukannya sendiri dan anggota jaringan di dalam kelompoknya.

2.1.6 PERANCANGAN DATABASE

Database merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di simpanan luar komputer dan digunakan perangkat lunak tertentu untuk memanipulasinya (Jogiyanto, Hal 217, 1995).

2.2 FLOW OF DIAGRAM (FOD)

Merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusannya, bagan alir dokumen tersebut menggunakan simbol – simbol berikut :

1. Document (Dokumen)

Simbol ini bisa diartikan sebagai sebuah dokumen (berbentuk kertas, dikenal dengan sebutan ‘hard copy’) data masukan, atau sebuah laporan yang dihasilkan (dicetak) oleh suatu proses tertentu. Jika simbol dokumen masukan maka akan ada simbol arah yang keluar darinya. Jika simbol laporan hasil keluaran maka akan ada simbol arah yang menuju pada simbol tersebut.



Gambar 2.1 Simbol Dokumen dalam FOD

2. Clerical Operation (Operasi Manual)

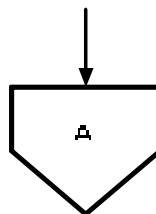
Simbol ini digunakan untuk menjelaskan adanya proses manual yang terjadi dalam rangkaian proses tersebut. Misalnya, tentang adanya pemeriksaan secara fisik oleh petugas atau suatu hasil laporan tertentu sebelum dilanjutkan ke langkah proses berikutnya.



Gambar 2.2 Simbol Clerical Operation dalam FOD

3. Off-Page Connector (Penghubung Halaman)

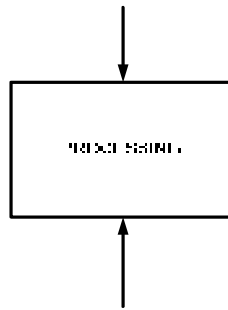
Simbol yang menandakan adanya suatu hubungan rangkaian langkah proses, yang harus berlanjut di halaman lain. Sebuah diagram sistem yang besar sering kali tak bisa dimuatkan secara sekaligus dalam sebuah lembar kertas. Jika lanjutan suatu langkah terpaksa harus dilanjutkan di lembar lain, maka perlu menggunakan simbol ini. Misalnya, di halaman 1 diperlukan adanya simbol tersebut, maka pada penulisan simbol ini harus diberikan suatu tanda tertentu pada bidang yang ada di tengahnya. Misalnya, huruf 'A' nanti di halaman lain sebagai sambungannya, juga ada simbol serupa, dengan kode 'A' yang sama pula. Kode yang sama ini untuk membantu pembaca diagram ini untuk mengetahui kelanjutan langkah proses yang terpaksa harus keluar dari halaman 1 tadi.



Gambar 2.3 Off-Page Connector dalam FOD

4. Processing (Proses)

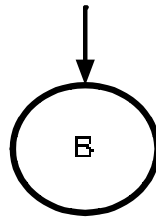
Simbol ini menyatakan adanya sebuah pengolahan data pada langkah tersebut. Simbol ini akan menerima masukan dari simbol lainnya, dan akan mengeluarkan hasil proses, untuk dilanjutkan pada langkah berikutnya. Jadi akan terdapat dua garis penunjuk arah masuk (ke dalam simbol ini) dan keluar. Di dalam kotak simbol akan dituliskan proses-proses apa saja yang akan dikerjakan pada tahap ini.



Gambar 2.4 Simbol Processing dalam FOD

5. Connector (Penghubung)

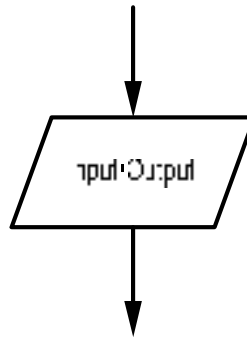
Simbol Connector dalam penggunaannya hampir sama dengan simbol off-page connector yang gunanya adalah untuk menyambung suatu rangkaian langkah proses. Tetapi simbol ini dipakai hanya untuk sambungan yang ada di halaman yang sama. Cara penggunaannya sama, yaitu dengan membubuhkan suatu tanda tertentu yang sama, baik pada saat hubungan tadi terputus atau disambungannya. Simbol ini dipakai agar penggambaran flowchart di lembar tersebut tidak terlalu sukar untuk dilihat, karena adanya garis-garis yang saling menyilang di antara garis yang lain.



Gambar 2.5 Simbol Connector dalam FOD

6. Input/Output

Simbol ini digunakan untuk adanya proses pembacaan data file input, atau melakukan penulisan pada file keluaran (Output).



Gambar 2.6 Simbol Input/Output dalam FOD

7. Online Keyboard (Input yang dilakukan langsung dari Keyboard)

Simbol ini digunakan untuk menyatakan adanya intervensi proses yang harus diberikan oleh operator, pada saat berlangsungnya suatu rangkaian proses dikomputer.



Gambar 2.7 Simbol Online Keyboard dalam FOD

2.3 DIAGRAM ALUR DATA (DATA FLOW DIAGRAM)

Adalah suatu model alat bantu yang memungkinkan sistem analis menggambarkan suatu sistem sebagai suatu jaringan kerja proses dan fungsi yang dihubungkan 1 sama lain oleh penghubung yang disebut dengan data flow (Alur Data).

Komponen – komponen yang digunakan untuk menggambarkan Data flow Diagram (DFD) terdiri dari :

1. External Entity (Terminator/Kesatuan Luar)

External Entity sebagai finish (tujuan) dan source (sumber), Entity harus merupakan suatu bagian atau sekumpulan orang yang berada diluar lingkup pengolahan data yang digambarkan dalam bentuk bujur sangkar.



Gambar 2.8 Simbol Terminator dalam DFD

2. Data Flow/Alur Data

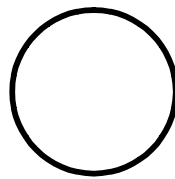
Merupakan perpindahan data atau informasi dari suatu bagian ke bagian lain dalam suatu sistem, Data flow atau alur data digambarkan sebagai anak panah untuk menunjukkan arah, baik dari kiri ke kanan maupun dari kanan ke kiri.



Gambar 2.9 Simbol Alur Data dalam DFD

3. Proses

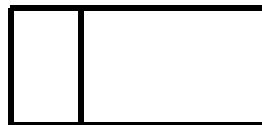
Simbol ini sering dikenal dengan fungsi atau informasi. Komponen proses menggambarkan bagian dari sistem yang mentransformasikan input ke output. Proses digambarkan dengan suatu lingkaran, lingkaran tersebut diberi nomor label sesuai dengan pekerjaan yang dilakukan.



Gambar 2.10 Simbol Proses dalam DFD

4. Data Store (Penyimpanan Data)

Data store digunakan sebagai sarana untuk mengumpulkan data, suatu nama perlu diberikan pada data store karena nama itu menunjukkan nama dari fieldnya. Penyimpanan data store ini digambarkan dengan 2 garis sejajar dan diberi penutup pada salah satu ujungnya, atau cukup dengan 2 garis sejajar tanpa diberi penutup pada salah satu ujungnya.



Gambar 2.11 Simbol Data Store dalam DFD

2.4 ENTITY RELATIONSHIP DIAGRAM (ERD)

Merupakan suatu penyajian data dengan menggunakan entity dan relationship. Dalam ERD ada 3 komponen yaitu :

1. Entity

Yaitu sesuatu yang dapat dibedakan dalam dunia nyata dimana informasi yang berkaitan dengannya dikumpulkan, sedangkan Entity Set adalah

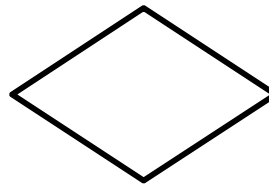
kumpulan dari entity yang sejenis, Symbol yang digunakan adalah Persegi Panjang.



Gambar 2.12 Simbol Entity dalam ERD

2. Relationship

Relationship adalah hubungan persekutuan yang terjadi antara 1 atau lebih entity relationship tidak mempunyai keberadaan fisik yang memenuhi dari hubungan antara entity tersebut. Sedangkan Relationship Set adalah kumpulan dari relationship yang sejenis, relationship digambarkan dengan belah ketupat atau Wajik.



Gambar 2.13 Simbol Relationship dalam ERD

3. Attribut

Attribut adalah karakteristik dari entity atau relationship yang menyediakan penjelasan detail tentang entity atau relationship tersebut. Nilai attribut merupakan suatu data aktual atau informasi yang disimpan pada suatu attribut di dalam suatu entity atau relationship. Attribut digambarkan dengan Elips.



Gambar 2.14 Simbol Attribut dalam ERD

2.5 NORMALISASI

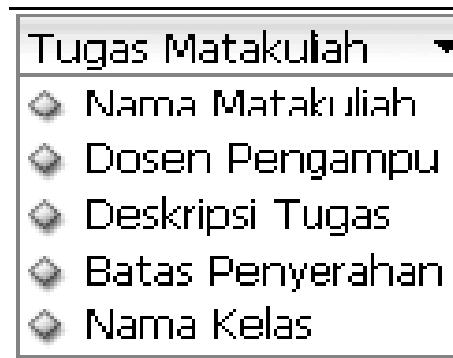
Normalisasi database biasanya jarang dilakukan dalam database skala kecil, dan dianggap tidak diperlukan pada penggunaan personal. Namun seiring dengan berkembangnya informasi yang dikandung dalam sebuah database, proses normalisasi akan sangat membantu dalam menghemat ruang yang digunakan oleh setiap tabel di dalamnya, sekaligus mempercepat proses permintaan data. Berikut ini dipaparkan metodologi logis sederhana untuk menormalkan model data dalam sebuah database, diiringi contoh pembuatan database untuk tugas - tugas matakuliah dalam sebuah fakultas (fiktif) dengan atribut yang disederhanakan. Proses normalisasi model data dapat diringkas sebagai berikut:

1. Temukan entitas-entitas utama dalam model data.
2. Temukan hubungan antara setiap entitas.
3. Tentukan atribut yang dimiliki masing-masing entitas.

Normalisasi model data dilakukan dengan mengikuti langkah-langkah sederhana, mengubahnya agar memenuhi apa yang disebut sebagai bentuk normal pertama, kedua, lalu ketiga secara berturutan. Langkah - Langkah dalam pembuatan Normalisasi adalah sebagai berikut :

2.5.1 Bentuk Normal Pertama (1NF)

Sebuah model data dikatakan memenuhi bentuk normal pertama apabila setiap atribut yang dimilikinya memiliki satu dan hanya satu nilai. Apabila ada atribut yang memiliki nilai lebih dari satu, atribut tersebut adalah kandidat untuk menjadi entitas tersendiri. Entitas utama untuk database tugas matakuliah tentu saja Tugas Matakuliah. Sebagian atribut yang dimiliki entitas ini tertera dalam Gambar 2.15.



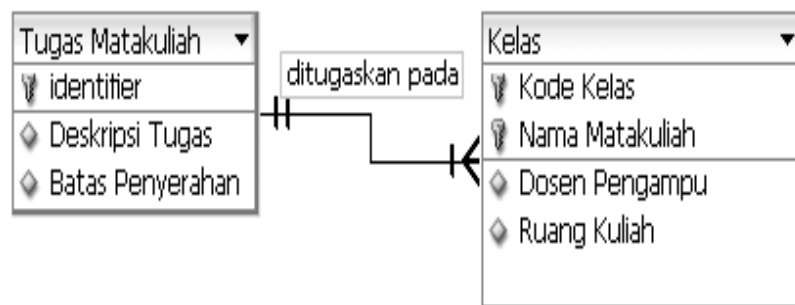
Gambar 2.15. Entitas pertama dalam contoh model data untuk database tugas matakuliah.

Atribut Nama Kelas mencantumkan kelas-kelas di mana tugas tersebut berlaku. Apabila pendaftar untuk sebuah matakuliah melebihi kapasitas ruangan yang dimiliki fakultas, kebijakan yang umum diambil Kepala Program Studi adalah membagi kegiatan perkuliahan untuk matakuliah tersebut menjadi beberapa kelas. Karenanya atribut ini rentan memiliki nilai jamak, dan lebih sesuai menjadi entitas baru atau atribut dari entitas lain. Untuk sementara kita membuat entitas baru, Kelas, dimana sebagian atributnya berasal dari Tugas Matakuliah yang secara logis lebih sesuai menjadi atribut entitas ini. Sementara itu, hampir semua atribut entitas Tugas Matakuliah selain Nama Kelas memiliki nilai tunggal (dengan asumsi setiap matakuliah diampu oleh satu dosen saja).

2.5.1.1 Relasi Antar-Entitas dan Identifier

Masalah yang kita hadapi sekarang adalah menghubungkan Tugas Matakuliah dengan Kelas. Satu tugas dapat diberikan pada beberapa kelas yang berbeda dalam terminologi pemodelan data, ini berarti antara entitas Tugas Matakuliah dan entitas Kelas terdapat **relasi 1:N** (atau 1-N) untuk nilai N lebih dari satu. Cara paling intuitif untuk menghubungkan kedua entitas tersebut adalah menyertakan **identitas** satu entitas sebagai atribut entitas lain.

Identitas sebuah entitas haruslah unik untuk menghindarkan ambiguitas saat akan merujuk pada satu objek khusus dari entitas tersebut. Entitas Tugas Matakuliah akan menggunakan pengidentifikasi arbitrer berupa angka yang berbeda antara satu objek Tugas Matakuliah dengan objek Tugas Matakuliah lain. Entitas Kelas dapat diidentifikasi dengan matakuliah dan kode kelas yang bersangkutan, sehingga kita cukup menambahkan atribut pengidentifikasi (*identifier*) dalam kedua entitas. Entitas ini beserta semua atribut baru dan hubungannya dengan Tugas Matakuliah diperlihatkan dalam Gambar 2.16 dengan menggunakan notasi relasi *crows foot* (dengan simbol “kaki gagak”



Gambar 2.16 Hubungan antara Tugas Matakuliah dan entitas baru, Kelas.

Sejauh ini tidak ada atribut entitas yang memiliki nilai lebih dari satu, sehingga rasanya cukup aman mengatakan bahwa model ini memenuhi bentuk normal pertama.

2.5.2. Bentuk Normal Kedua (2NF)

Sebuah model data dikatakan memenuhi bentuk normal kedua apabila ia memenuhi bentuk normal pertama dan setiap atribut non-*identifier* sebuah entitas bergantung sepenuhnya hanya pada semua *identifier* entitas tersebut. Apabila kita perhatikan kembali model data yang telah kita hasilkan di atas, segera terlihat bahwa atribut dari entitas Kelas tidak sepenuhnya bergantung pada identitas unik Kelas tersebut. Seorang dosen akan tetap ada meskipun kelas matakuliah yang ia ampu sudah tidak ada lagi. Dalam hal ini, dosen adalah entitas tersendiri (yang nantinya

dapat dilekatkan pada entitas Fakultas atau Universitas bilamana kedua entitas tersebut dirasa perlu ada, tergantung pada kebutuhan pemodelan data kita).

2.5.3. Bentuk Normal Ketiga (3NF)

Sebuah model data dikatakan memenuhi bentuk normal ketiga apabila ia memenuhi bentuk normal kedua dan tidak ada satupun atribut *non-identifying* (bukan peng identifikasi unik) yang bergantung pada atribut *non-identifying* lain. Apabila ada, pisahkan salah satu atribut tersebut menjadi entitas baru, dan atribut yang bergantung padanya menjadi atribut entitas baru tersebut. Dalam model data sederhana yang kita gunakan di sini, tidak ada satupun atribut *non-identifying* (seperti Deskripsi Tugas Matakuliah, atau Nama Dosen) yang bergantung pada atribut *nonidentifying* lain. Namun demi adanya contoh, kita misalkan entitas Dosen memiliki atribut informasi Alamat Rumah dan Nomor Telepon Rumah. Keduanya tidak dapat secara unik mengidentifikasi objek tertentu dari entitas Dosen, namun keduanya saling bergantung. Sebagaimana dalam dua langkah normalisasi sebelumnya, jenis kebergantungan seperti ini dapat dihilangkan dengan membuat entitas baru lagi (yang tidak akan diciptakan karena tiga entitas sudah cukup banyak untuk satu artikel). Model terakhir yang kita dapat ini telah memenuhi bentuk normal ketiga (*third normal form*) dan siap dikonversi menjadi tabel. Namun sebelumnya, kita perlu membahas berbagai jenis relasi yang kerap ditemui dalam pemodelan data, termasuk yang kita temui dalam contoh model data kali ini.

2.6 Microsoft Visual Basic 6.0

Microsoft Visual Basic adalah bahasa pemrograman yang digunakan untuk membuat aplikasi windows yang berbasis grafis (*GUI Graphical User Interface*). Visual Basic merupakan *event-driven programming* (pemrograman terkendali kejadian) artinya program menunggu sampai adanya respon dari pemakai berupa

event atau kejadian tertentu (tombol klik, menu dipilih, dll). Ketika event terdeteksi, kode yang berhubungan dengan event (procedure event) akan dijalankan.

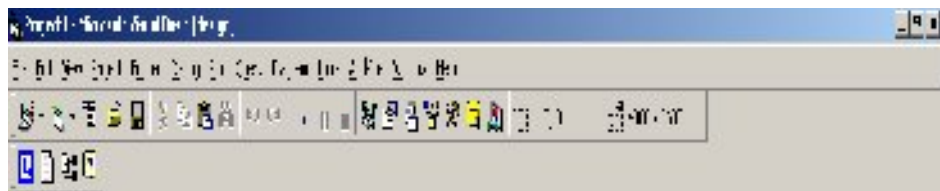
2.6.1 Struktur aplikasi Visual Basic

Aplikasi (Proyek) dalam Microsoft Visual Basic 6.0 terdiri dari :

- **Form** – Windows atau jendela dimana anda akan membuat user interface atau tampilan.
- **Control** – Tampilan berbasis grafis yang dimasukkan pada form untuk membuat interaksi dengan pemakai (textbox, label, scrollbar, tombol command). Form dan Control adalah objek.
- **Properti/Properties** – Nilai/karakteristik yang dimiliki oleh sebuah objek Visual Basic. Contoh : name, caption, size, color, position dan text. Visual Basic menerapkan properti default atau standar. Anda dapat mengubah properti saat mendesign program atau runtime/ketika program dijalankan.
- **Metode/Methods** – serangkaian perintah yang sudah tersedia pada suatu objek yang dapat diminta untuk mengerjakan tugas khusus.
- **Prosedur Kejadian/Event Procedures** – kode yang berhubungan dengan suatu objek. Kode ini akan dieksekusi ketika ada respon dari pemakai berupa event tertentu.
- **Prosedur Umum/General Procedures** – kode yang tak berhubungan dengan suatu objek. Kode ini harus diminta oleh aplikasi.
- **Modul/Module** – Kumpulan dari prosedur umum, deklarasi variabel dan definisi konstanta yang digunakan oleh aplikasi.

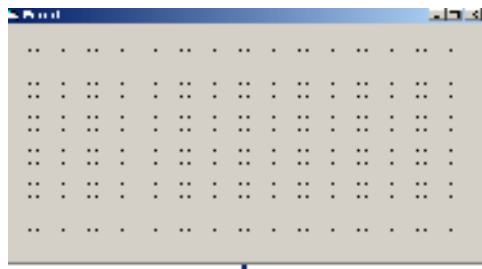
2.6.2 Langkah-langkah untuk mengembangkan aplikasi dalam Microsoft Visual Basic 6.0

- **Main Windows** (Jendela Utama) terdiri dari title bar (baris judul), menu bar, dan tool bar. Baris judul berisi nama proyek, mode operasi Visual Basic sekarang dan form yang aktif. Menu bar merupakan menu drop-down dimana anda dapat mengontrol operasi dari lingkungan Visual Basic. Tool bar berisi kumpulan gambar yang mewakili perintah yang ada di menu. Jendela utama juga menampilkan lokasi dari form yang aktif relatif terhadap sudut kiri atas layar (satuan ukurannya *twips*), juga lebar dan panjang dari form yang aktif.



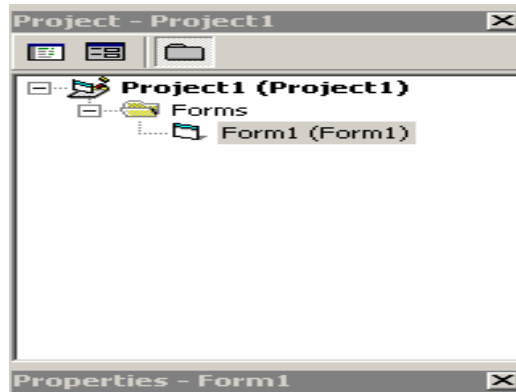
Gambar 2.17 Jendela utama Visual Basic 6.0

- **Form Windows** (Jendela Form) adalah pusat dari pengembangan aplikasi Visual Basic. Di sinilah tempat anda “menggambar” aplikasi.



Gambar 2.18 Jendela Form

- **Project Windows** (Jendela Proyek) menampilkan daftar form dan modul proyek anda. Proyek merupakan kumpulan dari modul form, modul class, modul standar, dan file sumber yang membentuk suatu aplikasi.



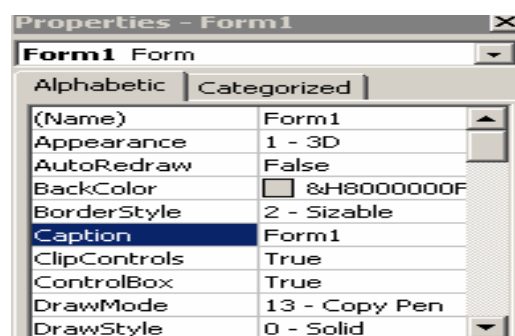
Gambar 2.19 Jendela Proyek

- **Tool Box** adalah kumpulan dari objek yang digunakan untuk membuat user interface serta kontrol bagi program aplikasi. Ada dua cara untuk menempatkan kontrol pada suatu form :
 - Klik ganda kontrol dalam tool box. Selanjutnya anda dapat mengubah besar dan ukurannya serta memindahkannya dengan metode drag and drop.
 - Klik kontrol dalam tool box, kemudian pindahkan pointer mouse pada jendela form. Kursor berubah menjadi *crosshair*. Tempatkan *crosshair* pada sudut kiri atas dimana anda inginkan kontrol diletakkan, tekan tombol mouse kiri dan tahan ketika menyeret (drag) kursor ke arah sudut kanan bawah.



Gambar 2.20 Toolbox

- **Properties Windows** (Jendela Properti) berisi daftar struktur setting properti yang digunakan pada sebuah objek terpilih. Kotak drop – down pada bagian atas jendela berisi daftar semua objek pada form yang aktif. Ada dua tab tampilan :
 - Alphabetic (urut abjad) dan categorized (urut berdasar kelompok). Di bagian bawah kotak terdapat properti dari objek terpilih.



Gambar 2.21 Jendela Properti

- Form Layout Windows (Jendela Layout Form) menampilkan posisi form relatif terhadap layar monitor.






Gambar 2.22 Jendela Layout Form


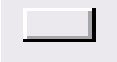








2.6.3 Unit Kontrol Dalam Microsoft Visual Basic 6.0

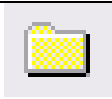
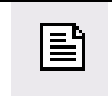



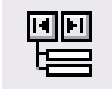

Kontrol adalah alat Bantu yang dipakai untuk membuat sebuah objek pada form Visual Basic. Setelah control ditambahkan ke form, control tersebut berubah menjadi sebuah objek yaitu elemen perantara pemakai yang dapat diprogramkan.

TABEL 2.1

UNIT KONTROL MICROSOFT VISUAL BASIC 6.0

Gambar Kontrol	Nama Kontrol	Awalan	Fungsi
	Picture Box	Pic	Untuk Menampilkan file gambar: .bmp, .jpg, .gif, .wmf dan .ico.
	Label	Lbl	Menampilkan teks tetapi pemakai tidak dapat berinteraksi dengannya atau mengubahnya.
	Text Box	Txt	Menampilkan teks dimana pemakai dapat mengisiatu melihat teks yang ditampilkan sebagai output suatu proses.

	Frame	Fra	Mengidentifikasi sebuah grup pengontrolan.
	Command Button	Cmd	Memberikan sebuah perintah dan tindakan ketika digunakan.
	Check Box	Chk	Memberikan perintah pilihan benar/salah (True/False) atau ya/tidak (Yes/No). Pemakai dapat memilih beberapa check box secara bersamaan.
	Option Button	Opt	Memberikan grup pilihan. Pemakai hanya dapat memilih satu option pada grup pilihan.
	Combo Box	Cbo	Unit control kombinasi antara text box dengan list box. Pemakai bisa mengetikkan pilihan atau memilih lewat daftar drop-down.
	List Box	Lst	Menampilkan daftar item yang dapat dipilih salah satu oleh pemakai.
	Horinzontal Scroll Bar	Hsb	Menampilkan balok gulung horizontal.
	Vertical Scroll Bar	Vsb	Menampilkan balok gulung vertikal.
	Timer	Tmr	Untuk mengeksekusi waktu kejadian pada rutin program termasuk selang waktu (interval).
	Drive List Box	Drv	Menampilkan daftar drive.

	Directory List Box	dir	Menampilkan daftar direktori pada drive terpilih.
	File List Box	Fil	Menampilkan daftar file pada drive dan direktori terpilih.
	Shape	Shp	Membuat lingkaran, elips dan kotak pada form.
	Line	Lin	Membuat garis lurus pada form.
	Image	Img	Menampilkan gambar: .bmp, .gif, .jpg, .wmf dan .ico. Perbedaan dengan picture box adalah control ini memberikan perintah seperti command button saat pemakai mengklik gambar menggunakan mouse.
	Data	Dat	Menampilkan database pada suatu form.
	OLE (Object Linking and Embedding)	Ole	Memungkinkan pemakai menempelkan suatu objek dari aplikasi yang mendukung OLE ke aplikasi Visual Basic.

2.7 Microsoft Access 2000

Microsoft Access 2000 merupakan salah satu program aplikasi basis data paling lengkap dan canggih saat ini. Dengan menggunakan Access 2000 kita dapat merancang, membuat dan mengelola database dengan mudah. Access 2000 merupakan pengembangan dari Access versi sebelumnya, dengan harapan program

aplikasi database ini lebih mudah dipakai, mudah diintegrasikan dengan program aplikasi Microsoft Office 2000 lainnya dan dapat memanfaatkan semua fasilitas yang terdapat pada internet ataupun intranet.

2.7.1 Memulai Access 2000

Banyak cara yang dapat digunakan untuk memulai atau menjalankan Access 2000.

2.7.1.1 Memulai Access 2000 dengan menggunakan tombol start di taskbar

Untuk memulai atau menjalankan Access 2000 dengan menggunakan tombol Start yang ada di taskbar, langkah – langkahnya adalah sebagai berikut :

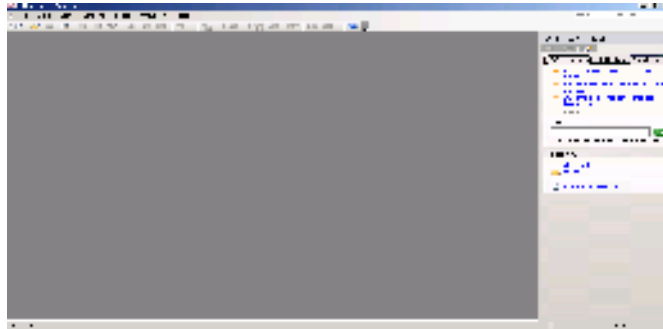
1. Nyalakan komputer Anda, tunggu sampai komputer menampilkan area kerja (Dekstop) Windows 95/98/NT.
2. Klik Tombol **Start** yang ada di taskbar.
3. Pilih menu **Programs**.



Gambar 2.23 Cara memulai Access 2000 melalui tombol Start

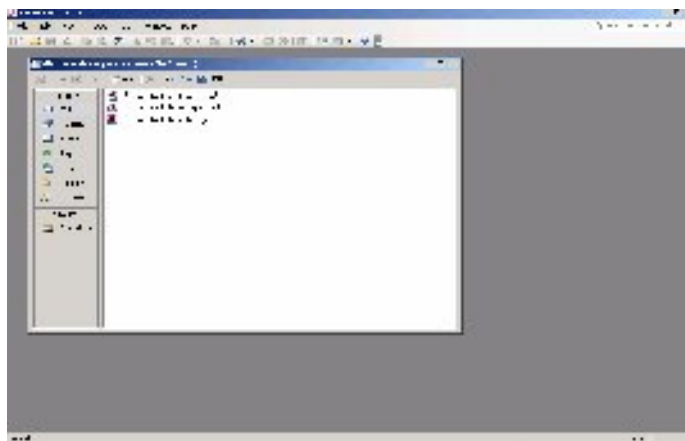
4. Kemudian pilih dan klik program **Office**.
5. Kemudian pilih dan klik **Database**, lalu **Microsoft Office Access 2000**, lalu pilih **Microsoft Access 2000**. Lihat Gambar 1.1. Kemudian

tunggu sampai jendela program aplikasi Access 2000 ditampilkan dan kotak dialog startup Access ditampilkan. Lihat Gambar 1.2.



Gambar 2.24 kotak dialog Microsoft Access 2000

6. Pada kotak dialog tersebut, pilih **Create a New file**. Kemudian pilih **Blank Database**.
7. Setelah di pilih **Blank Database** simpanlah file tersebut kemudian klik tombol **Create**
8. setelah tombol Create diklik kemudian ikuti langkah selanjutnya sesuai dengan pemilihan yang anda lakukan pada kotak dialog dibawah ini.

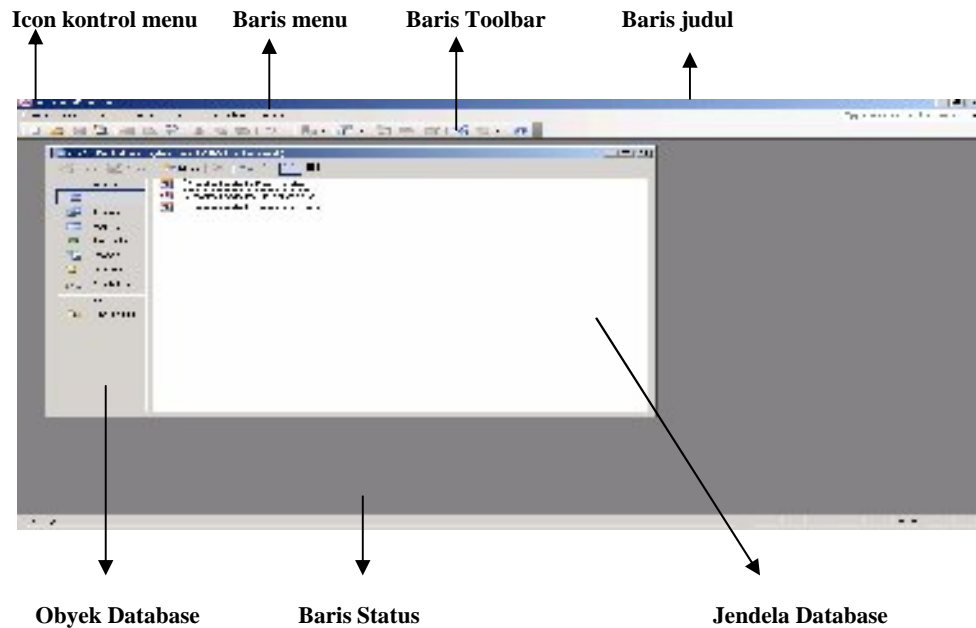


Gambar 2.25 Kotak dialog Access 2000 file format

2.7.2 Mengenal Elemen Dasar Access 2000

Elemen dasar Access 2000 yang perlu anda ketahui di antaranya adalah :

- **Icon kontrol Menu (Contol Menu Icon)**, digunakan untuk mengontrol jendela yang sedang aktif.
- **Baris Judul (Title Bar)**, berisi nama file dan nama program aplikasi yang sedang aktif. Baris judul ini dapat juga digunakan untuk memindahkan jendela ke posisi lain.
- **Baris Menu (Menu Bar)**, berisi barisan perintah berupa menu, seperti **File, Edit, View, Insert, Windows, dan Help.**
- **Baris Toolbar (Toolbars)**, berisi tombol – tombol yang digunakan untuk menjalankan suatu perintah dengan cepat dan mudah, terutama untuk perintah – perintah yang sering digunakan.
- **Baris Penggulung (Scroll Bar)**, untuk menggeser layar ke kiri atau ke kanan gunakan baris penggulung mendatar (Horizontal Scrool Bar) sedangkan untuk menggeser layar ke atas atau kebawah gunakan baris penggulung tegak (Vertical Scroll Bar).
- **Jendela Database**, menampilkan file database yang sedang aktif, lengkap dengan seluruh obyek databasenya.



Gambar 2.26 Menu Utama Microsoft Access

2.7.3 Mengenal database Access 2000

Database atau biasa disebut basis data adalah kumpulan dari data yang berhubungan dengan suatu obyek, topik atau tujuan khusus tertentu. Sebagai contoh buku telepon, kamus bahasa, katalog buku dipergustakaan, data koleksi musik dan video, data pelanggan dan lain-lain.

Microsoft Access 2000 dapat mengelola seluruh data yang dimiliki kedalam suatu file database. Database pada Access 2000 dapat terdiri atas satu atau beberapa tabel, query, form, report, page, makro, dan modul yang semuanya berhubungan atau saling terkait.

- **Tables**, berupa tabel kumpulan data yang merupakan komponen utama dari sebuah database.
- **Queries**, digunakan untuk mencari dan menampilkan data yang memenuhi syarat tertentu dari satu tabel atau lebih. Query dapat juga digunakan untuk mengUpdate atau menghapus beberapa record data pada satu saat yang sama. Selain itu query dapat digunakan untuk menjalankan perhitungan terhadap sekelompok data. Sebuah query dapat memiliki sampai 255 field yang berbeda.
- **Forms**, dipergunakan untuk menampilkan data, mengisi data dan mengubah data yang ada didalam tabel.
- **Reports**, dipergunakan untuk menampilkan laporan hasil analisa data. Kita juga dapat mencetak sebuah report (Laporan) yang telah dikelompokkan, dihitung subtotal dan total datanya berdasarkan kriteria tertentu. Dalam Microsoft Access 2000 kita juga dapat membuat report (Laporan) yang berisi grafik atau tabel data.
- **Pages**, Diperguanakn untuk membuat halaman Web (Page) berupa data Access page yang dapat ditempatkan diServer sistem jaringan internet atau intranet.
- **Macros**, untuk mengotomatisasi perintah – perintah yang sering digunakan dalam mengolah data.
- **Modules**, Digunakan untuk perancangan berbagai modul aplikasi pengolahan database tingkat lanjut sesuai dengan kebutuhan. Module ini berisi kode Visual Basic for Applications yang kita tulis untuk menangani even (peristiwa) dalam Access 2000.

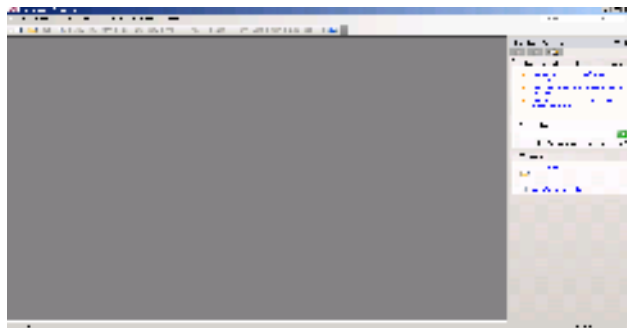
2.7.4 Membuat Database Baru

Database Access 2000 disimpan kedalam sebuah file yang berekstensi mdb. Sebuah file database terdiri dari bagian-bagian yaitu : Tables, Queries, Forms, Reports, Pages, Macros, dan Modules. Database dalam Access 2000 dapat memiliki maksimum 32768 obyek (kombinasi tabel, query, form, report, pages, dan sebagainya).

2.7.4.1 Membuat database baru yang masih kosong

Untuk membuat database baru yang masih kosong, ikuti langkah berikut ini :

Jalankan atau aktifkan program aplikasi Access 2000. Kotak dialog Microsoft Access Startup akan ditampilkan.



Gambar 2.27 Kotak dialog Microsoft Access

2. Pada kotak dialog tersebut, beri tanda atau klik tombol pilihan **Blank Access database**. Kemudian klik **OK**. Kotak dialog file New Database akan ditampilkan.

3. Klik dua kali Icon Kontrol Menu yang berada di pojok kiri atau dari jendela database yang sedang aktif.

2.7.6 Membuka File Database

Untuk membuka kembali sebuah file database tertentu, ikuti langkah-langkah berikut ini :

1. Jalankan atau aktifkan program aplikasi Access 2000. Kotak dialog Microsoft Access Startup akan ditampilkan.
2. Pada kotak dialog tersebut, beri tanda atau klik tombol pilihan **Open an exiting file**. Kemudian pada kotak daftar pilihan yang ada dibawahnya, pilih dan klik nama file database yang diinginkan.
3. Klik **OK**. Tunggu hingga jendela kerja Access 2000 dan database yang anda inginkan terbuka.

Apabila program aplikasi Access 2000 sudah terbuka, kita dapat membuka sebuah database menggunakan langkah berikut ini :

- Pilih dan klik menu **File, Open (CTRL+O)** atau klik tombol toolbar Open.

Kotak dialog Open akan ditampilkan.




Gambar 2.30 Kotak Dialog Open





- Pada tombol daftar pilihan **Look in**, Pilih dan klik Drive dan folder tempat dimana file tersimpan
- Pada kotak isian **File name**, Ketikkan nama file database yang kita inginkan atau klik dua kali pada nama file yang anda inginkan.
- Klik tombol perintah **Open**.




Menggunakan tombol toolbar :

Tabel 2.2
BEBERAPA TOMBOL YANG ADA DI TOOLBAR
MICROSOFT ACCESS

Toolbar	Perintah	Keterangan
	Open	Digunakan untuk membuka kembali file database yang telah anda simpan

Pada saat anda menggunakan kotak dialog Open, ada beberapa tombol toolbar yang dapat anda gunakan, diantaranya :

Toolbar	Keterangan
	Digunakan untuk kembali ke pemilihan drive atau pemilihan folder sebelumnya.
	Digunakan untuk naik satu level ke level folder sebelumnya atau di atasnya.
	Digunakan untuk membuka web browser yang terpasang pada komputer anda.
	Digunakan untuk menghapus file yang sedang dipilih.

	Digunakan untuk membentuk atau membuat folder baru.
	Digunakan untuk mengatur tampilan isi drive atau folder tertentu.
	Digunakan untuk mengatur folder dan file yang sedang anda pilih.