

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Definisi Registrasi dan Administrasi**

Registrasi adalah suatu kegiatan pencatatan yang dilakukan pertama kali pada saat melakukan pendaftaran.

Dalam sebuah instalasi / organisasi, administrasi merupakan hal yang sangat penting. Suatu instalasi / organisasi yang mengalami masalah dalam administrasi akan menjadi tak berguna dan tidak dapat bertahan lama.

❖ **Administrasi dapat dibedakan dalam dua pengertian yaitu :**

1. Administrasi dalam arti sempit yaitu dari kata *Administratie* (Bahasa Belanda) yang meliputi kegiatan : catat-mencatat, surat-menyurat, pembukuan, ketik-mengetik, agenda dan sebagainya yang bersifat teknis ketatausahaan (*Clerical Work*).

Jadi tata usaha dapat diartikan sebagai bagian kecil kegiatan dari pada administrasi yang akan dipelajari.

2. Administrasi dalam arti luas berasal dari kata *Administration* (Bahasa Inggris). Dibawah ini akan dikemukakan pendapat, arti atau definisi dari administrasi dalam arti luas yaitu :

Menurut Leonard D. White, dalam bukunya : *Introduction to study of public Administration*, memberikan definisi sebagai berikut :

“Administration is a proses common to all group suffrot, public or private, civil or military, large scale or small scale...etc”.

(Administrasi suatu proses yang pada umumnya terdapat pada semua usaha kelompok, swasta atau Negara, sipil atau militer, usaha yang besar atau kecil...dan sebagainya).

Sehingga administrasi pembayaran dapat didefinisikan sebagai kegiatan-kegiatan dari pada bidang pembayaran, dalam hal ini

administrasi itu sendiri meliputi kegiatan : mencatat, ketik-mengetik sebagai suatu proses dalam usahanya mencapai tujuan yang hendak dicapai.

## **2.2 Perancangan Sistem**

Sistem adalah sekumpulan unsur atau elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan. Tujuan dari perancangan sistem yaitu untuk memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada program. Dalam perancangan suatu sistem, diperlukan alat-alat bantu berupa :

1. Flow of Document (FOD)
2. Diagram Alir Data (Data Flow Diagram / DFD)
3. Entity Relationship Diagram (ERD)
4. Normalisasi

### **2.2.1 Flow of Document (FOD)**

Flow of Document (FOD) atau bagan alir dokumen adalah bagan alir yang menunjukkan arus laporan dan formulir termasuk tembusannya (Fathansyah, 2000, h : 23-26).

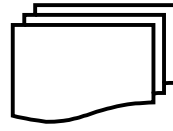
- **Adapun komponen-komponen yang terdapat dalam bagian alir dokumen antara lain adalah :**

#### **1. Dokumen**

Menggunakan simbol berupa secarik kertas yang fungsinya sebagai masukan maupun keluaran untuk suatu proses yang bersifat Non computer maupun dengan menggunakan komputer.



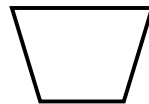
Gambar 2.1. Simbol Dokumen



Gambar 2.2. Simbol Multi Dokumen

## 2. Proses

Merupakan penggambaran suatu proses atau kegiatan yang dikerjakan di dalam sistem.



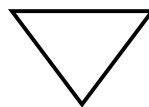
Gambar 2.3. Simbol Proses Non Komputer



Gambar 2.4. Simbol Proses Komputer

## 3. Arsip / File

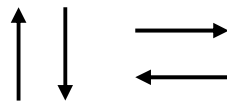
Merupakan media penyimpanan data disk / media lainnya. Digambarkan dengan simbol segitiga.



Gambar 2.5. Simbol Arsip / File

#### 4. Garis Alir

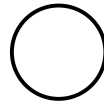
Merupakan garis yang menunjukkan arus dari proses data sebagai penunjuk arah dokumen berjalan, masuk dan keluar dari sistem, simbol yang digunakan untuk garis alir adalah arah panah.



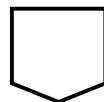
Gambar 2.6. Simbol Garis Alir

#### 5. Konektor

Merupakan penghubung aliran dokumen dalam satu halaman atau antar halaman.



Gambar 2.7. Simbol Penghubung



Gambar 2.8. Simbol Penghubung antar halaman

### 2.2.2 Diagram Alir Data (Data Flow Diagram / DFD)

DFD sebagai model logis yang menjelaskan sistem sebagai jaringan kerja dari proses yang dihubungkan satu dengan lainnya atau tempat penyimpanan data serta dihubungkan pula dengan sumber tujuan.

#### 2.2.2.1 Komponen Data Flow Diagram (DFD)

Data Flow Diagram (DFD) terdiri dari empat komponen yaitu :

1. Terminator
2. Process (Proses)

3. Data Store (Simpan Data)

4. Data Flow (Arus Data)

➤ **Simbol-simbol yang digunakan adalah sebagai berikut :**

**1. Terminator**

Terminator dapat berupa sekelompok orang, organisasi, atau perusahaan yang sama tetapi diluar kendali sistem yang sedang dibuat modelnya.

Simbol yang digunakan yaitu notasi balok :

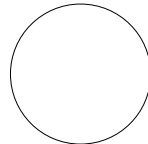


Gambar 2.9. Simbol Terminator

**2. Process (Proses)**

Proses merupakan bagian dari sistem yang mentransformasikan input ke output. Proses ini menerangkan proses atau kegiatan yang dilaksanakan.

Simbol yang digunakan yaitu notasi lingkaran :

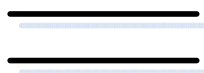


Gambar 2.10. Simbol Process (Proses)

**3. Data Store (Simpan Data)**

Digunakan untuk membuat model sekumpulan paket data. Data store ini sangat berkaitan dengan penyimpanan secara komputerisasi, contoh file disket, file harddisk dan pita magnetic. Data ini juga berkaitan dengan penyimpanan secara manual.

Simbol yang digunakan yaitu segi empat terbuka :



Gambar 2.11. Simbol Data Store

#### 4. Data Flow (Arus Data)

Suatu proses yang menghubungkan antara Arus Data dan Data Store adalah sebagai berikut :

- Arus Data yang berasal dari Data Store, berarti proses yang membutuhkan data yang berasal pada Data Store tersebut.
- Arus Data yang menuju ke Data Store, berarti suatu proses yang akan menghasilkan output atau keluaran yang disimpan pada Data Store tersebut.
- Arus Data yang berasal dan yang menuju ke Data Store berarti suatu proses akan mengupdate data, menghapus dan mengubah data.

Data flow dipresentasikan dengan anak panah yang menuju ke dan keluar dari suatu proses. Alur data ini digunakan untuk menerangkan perpindahan data atau suatu paket data dari suatu bagian ke bagian lainnya.

Simbol yang digunakan yaitu anak panah :



Gambar 2.12. Simbol Arus Data

##### 2.2.2.2 Tingkatan Data Flow Diagram (DFD)

Tingkatan-tingkatan pada Data Flow Diagram (DFD) terdiri dari :

- Diagram Konteks :  
Merupakan diagram paling atas yang mempunyai hanya satu proses dan menggambarkan ruang lingkup sistem.
- Diagram Zero :  
Merupakan diagram yang menggambarkan proses-proses utama sistem dan alur datanya.
- Diagram Detail :  
Merupakan diagram zero yang tidak dapat diuraikan.

### 2.2.3 Entity Relationship Diagram (ERD)

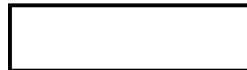
ERD adalah semesta data yang ada di dunia nyata diterjemahkan atau ditransformasikan dengan memanfaatkan sejumlah perangkat konseptual menjadi sebuah diagram. Tujuan utama dari penggambaran ERD adalah agar dapat menggambarkan hubungan antar entitas dengan jelas serta hubungan antara objek data (entity) dan hubungan (relationship) yang ada pada objek tersebut. Agar mudah dimengerti oleh pemakai (user), dan mudah disajikan oleh perancang database.

**Komponen-komponen dari sebuah ERD adalah sebagai berikut :**

#### 1. Entitas

Entitas merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain. Entitas menunjuk pada individu suatu objek, sedang hubungan entitas menunjuk pada rumpun dari individu tersebut. Misalnya seorang pasien, dokter, perawat kumpulan entitas yang sejenis disebut himpunan entitas.

Simbol yang digunakan untuk entitas adalah empat persegi panjang :

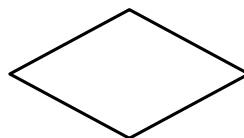


Gambar 2.13. Simbol Entitas

#### 2. Relationship

Relationship adalah asosiasi atau kaitan antara dua entitas yang terjadi antara satu atau lebih entity. Relationship tidak mempunyai keberadaan fisik kecuali yang diwarisi dari hubungan antara entity tersebut. Kumpulan yang sejenis disebut himpunan relationship.

Simbol yang digunakan untuk relationship adalah belah ketupat :



Gambar 2.14. Simbol Relationship

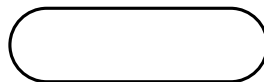
### 3. Atribut

Atribut atau property adalah item data yang menjadi bagian dari suatu entitas atau elemen data.

Atribut value adalah suatu data aktual yang disimpan pada setiap atribut didalam suatu entitas atau relationship.

Atribut terdiri dari dua jenis yaitu identifier (*key*), digunakan untuk menentukan suatu entitas secara unik sedangkan descriptor (*non key atribut*) digunakan untuk menspesifikasikan karakteristik dari suatu entitas yang tidak unik.

Simbol yang digunakan untuk atribut berbentuk elips :



Gambar 2.15. Simbol Atribut

Cardinality adalah relasi yang menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain atau digunakan untuk batasan pada jumlah entity yang dihubungkan melalui relationship.

**Ada tiga jenis cardinality yaitu :**

#### 1. One to One (1:1)

Setiap entitas suatu relasi hanya bisa berhubungan paling banyak dengan satu entitas saja atau satu banding satu.



Gambar 2.16. Cardinality One to one

## 2. One to Many / Many to One (1:M / M:1)

Setiap entitas suatu relasi yang terjadi dengan batasan relasi yang berupa satu berbanding banyak atau banyak berbanding satu.



Gambar 2.17. Cardinality One to Many / Many to One

## 3. Many to Many (M:M)

Setiap entitas suatu relasi dapat berhubungan dengan banyak entitas.



Gambar 2.18. Cardinality Many to Many

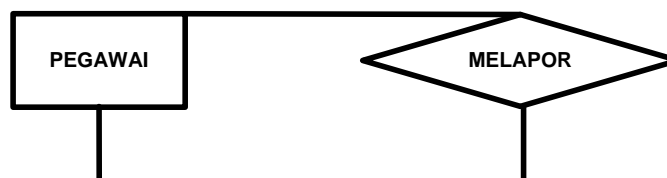
## Derajat Relationship

Menggambarkan sejumlah kemungkinan relasi untuk setiap entitas yang berpartisipasi dalam suatu relationship.

**Derajat relationship memiliki tiga derajat yaitu :**

### 1. *Unary Degree (Derajat Satu)*

Sebuah relasi yang menghubungkan dengan entitas itu sendiri yang berpartisipasi dengan tugas yang berbeda.



Gambar 2.19. Unary Degree

### 2. *Binary Degree (Derajat Dua)*

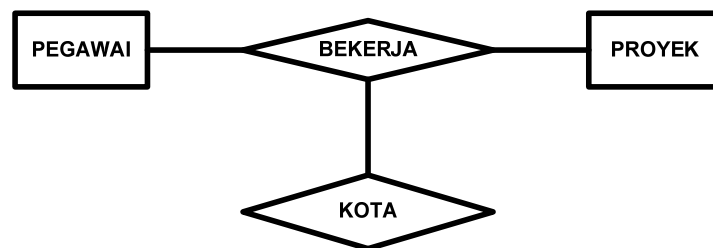
Sebuah relasi yang terdiri dari dua buah entitas yang saling berhubungan.



Gambar 2.20. Binary Degree

### 3. *Ternary Degree (Derajat Tiga)*

Sebuah relasi yang terjadi karena adanya hubungan antara tiga buah entitas.



Gambar 2.21. Ternary Degree

#### 2.2.4 Normalisasi

Normalisasi adalah proses pengelompokan data kedalam bentuk tabel yang menyatakan entitas dan menghubungkan mereka sehingga terwujud satu bentuk yang memudahkan adanya perubahan dengan dampak sekecil mungkin.

Tujuan dari normalisasi yaitu untuk mengurangi kompleksitas, untuk memudahkan pemodifikasian data, serta untuk menghilangkan kerangkapan data.

- **Tingkat-tingkat normalisasi :**

- 1. Bentuk Tidak Normal (Unnormalized form)**

Merupakan kumpulan data yang direkam, tidak ada keharusan mengikuti format tertentu dan dapat saja tidak lengkap. Data dikumpulkan apa adanya sesuai kedatangannya.

- 2. Bentuk Normal Pertama (First Normal Form / 1NF)**

Bentuk normal tahap pertama terpenuhi jika sebuah tabel tidak memiliki atribut bernilai banyak (*Multivalued Attribute*) atau lebih dari satu atribut dengan domain nilai yang sama. Ciri-ciri 1NF adalah setiap data tidakkosong, data dibentuk persatu record, nilai dari field atomic, tidak ada nilai atribut ganda.

- 3. Bentuk Normal Kedua (Second Normal Form / 2NF)**

Bentuk normal tahap kedua terpenuhi jika pada sebuah tabel, semua atribut yang tidak termasuk dalam key primer memiliki ketergantungan fungsional pada key primer secara utuh.

- 4. Bentuk Normal Ketiga (Third Normal Form / 3NF)**

Bentuk normal tahap tiga terpenuhi jika bentuk relasi tersebut mencakup bentuk 2NF dan setiap Non key atributnya tidak tergantung transitif pada primary key.

- 5. Bentuk Normal Boyce-Codd (Boyce-Codd Normal Form / BCNF)**

Bentuk normal tahap ini dikatakan BCNF jika dan hanya jika setiap determinan adalah kandidat key.

- 6. Bentuk Normal Keempat (4NF)**

Bentuk normal tahap keempat berkaitan dengan sifat ketergantungan banyak nilai (*Multivalued Dependency*) pada suatu tabel yang merupakan pengembangan dari ketergantungan fungsional.

### 7. Bentuk Normal Kelima (5NF)

Bentuk normal tahap kelima jika dari bentuk 4NF ketergantungan join yang bukan merupakan candidate key dihilangkan atau nama lainnya yaitu Project-join Normal Form (*PJNF*), yaitu ketegantungan relasi antar tabel (*Join Dependency*).

### 2.3 Sekilas Tentang Microsoft Visual Basic 6.0

Apakah Visual Basic itu? Kata “VISUAL” dimaksudkan untuk satu metode yang digunakan dalam membuat tampilan sebuah program atau aplikasi dengan menempatkan objek-objek atau elemen-elemen pada program atau aplikasi yang dibuat dari pada penulisan sejumlah kode program yang sangat banyak untuk membuat tampilan program atau aplikasi tersebut. Kata “BASIC” adalah “*Beginners All Purpose Symbolic Instruction Code*” yang dibuat oleh **John G. Kemeney** dan **Thomas E. Kurtz** Tahun 1963 di Darmouth College, AS.

Visual Basic 6.0 adalah Bahasa pemrograman tingkat tinggi berbasis Windows yang merupakan bahasa pemrograman berorientasi pada objek (*Object Oriented Programming*) dan mendukung Event-Driven Programming. Seperti aplikasi-aplikasi lainnya, Visual Basic 6.0 juga dipasarkan dalam berbagai versi.

**Beberapa versi dari Visual Basic 6.0 diantaranya adalah :**

- **Standart Edition / Learning Edition :**

Adalah versi standart yang sudah mencakup berbagai sarana dasar dari Visual Basic 6.0 untuk mengembangkan aplikasi.

- **Profesional Edition :**

Versi ini memberikan berbagai sarana ekstra yang dibutuhkan oleh para programmer professional.

- **Enterprise Edition :**

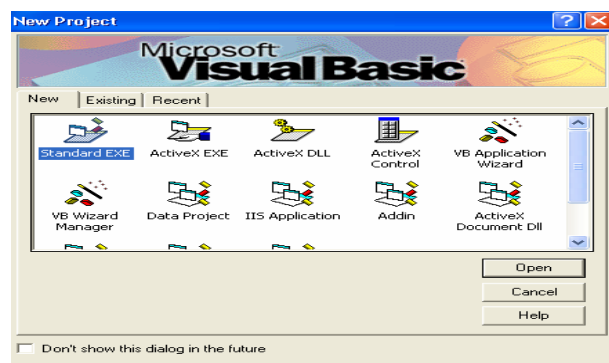
Versi ini dikhususkan untuk para programmer yang ingin mengembangkan aplikasi client server dan biasanya versi ini digunakan untuk membuat aplikasi jaringan.

**Jenis aplikasi yang dapat dibuat dengan Microsoft Visual Basic 6.0 yaitu :**

- Pemanfaatan fasilitas dan access untuk aplikasi database.
- Pemanfaatan aplikasi lain yang dapat kita buat yaitu MS Word, Spread Sheet, MS Excel dan sebagainya dengan memanfaatkan fasilitas OLE.
- Aplikasi yang menggunakan run time DLL (*Dynamic Link Library*) yaitu Pustaka rutin dari bahasa pemrograman lain seperti C, Fortran.
- Aplikasi umum seperti permainan, animasi, multimedia, dan lain-lainnya yang ditampilkan dan dihubungkan kedalam aplikasi tersebut dijalankan.

## 2.4 Menjalankan Visual Basic

Visual Basic juga merupakan sebuah program aplikasi windows. Oleh karena itu untuk menjalankan Visual Basic Anda dapat melalui control menu >start>Programs>Microsoft Visual Basic 6.0. Sebuah kotak dialog akan muncul, disini anda dapat memilih jenis dari program aplikasi yang akan dibuat.



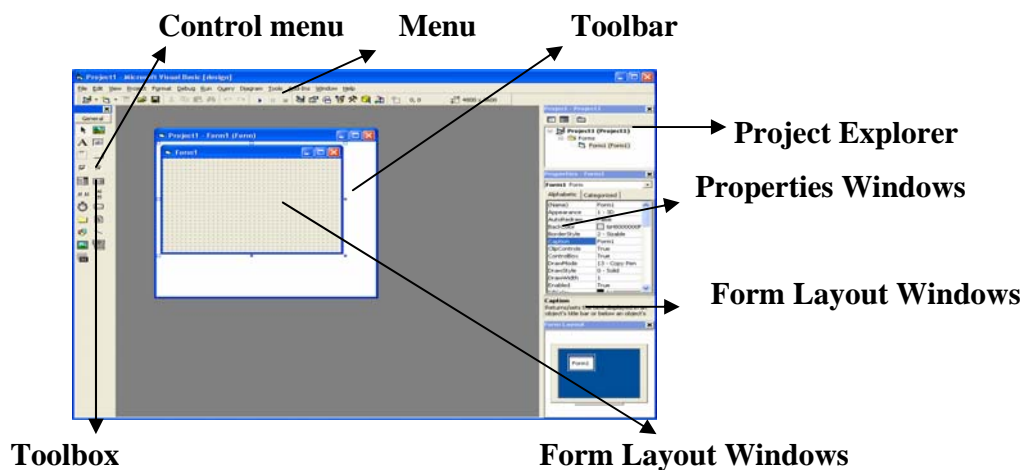
*Gambar 2.22 Kotak dialog New Project*

### 2.4.1 Lingkungan Visual Basic

Lingkungan pengembangan aplikasi Visual Basic berupa layar yang terdiri dari beberapa bagian-bagian kecil yang kesemuanya memiliki sifat:

- **Floating** : dapat digeser-geser ke posisi mana saja.
- **Sizable** : dapat diubah-ubah ukurannya.
- **Dockable** : dapat menempel dengan bagian lain yang berdekatan.

Komponen layar visual basic ini dapat di manipulasi sama seperti memanipulasi jendela Windows.



Gambar 2.23 Lingkungan Visual Basic

### 2.4.2 Control Menu

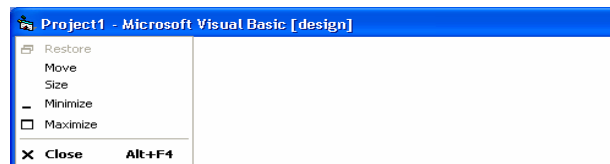
Control Menu adalah menu yang digunakan terutama untuk memanipulasi jendela Visual Basic. Dari menu ini kita bisa mengubah ukuran, memindahkan, atau menutup jendela Visual Basic atau jendela windows lainnya.

Adapun isi dari control menu ini antara lain :

- **Restore** : mengubah ukuran jendela ke ukuran jendela sebelumnya.
- **Move** : untuk memindahkan letak jendela.
- **Size** : untuk mengubah ukuran jendela.
- **Minimize** : untuk meminimalkan ukuran jendela.
- **Maximize** : untuk memaksimalkan ukuran jendela.

- **Close** : untuk menutup jendela.

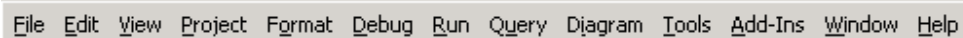
### Control Menu



*Gambar 2.24 Menu Kontrol*

### 2.4.3 Menu Visual Basic

Menu Visual Basic berisi semua perintah Visual Basic untuk melakukan tugas tertentu. Isi dari menu sebagian hampir sama dengan program-program windows pada umumnya. Cara memilihnya bisa dengan mouse atau dengan menekan ALT dan karakter bergaris bawah.



*Gambar 2.25 Menu Visual Basic*

### 2.4.4 Toolbar

Toolbar adalah tombol-tombol yang mewakili suatu perintah tertentu dari Visual Basic. Setiap tombol tersebut dapat langsung diklik untuk melakukan perintah tertentu. Biasanya tombol-tombol ini merupakan perintah-perintah yang sering digunakan dan terdapat pula pada menu Visual Basic.



*Gambar 2.26 Toolbar Standar Visual Basic*

❖ **Kegunaan dari masing-masing toolbar tersebut diatas adalah :**



#### **Add Project**

Adalah untuk menambahkan proyek ke dalam proyek yang sudah ada.

**Add Item**

Adalah untuk menambahkan komponen atau objek kedalam jendela Form.

**Menu Editor**

Adalah untuk menampilkan Menu Editor untuk mengubah tampilan menu.

**Open Project**

Adalah untuk membuka proyek Visual Basic yang sudah ada.

**Save Project Group**

Adalah untuk menyimpan proyek Visual Basic pada komputer.

**Cut**

Adalah untuk memotong Elemen yang dipilih pada layar.

**Copy**

Adalah untuk meng-copy elemen yang dipilih pada layar.

**Paste**

Adalah untuk menyalin elemen yang sebelumnya sudah dipotong (Cut) atau disalin (copy).

**Find**

Adalah untuk mencari teks tertentu

**Undo**

Adalah untuk membatalkan perintah atau tindakan terakhir.

**Redo**

Adalah untuk mengulangi perintah atau tindakan terakhir yang dibatalkan.

**Start**

Adalah untuk menjalankan proyek yang dibuat pada Visual Basic.

**Break**

Adalah untuk menghentikan running program untuk sementara.

**End**

Adalah untuk menghentikan running program.

**Project Explorer**

Adalah untuk menampilkan jendela Project Explorer.

**Properties Window**

Adalah untuk menampilkan jendela Properties.

**Form Layout Window**

Adalah untuk menampilkan jendela Form Layout Window.

**Object Browser**

Adalah untuk menampilkan jendela Object Browser.

**Toolbox**

Adalah untuk menampilkan jendela Toolbox.

**Data View Window**

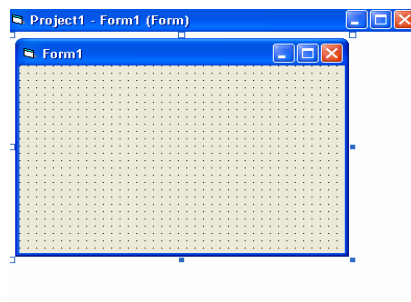
Adalah untuk menampilkan jendela Data View Window.

**Visual Component Manager**

Adalah untuk menampilkan jendela Visual Component Manager.

### 2.4.5 Form Window

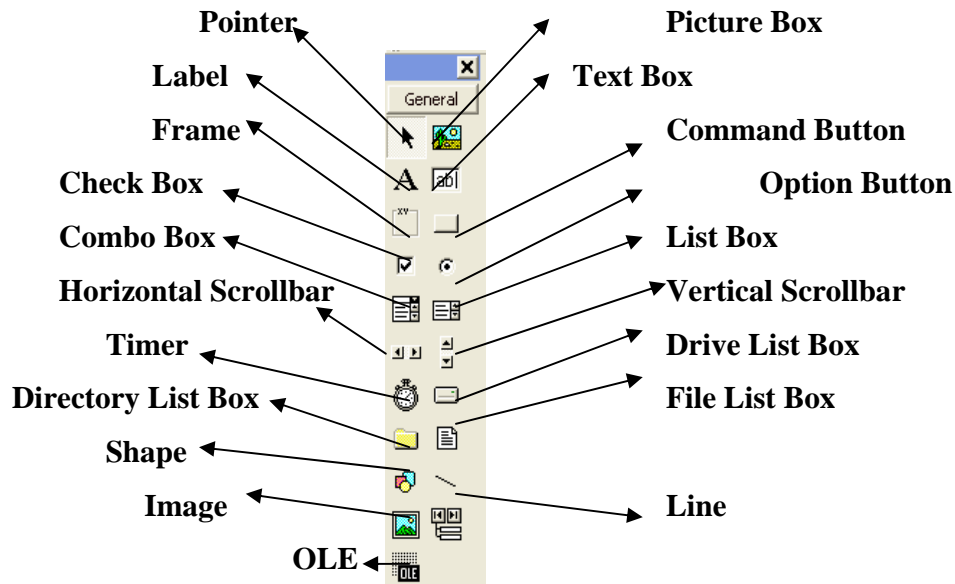
Form Window atau jendela form adalah daerah kerja utama, tempat dimana program-program aplikasi Visual Basic dibuat. Di form window inilah objek interaktif seperti teks, gambar, tombol-tombol perintah, scrollbar dan sebagainya diletakkan. Pada awalnya jendela form ini kelihatan kecil, tetapi ukurannya bisa diubah-ubah sesuai kebutuhan. Jendela Form ini pula yang akan menjadi latar belakang dari aplikasi.



*Gambar 2.27 Jendela Form.*

### 2.4.6 Toolbox

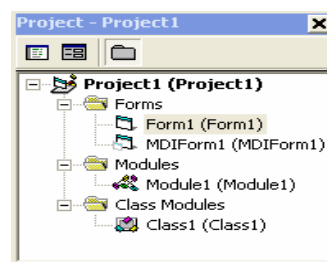
Toolbox adalah sebuah "kotak peranti" yang mengandung semua objek atau kontrol yang dibutuhkan untuk membentuk suatu program aplikasi. Jika baru pertama kali Visual Basic dijalankan maka yang akan muncul adalah Toolbox yang berisi objek-objek standar, akan tetapi objek (control tersebut dapat ditambahkan dengan cara memilih menu Project>Components. Objek yang disediakan oleh Project>Components juga tergantung dari versi Visual Basic itu sendiri.



*Gambar 2.28 Toolbox*

#### 2.4.7 Project Explorer

Jendela Project Explorer adalah jendela yang mengandung semua file didalam aplikasi Visual Basic. Setiap aplikasi dalam Visual Basic disebut dengan istilah Project (proyek), dan setiap proyek bisa mengandung lebih dari satu file. Pada Project Explorer ditampilkan semua file yang terdapat pada aplikasi yang sedang dibuat, misalnya form, modul, class, dan sebagainya.



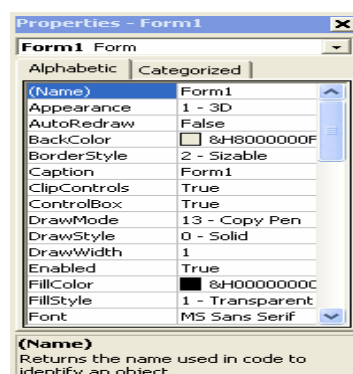
*Gambar 2.29 Jendela Project Explorer.*

#### 2.4.8 Jendela Properties

Jendela Properties adalah jendela yang mengandung semua informasi mengenai objek yang terdapat pada aplikasi Visual Basic. Properties adalah sifat dari sebuah objek, misalnya seperti namanya, warna, ukuran, posisi, dan

lain-lain. Sebagian besar objek memiliki properties yang hampir sama, tetapi ada pula yang berbeda.

Lewat Jendela Properties ini pula pengaturan bentuk dan karakteristik dari setiap objek dilakukan. Dibagian paling atas dari properties terdapat kotak yang menunjukkan nama objek yang sedang aktif, sedang propertinya ditampilkan dibagian bawah Jendela Properties tersebut.



*Gambar 2.30 Jendela Properties*

#### 2.4.9 Form Layout Window

Form Layout Window adalah jendela yang menggambarkan posisi dari form yang ditampilkan pada layar monitor. Posisi form pada Form Layout Window inilah yang merupakan petunjuk dimana aplikasi akan ditampilkan pada layar monitor saat dijalankan nanti. Untuk memindahkan ke posisi yang diinginkan dapat dilakukan dengan cara menggeser gambar form tersebut pada jendela Form Layout Window. Menu yang disediakan pada saat meng-klik kanan form yang dapat dipergunakan untuk mengatur form yaitu:

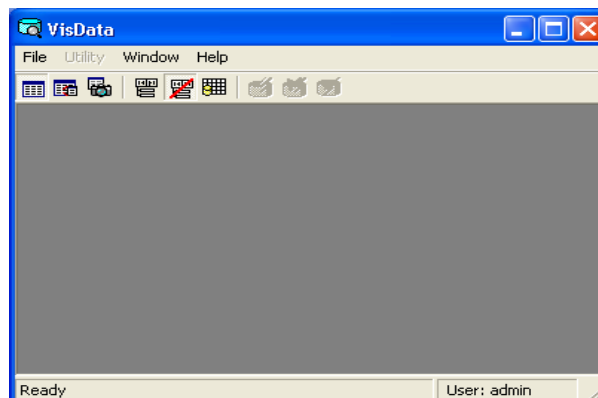
- **Resolution Guides:** menampilkan garis panduan resolusi layar.
- **Startup Position:** menentukan posisi form saat aplikasi dijalankan.
- **Dockable:** membuat Form Layout Window bersifat menempel pada jendela lainnya.
- **Hide:** menyembunyikan Form Layout Window.



sebagai interaksi antara pemakai dengan program. Program akan memberikan reaksi tertentu sesuai event yang didefinisikan.

## 2.5 Visual Data Manager

Visual Data Manager adalah suatu program “Add-Ins”, yaitu program terpisah yang “ditempelkan” pada Visual Basic sehingga bisa dipakai sebagai bagian dari program Visual Basic untuk membuat dan mengedit struktur database. Visual Data Manager ini dapat diaktifkan dengan cara memilih menu Add-Ins pada menu utama Visual Basic, kemudian memilih Visual Data Manager.



Gambar 2.33 Jendela Visual Data Manager

## 2.6 Tipe Data

Visual Basic menyediakan beberapa data seperti table dibawah ini:

| Tipe           | Jangkauan  | Karakter | Ukuran |
|----------------|--|----------|--------|
| <b>Integer</b> | -32.768 s/d 32.767   | %        | 2 byte |
| <b>Long</b>    | -2.147.463.648 s/d -2.147.463.647  | &        | 4 byte |
| <b>Single</b>  | -3.403823 e38 s/d -1,401298 e-45<br>(negatif)<br>1.401298 e-45 s/d 3,403823 e38<br>(positif)                                 | !        | 4 byte |
| <b>Double</b>  | -1,79769313486232 e308 s/d<br>-4.94065645841247 e324(negatif)<br>4,94065645841247 e324 s/d<br>1,79769313486232 e308(positif) | #        | 8 byte |

|                 |  |    |               |
|-----------------|--|----|---------------|
| <b>Currency</b> | -922,337,203,685,477.5808 s/d<br>922,337,203,685,477.5807                      | @  | 8 byte        |
| <b>String</b>   | 0 s/d 65.000 karakter (Win31)<br>0 s/d 2e32 karakter (Win95/NT)                | \$ | 1<br>byte/kar |
| <b>Boolean</b>  | True dan False   |    | 2 byte        |
| <b>Date</b>     | 1 Januari 100 s/d 31 Desember 9999   |    | 8 byte        |
| <b>Object</b>   | Referensi Obyek  |    | 4 byte        |
| <b>Byte</b>     | 0 s/d 255  |    | 1 byte        |
| <b>VARIANT</b>  | Null, Error, Numeric dengan tipe<br>Double,<br>karakter teks, obyek atau array |    | 16 byte       |

*Tabel 2.1 Tipe Data*

## 2.7 Operator

Untuk melakukan berbagai manipulasi dan pengolahan data, Visual Basic menyediakan bermacam-macam operator.

### 2.7.1 Operator Penugasan (Assignment)

Operator penugasan (assignment) disimbolkan dengan tanda sama dengan (=) dan berfungsi memasukkan suatu data kedalam variable.

### 2.7.2 Operator Aritmatika

Digunakan untuk melakukan operasi aritmatika, serta mempunyai hirarki paling tinggi dibanding operator lain. Bila dalam satu baris terdapat operator aritmatika maka operasi pertama yang dilakukan adalah operasi aritmatika, setelah itu baru operator yang lain.

*Operator Aritmatika dengan hirarki paling tinggi dan rendah:*

| <b>Operator</b> | <b>Operasi</b>             |
|-----------------|----------------------------|
| ^               | Pemangkatan                |
| -               | Tanda Negatif              |
| *, /            | Tanda Pembagian            |
| \               | Pembagian Integer          |
| Mod             | Modulus (sisa pembagian)   |
| +, -            | Penambahan dan pengurangan |
| +, &            | Penggabungan String        |

*Tabel 2.2 Operator Aritmatika*

### **2.7.3 Operator Pembagian**

Digunakan untuk membandingkan suatu data dengan data yang lain dan menghasilkan nilai logika (Boolean) benar atau salah. Dua data yang dibandingkan harus memiliki tipe yang sama. Nilai Boolean pada Visual Basic diwakili bilangan integer -1 yang berarti benar (true) dan 0 yang berarti salah (false).

| <b>Operator</b> | <b>Keterangan</b>          |
|-----------------|----------------------------|
| =               | Sama Dengan                |
| <>              | Tidak Sama Dengan          |
| <               | Lebih Kecil                |
| >               | Lebih Besar                |
| <=              | Lebih Kecil Sama Dengan    |
| >=              | Lebih Besar Sama Dengan    |
| Like            | Mempunyai Ciri Yang Sesuai |
| Is              | Sama Referensi Objek       |

*Tabel 2.3 Operator Pembagian*

### 2.7.4 Operasi Logika

Digunakan untuk mengekspresikan satu nilai atau lebih data (ekspresi) logika (boolean) menghasilkan data logika (boolean) baru.

Tabel Operator Logika dengan hirarki dari atas ke bawah:

| Operator | Keterangan   |
|----------|--------------|
| Not      | Tidak        |
| And      | Dan          |
| Or       | Atau         |
| Xor      | Exclusive Or |
| Eqv      | Ekivalen     |

*Tabel 2.4 Operator Logika*

### 2.8 Keluar dari Visual Basic

Untuk keluar dari Visual Basic dilakukan dengan cara mengaktifkan perintah EXIT di menu File atau menekan ALT-F4.

### 2.9 Microsoft Access

Microsoft Access adalah program aplikasi pengolah basis data (database). Paket program ini merupakan sistem manajemen database relational untuk Microsoft windows yang memiliki kemampuan untuk menyortir, mengatur serta melaporkan informasi penting yang dibutuhkan.

Database dalam Microsoft Access merupakan kumpulan dari informasi yang saling berhubungan dan terdiri atas beberapa komponen. Komponen-komponen tersebut disimpan dalam satu file.