

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Gambaran Umum UD Bella Tani**

Pada tahun 1998, UD Bella Tani didirikan oleh Ir.Heru Santoso. UD Bella Tani beralamat di Jl. Raya Baran Nguter dan telah mempunyai cabang diselogiri. Dengan menempati kios yang cukup luas milik Pribadi.

UD Bella Tani merupakan satu Usaha Dagang yang bergerak dibidang penjualan Pupuk, obat-obat pertanian, Bibit padi, secara umum UD ini melayani dan memenuhi kebutuhan Pupuk, obat pertanian, bibit padi kepada Petani (konsumen). dengan usianya yang sekitar 7 tahun saat ini sudah mempunyai cabang di selogiri agar dapat melayani konsumen yang berada diselogiri dan dalam upayanya untuk memperluas pasar.

Untuk mendukung kegiatan usaha saat ini UD Bella Tani mempekerjakan 5 orang karyawan, 2 orang sebagai pelayan dan 3 orang sebagai operasional. inventaris yang dibutuhkan dalam kegiatan usaha 1 unit mobil pick-up untuk mengantar barang langsung kepada petani, 1unit motor, mesin tablet untuk memadatkan pupuk dari serbuk menjadi tablet.

Saat ini manajemen pemasaran yang diterapkan oleh UD Bella Tani adalah dengan membentuk kelompok tani disetiap daerah, yaitu untuk daerah tertentu diorganisir oleh tokoh masyarakat, dengan manajemen tersebut untuk daerah yang membutuhkan pupuk menghubungi UD Bella Tani kemudian dari pihak UD Bella Tani Mengirim Pesanan.

## **2.2 Sistem Perdagangan di UD Bella Tani**

UD Bella Tani merupakan usaha dagang yang bergerak dibidang perdagangan, setiap harinya melakukan transaksi perdagangan yaitu pembelian dan penjualan Pupuk. Dalam kegiatan perdagangan ini, UD Bella Tani mencatat transaksi pembelian pupuk dari supplier dan penjualan pupuk ke konsumen.

UD Bella Tani membeli pupuk dari supplier dan menyalurkannya kepada masyarakat atau petani seperti urea tabur atau tablet, dasar PGP, regent, SP2, Ponska, insektisida, herbisida, fungisida. Perangsang buah atau daun, benih padi, setelah melihat bahwa stok barang sudah menipis. Maka UD Bella Tani melakukan transaksi pembelian dan memperoleh barang beserta nota pembeliannya. Dari transaksi ini ada beberapa hal yang perlu dicatat yaitu mengenai petugas yang melakukan transaksi pembelian, identitas supplier, data barang yang baru saja dibeli dan tanggal terjadi transaksi.

UD Bella Tani menjual barang yang berupa kebutuhan pertanian seperti pupuk Dasar, insektisida, fungisida, herbisida, Perangsang Buah, Benih padi.. Dari transaksi ini UD Bella Tani menyerahkan barang yang dibeli kepada konsumen dengan disertai nota penjualan. Dari transaksi ada beberapa hal yang perlu dicatat yaitu menangani petugas yang melakukan transaksi, identitas konsumen, data barang yang dijual dan tanggal transaksi penjualan.

Dalam sistem persediaan UD Bella Tani selalu mempunyai persediaan barang. Sebelum persediaan habis maka UD Bella Tani melakukan transaksi pembelian pada supplier. Karena jika stock barang habis transaksi penjualan tidak dapat dilakukan.

### 2.3 Pemilihan Perangkat Lunak Dalam Pembangunan Aplikasi

Ada banyak sekali perangkat lunak (*software*) yang dapat digunakan untuk membangun aplikasi. Beberapa hal yang harus dipertimbangkan dalam menentukan perangkat lunak pembangun aplikasi adalah sebagai berikut :

1. Karena aplikasi yang akan dibuat berhubungan dengan basis data, maka perangkat lunak yang dipilih harus dapat menjamin tersedianya fasilitas atau perintah-perintah yang dapat digunakan untuk berinteraksi dengan DBMS secara penuh.
2. Tidak semua perangkat lunak pembangun aplikasi cocok atau dapat digunakan pada arsitektur sistem yang *multiuser*. Karena aplikasi yang dibangun nantinya berbasis *multiuser* maka perangkat lunak yang dipilih harus dapat memberi dukungan yang baik pada sistem *multiuser*.
3. Aplikasi *multiuser* bekerja dengan memakai database secara bersama-sama. Oleh karena itu dipilih perangkat lunak yang menyediakan fasilitas yang mempunyai kemampuan untuk mengakses berbagai jenis database. Aplikasi yang dibuat diharapkan dapat mengantisipasi penggunaan DBMS yang beragam, oleh karena itu perlu dipertimbangkan dalam memilih perangkat lunak pembangun aplikasi, yaitu perangkat lunak yang bisa cocok untuk semua DBMS.
4. Teknologi Informasi senantiasa berubah, DBMS juga selalu diperbaharui. Akan selalu muncul versi-versi baru yang menawarkan lebih banyak kelengkapannya. Maka untuk menghadapi kemungkinan pembangunan itu

dipilih perangkat lunak yang memberi dukungan baik untuk pengembangan dimasa yang akan datang.

Dengan beberapa pertimbangan diatas maka untuk membangun aplikasi pengolahan data penjualan dan pembelian barang berbasis *multiuser* dipilih bahasa pemrograman Visual Basic 6.0

## **2.4 Konsep Dasar Sistem Informasi**

Informasi merupakan hal yang sangat penting bagi manajemen dalam pengambilan keputusan. Dalam dunia usaha dan dunia kerja. Informasi yang tepat, akurat waktu dan relevan sangat penting dan berharga karena sangat penting dalam pengambilan keputusan dan menentukan langkah-langkah yang harus dilakukan untuk mempertahankan dan mengembangkan organisasinya.

### **2.4.1 Informasi**

Sumber dari informasi adalah data. Data adalah bagian dari informasi yang terdiri dari fakta-fakta, gambar-gambar, dan angka-angka dalam sekelompok lambang tertentu yang tidak diacak sehingga dapat menentukan jumlah dan tingkatan.

Informasi menurut *Robert J Verzello / John Reuter III* adalah kumpulan data yang relevan dan mempunyai arti yang menggambarkan suatu kejadian atau suatu kegiatan-kegiatan.

### **2.4.2 Sistem**

Pengertian Sistem terdiri menjadi dua yaitu:

- Menekankan pada Prosedurnya adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk menyelesaikan suatu sasaran tertentu.
- Menekankan pada Elemen atau Komponennya adalah sistem merupakan kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.

Dari definisi diatas dapat disimpulkan sistem merupakan kumpulan dari elemen-elemen yang saling berhubungan satu sama lain dan bertanggung jawab untuk memproses masukan (*input*) atau keluaran (*output*) sesuai dengan yang diharapkan. tahap perancangan sistem mempunyai dua maksud dan tujuan yaitu:

1. Untuk memenuhi kebutuhan pemakai sistem
2. Untuk memberikan gambaran yang jelas rancang bangun yang lengkap kepada pemrogram komputer dan ahli-ahli tehnik lainnya yang terlibat

Tujuan yang kedua ini lebih cenderung pada perancangan sistem yang terinci, yaitu pembuatan rancang bangun yang jelas dan lengkap dimana nantinya digunakan untuk pembuatan program komputer. Untuk menghasilkan *output* yang baik maka sistem harus didukung oleh sistem perangkat keras dan perangkat lunak yang handal. Yang dimaksud sistem pendukung adalah sistem perangkat keras (*hardware*) dan sistem perangkat lunak (*software*).

*Stephen A. Moscove dan Mark G. Simkin* mendefenisikan sistem sebagai berikut : sistem adalah suatu kesatuan yang terdiri dari interaksi sub sistem yang berusaha untuk mencapai tujuan (*goal*) yang sama. Menurut Frederick H. WU: suatu sistem beroperasi dan berinteraksi dengan lingkungannya untuk mencapai

sasaran (*objectives*) tertentu, suatu sistem menunjukkan tingkah lakunya melalui interaksi di antara komponen-komponen di dalam sistem dan di antara lingkungannya. Menurut *M. J. Alexander* di dalam buku *Information System Analysis: Theory and Application*, (*Palo alto, California: Science Research Associates, 1974*): suatu sistem adalah suatu group dari elemen-elemen baik berbentuk fisik maupun bukan fisik yang menunjukkan suatu kumpulan saling berhubungan di antaranya dan berinteraksi bersama-sama menuju satu atau lebih tujuan, sasaran atau akhir dari sistem.

Dari beberapa pengertian mengenai sistem yang diberikan dapat di kelompokkan ke dalam dua kelompok. Yang pertama lebih menekankan pada elemen-elemen dalam sistem. Elemen-elemen atau komponen-komponen atau bagian-bagian dari suatu sistem dapat berupa sub sistem, dan yang kedua lebih menekankan pada prosedur.

Kedua kelompok pengertian ini adalah benar dan tidak bertentangan, yang berbeda adalah cara pendekatannya. Mempelajari suatu sistem akan lebih mengena bila mengetahui terlebih dahulu apakah sistem itu, serta pengertian tentang sistem pertama kali didapat dari defenisinya. Dengan demikian defenisi akan mempunyai peranan yang penting di dalam pendekatan untuk mempelajari suatu sistem. Pendekatan sistem yang merupakan jaringan kerja dari prosedur lebih menekankan urutan-urutan operasi di dalam sistem.

### 2.4.3 Sistem Informasi

Menurut *Henry C Lucas* Sistem informasi adalah suatu kegiatan dari prosedur-prosedur yang diorganisasikan, bilamana dieksekusi dan menyediakan informasi untuk mengambil keputusan dan pengendalian didalam organisasi.

### 2.5 Konsep Perancangan Sistem Informasi


Perancangan sistem informasi dilakukan setelah tahap analisis sistem tahap desain sistem dibagi menjadi dua, yaitu:

1. Desain sistem secara umum atau sering disebut desain konseptual, desain lokal, dan desain secara makro.
2. Desain sistem secara terinci atau sering disebut desain phisik dan desain internal.

Desain sistem dapat berarti:

1. Tahap setelah analisis dari siklus pengembangan sistem,
2. Pendefinisian dari kebutuhan-kebutuhan fungsional,
3. Persiapan untuk rancang bangun implementasi,
4. menggambarkan bagaimana suatu sistem dibentuk,
5. dapat berupa penggambaran, perancangan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi,
6. Termasuk menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem.

Desain sistem mempunyai tujuan utama, yaitu:

-  Untuk memenuhi kebutuhan kepada pemakai sistem

■ Untuk memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada pemrogram komputer.

Agar dapat mencapai tujuan ini, analisis sistem harus dapat mencapai sasaran-sasaran seperti:

1. Desain sistem harus berguna, mudah dipahami dan mudah digunakan,
2. Desain sistem harus dapat mendukung tujuan utama perusahaan,
3. Desain sistem harus efisien dan efektif,
4. Desain sistem harus dapat mempersiapkan rancang bangun yang terinci.

## **2.6 Konsep Pemrograman Multiuser**

Pada lingkungan jaringan komputer, suatu data dapat digunakan secara bersama-sama (*multiuser*). Hal seperti ini menuntut penanganan program yang sedikit berbeda dengan program yang dipakai oleh seorang pemakai. Salah satu permasalahan yang muncul terhadap sistem yang *multiuser* adalah mencegah terjadinya suatu data dimodifikasi oleh dari satu orang pada saat waktu yang bersamaan. Jika keadaan ini tidak diantisipasi maka akan terjadi keadaan yang tidak konsisten.

Multiuser dapat pula diartikan suatu lingkungan yang memungkinkan sejumlah orang untuk berbagi data yang sama pada saat waktu yang bersamaan. Suatu data diletakkan pada suatu komputer yang di sebut server dan data ini dapat digunakan secara bersama-sama oleh pemakai pada komputer lain yang disebut terminal kerja (*workstation*).



## 2.7 Konsep Multiuser Menggunakan Visual Basic 6.0

Konsep multiuser adalah memakai database secara bersama-sama. Oleh karena itu database tersebut harus disimpan di drive jaringan, tetapi program dijalankan pada mesin lokal.

Latar belakang pemrograman berbasis client-server (*multiuser*) tidak lepas dari perkembangan teknologi komputer dan kebutuhan yang muncul pada perusahaan besar (*multinasional*). Pada masa itu proses pengolahan data dipusatkan pada satu komputer yang dihubungkan kesejumlah dumb terminal pada jarak tertentu. Batasan jarak yang terlalu dekat ini tidak memungkinkan mereka untuk berbagi informasi untuk jarak yang jauh, karenanya dikembangkan berbagai cara, baik itu dari segi piranti keras, aturan (*protocol*) bagaimana data dikomunikasikan, sampai pada pengaturan cara memperoleh data guna memungkinkan komunikasi data pada jarak yang jauh.

Seiring dengan perkembangan komputer pribadi yang dimotori oleh Apple Computer dan IBM PC pada akhir tahun 1970an dan awal tahun 1980an, maka makin banyak anak cabang perusahaan besar yang mampu untuk memiliki bagian atau departemen komputer di masing-masing lokasi usahanya

Teknologi LAN(*Local Areal Network*), terutama sistem operasi Novell turut membantu proses tersebut sehingga dengan biaya terjangkau sebuah anak cabang perusahaan dapat memiliki sistem komputer seperti yang dimiliki oleh perusahaan induk pada era tahun 1960-1970an. Istilah *workstation* pun meganti terminologi dumb terminal.

Kini *workstation* dapat memproses data sebelum data dikirim kekomputer server. Sayangnya masalah lain muncul yakni terjadi peningkatan arus lalu lintas data antar komputer server dan *workstation* sehingga mempengaruhi kinerja keseluruhan sistem.

Lalulintas data yang kian ramai ini harus segera diatasi karena makin berkembangnya jenis data dari semula berbasis teks atau database telah ditambah oleh jenis data lain seperti gambar, audio atau music, video yang membutuhkan jalur lalu lintas data lebih besar lagi. Untuk mengatasi hal tersebut, mulai dikembangkan konsep *clien-server* yang berbasis pada aturan bahwa komputer server hanya akan mengirim data yang dibutuhkan oleh *workstation* atau *clien*, dimana proses penyimpanan data dilakukan pada komputer server.

Proses tersebut sedikit banyak dapat mengurangi beberapa permasalahan, baik dari segi lalu lintas data maupun sumber daya dan biaya komputerisasi, karena kini sebuah perusahaan dapat menggunakan komputer berkemampuan rendah sebagai workstation dan memberi alokasi dana lebih besar untuk memperoleh komputer server dengan kemampuan lebih baik. Di samping itu keamanan data lebih terjamin. Salah satu contoh paling populer dari konsep *clien-server* adalah sistem jaringan internet, dimana dengan menggunakan komputer sederhana atau lewat ponsel, kita dapat mengakses data pada jaringan komputer lainnya.

Microsoft corp sebagai penyedia sistem operasi Windows, turut ambil andil dalam dunia komputer ini, Bill Gates, Microsoft Corp. Telah mengembangkan sistem operasi Windows berbasis DNA (*Distributed*

*Internetwork Architectur*), di mana semua komputer Windows dapat di gabungkan menjadi satu jaringan besar yang saling berhubungan, bahkan. Dengan demikian data dapat didistribusikan pada lokasi yang berbeda dan format data dapat dikhususkan penempatan sesuai dengan kebutuhan.

Untuk memperkuat visi tersebut, Microsoft telah mengembangkan konsep atau gagasan *Uneversal Data Access* yaitu gagasan untuk mengakses berbagi jenis data file pada suatu jaringan dengan cara yang sama. Salah satu contoh dari konsep atau gagasan tersebut adalah pengembangan metode atau model pengaksesan data yang dikenal dengan model ADO (*Active Data Objects*).

Pada saat ini telah tersedia jenis provider atau penyedia file untuk mengimplementasikan model ADO pada *Database Managenent System* (DBMS), yakni untuk SQL Server. Dengan menggunakan model ADO di dalam penulisan program, sebuah program nantinya dapat mengakses database, baik itu dalam format SQL Server maupun yang lainnya

## **2.8 Mengenal Visual Basic 6.0**

Visual basic merupakan bahasa pemrograman yang dikendalikan oleh kejadian (*Even Driven*), yang artinya menunggu sampai adanya respon dari pemakai berupa kejadian tertentu misalnya tombol diklik atau memilih menu. BASIC (*Beginer All Purpose Symbolic Instruction Code*) adalah sebuah bahasa pemrograman kono yang merupakan awal dari bahasa-bahasa pemrograman tingkat tinggi lainnya.

Visual Basic IDE (*Interface Develoment Environment*) adalah tampilan antarmuka program dengan menggunakan yang sudah bersifat GUI(*Graphical*

*User Interface*) yang menyajikan banyak kemudahan bagi programmer untuk dapat membuat dan merancang user friendly dari aplikasi yang dibuat berupa kelebihan Visual Basic 6.0 adalah:

- ➔ Visual basic 6.0 disertai dengan berbagai sarana untuk membuat aplikasi data base, sarana database visual basic, yang menjadikannya sarana terbaik untuk mengembangkan aplikasi client/server.
- ➔ Autolist, program akan otomatis menampilkan method atau property dengan mengetik dot (.). dengan adanya fasilitas autolist ini dapat memperkecil kesalahan pengguna atau programmer dalam membuat program.
- ➔ Visual basic bahasa pemrograman even driven artinya program menunggu sampai adanya report dari pemakai berupa kejadian tertentu, misal tombol dikelik atau memilih menu.
- ➔ Memiliki Compiler adalah yang dapat menghasilkan file executable yang lebih cepat dan lebih efisien dari sebelumnya
- ➔ Kemampuan membuat activeX dan fasilitas internet yang lebih banyak.

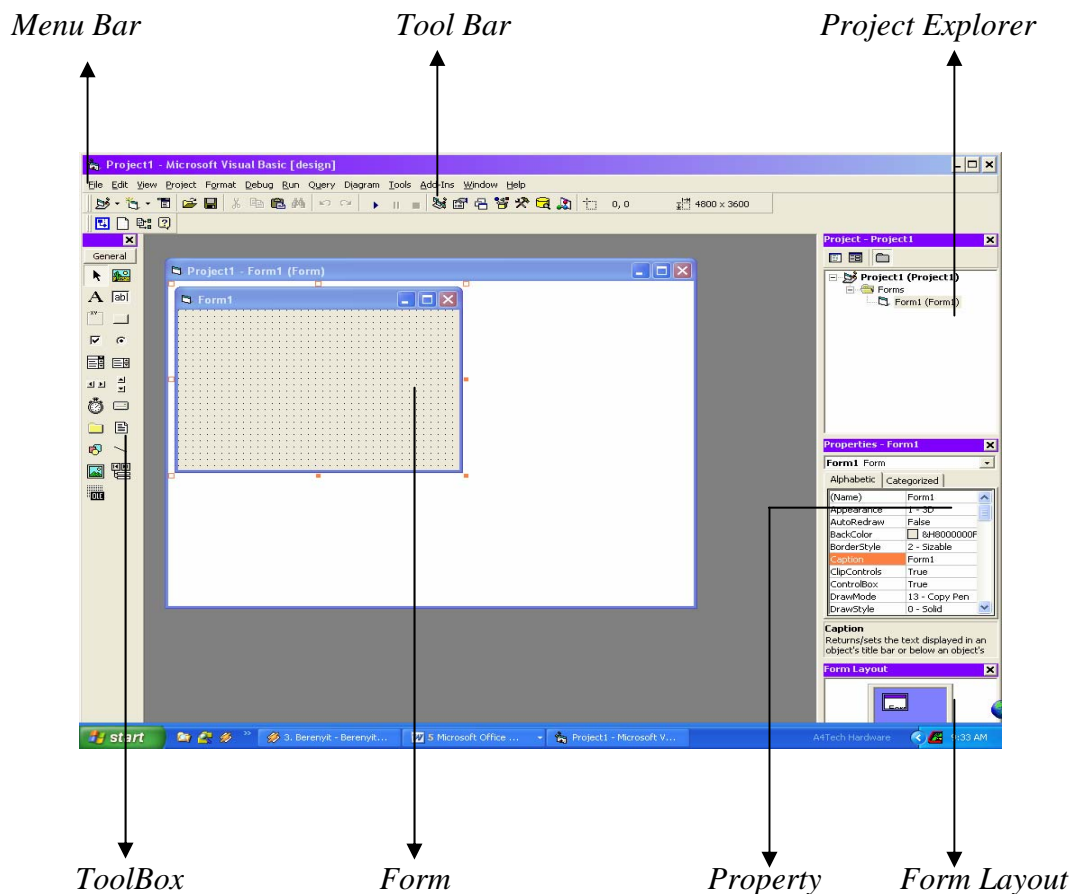
## **2.9 Aplikasi Visual Basic 6.0**

Aplikasi(Proyek) Visual Basic 6.0 terdiri dari:

- Form adalah Windows/jendela dimana anda akan membuat user interface/tampilan.
- Kontrol/Control adalah Tampilan berbasis grafis yang dimasukkan pada form untuk membuat interaksi dengan pemakai komputer (text box, label, scroll bar, tombol command).

- Properti adalah nilai yang dimiliki sebuah objek visual basic. Visual Basic menerapkan property standar, anda dapat mengubah property saat mendisain program atau saat program dijalankan.
- Metode adalah serangkaian perintah yang sudah tersedia pada suatu objek yang dapat diminta untuk mengerjakan tugas khusus.
- Prosedur Kejadian adalah Kode yang berhubungan dengan suatu objek. Kode ini akan dieksekusi ketika ada respon dari pemakai berupa event tertentu.
- Prosedur Umum adalah kode yang tak berhubungan dengan suatu objek. Kode ini harus diminta oleh aplikasi.
- Modul atau Module adalah Kumpulan dari prosedur umum, deklarasi variable dan definisi konstanta yang digunakan oleh aplikasi.

## 2.10 Tampilan Layar Visual Basic 6.0



Gambar 2.1 Menu Utama Visual Basic 6.0

- Menu Bar merupakan menu untuk perintah VB 6.0
- Tool Box adalah kumpulan dari objek yang digunakan untuk membuat user interface serta control bagi program aplikasi.

Komponen antar muka kepada memakai antara lain

### 1. Text Box

Kontrol ini bisa terdiri seperti sebuah kotak, yang digunakan untuk menginput data atau menampilkan data.

## 2. Command Button (*Tombol Tekan*)

Kontrol ini berupa suatu kotak menyerupai tombol yang didalamnya berisi label singkat menyerupai tindakan yang akan dilakukan jika tombol ini ditekan.

## 3. List Box (*Kotak Daftar*)

Untuk memasukan input nilai kedalam aplikasi dengan cara memilih dari daftar yang ada.

## 4. Option Buton (*Tombol Pilihan*)

Option Buton adalah sebuah kontrol dimana user dapat memilih salah satu dari beberapa pilihan yang ada.

## 5. Check Box (*Kotak Cek*)

Digunakan untuk memilih yang fungsinya hampir sama dengan Option Buton, namun pada Check Box anda dapat memilih lebih dari satu.

## 6. Combo Box (*Kotak Combo*)

Combo Box merupakan kotak pilihan yang sama seperti list box perbedaannya user dapat mengisikan nilai baru pada kotak combo tersebut

- Form adalah pusat dari pengembangan aplikasi Visual Basic.
- Project Explorer menampilkan daftar form dan modul proyek.
- Property berisi daftar struktur setting property yang digunakan pada sebuah objek terpilih.
- Tool Bar merupakan shortcut yang digunakan untuk membuat perintah suatu proyek.
- Form Layout menampilkan posisi form relative terhadap layar monitor.

## 2.11 Perintah-perintah yang digunakan dalam Visual Basic 6.0

Sebelum mengenal lebih jauh perintah yang digunakan dalam Visual Basic 6.0 ada baiknya mengetahui cara kerja Visual Basic 6.0. Visual Basic 6.0 merupakan bahasa pemrograman yang berbasis OOP (Object Oriented Programming), sebuah program dibagi menjadi bagian-bagian kecil yang disebut dengan *objek*. Setiap objek memiliki entiti yang terpisah dengan entiti objek-objeklain dalam lingkungannya. Objek-objek yang terpisah ini dapat diolahsendiri-sendiri, dan setiap Objek memiliki sekumpulan sifat dan metode yang melakukan fungsi tertentu sesuai dengan yang telah kita programkan kepadanya.

Setiap Objek mengandung tiga hal utama seperti dibawah ini:

### 1. Properti atau Atribut

*Properti* adalah karakteristik atau sifat dari sebuah objek.misalnya properti warna untuk teks adalah hitam.

### 2. Metode

*Metode* adalah serangkaian prosedur yang dimiliki oleh suatu objek yang akan dijalankan sesuai dengan respon yang diberikan oleh suatu perintah atau kejadian. Misalnya, Objek tombol EXIT memiliki metode untuk keluar dari aplikasi, Untuk mengatur metoded dengan menggunakan *Jendela Code*.

### 2. Event

Event adalah “Kejadian” segala sesuatu yang dapat dialami oleh sebuah objek yang diakibatkan baik oleh tindakan user atau tindakan dari program itu



sendiri. Sebagai contoh, meletakkan mouse diatas objek, mengklik tombol mouse pada sebuah tombol mengklik pada kotak teks, dan sebagainya.

Perintah-perintah dalam Visual Basic 6.0

Pada Visual Basic 6.0 untuk membuat perintah dengan menuliskan program pada jendela code dari metode yang diprintahkan akan memicu Event.

Memberi nama variabel

Perintah yang digunakan Dim (Dimension)

☐ Memberi nama caption pada objec

Untuk memberi nama pada objek dengan memilih caption pada jendela properti

☐ Memberi nama pada object

Dalam memberi nama sesuai dengan nama kontrol atau variabel dan fungsinya karena nama inilah yang akan dikenali oleh program.

☐ Memanggil form

Dalam menu utama terdapat kontrol untuk memanggil form2 atau form3, perintah yang digunakan load *nama form nama form.Show* sedangkan perintah yang digunakan untuk menutup form adalah Unload *nama form*.

☐ Keluar dari aplikasi

Printah yang digunakan untuk keluar dari program adalah *End* atau *Unload Me* keluar dari form.

☐ Pernyataan kesalahan

Pernyataan ini akan memberikan program apa yang terjadi jika terjadi kesalahan. Jenisnya ada 3 macam yaitu:

### 1. On Error Go To Line

Jika terjadi kesalahan run-time, kontrol akan melompat ke baris dan membuat penanganan kesalahan menjadi aktif.

### 2. On Error Resume Next

Khusus untuk kesalahan run-time kontrol akan mengembalikan kendali ke pernyataan setelah pernyataan yang menyebabkan kesalahan.

### 3. On Error Go To 0

Tidak mengaktifkan semua penanganan kesalahan pada prosedur sekarang.

#### Membuat pesan

Perintah untuk membuat pesan msgbox adalah *intNamaVariabel = MsgBox(Pesan[, intType, strjudul])*.

Menambahkan item, menghapus item, menghitung jumlah item pada listBox pada listbox perintah yang digunakan untuk menambahkan item adalah *List1.AddItem* "nama item", menghapus item adalah *List1.RemoveItem* atau *List1.Clear*, sedangkan untuk menambahkan *IntJumlah>List1.ListCount*

#### membuat Kotak Input

adapun perintah yang digunakan

*strNamaVariabel=InputBox(Pesan[, strjudul, strdefault, intXpos, intYpos])*

#### menambahkan Record ke database

menggunakan metode *AddNew* dalam sebuah prosedur kejadian untuk membuka record baru dalam database

### ☐ Menghapus Record dari Database

Untuk menghapus record dari database, pertama harus menampilkan record yang akan dihapus kemudian memakai metode delete dengan objek Record set untuk menghapus record.

### ☐ Mencari Data

Merupakan Prosedur pencarian data dan menampilkan sebuah kotak dialog pencarian untuk mendapat string pencarian dari pemakaian. Prosedur pencarian dengan metode Seek akan dicari item data dari field Title dalam database dari awal sampai akhir, sampai ditemukan yang sesuai atau sampai mencapai akhir daftar. Jika tidak ditemukan, sebuah pesan akan ditampilkan dan record pertama dari *recordset* akan muncul

### ☐ Perintah untuk memperbahatui *Update*

### ☐ Perintah untuk menghubungkan dengan SQL *Connect SQL*

### ☐ Perintah untuk menghapus *Delete*

### ☐ Perintah untuk selalu baru *refres*

### ☐ Perintah untuk memberi warna *ForeColor*

### ☐ Perintah untuk memberi nilai benar *True*

### ☐ Perintah untuk memberi nilai salah *Fals*

### ☐ Perintah untuk merekam data *RecordSet*

### ☐ Perintah untuk memfokuskan pada kontrol tertentu *SetFocus*

## 2.12 SQL Server

SQL adalah sebuah database relational yang dirancang untuk mendukung aplikasi dengan arsitektur *Client/Server*, dimana database terdapat pada komputer pusat yang disebut server, dan informasi digunakan bersama-sama oleh beberapa user yang menjalani aplikasi yang ada pada komputer lokalnya yang disebut *client*.

Ada banyak hal yang dipelajari oleh DBMS SQL Server disamping itu ia dapat juga dijadikan sebagai pengantar yang baik untuk memahami kerja DBMS lainnya. Untuk mengetahui beberapa hal penting pada SQL server, kita bisa memulainya dari penggambaran struktur SQL server.

### 2.12.1 Struktur SQL Server 2000

Pada struktur sql server ini ada suatu model yaitu ADO (*Active Data Object*) dimana ADO tersebut membagi sebuah objek besar menjadi kumpulan objek kecil. Ada empat objek utama pada model ADO, yaitu objek Connection, objek comand, objek recordset, dan objek file. Objac tersebut memiliki keterkaitan dengan pengkodean program. Tanpa objek tersebut program tidak dapat mengakses data maupun memanipulasi database. Lewat model ini dapat dipelajari beberapa hal yang menarik, yaitu:

1. Semua permintaan yang masuk ke sql akan ditampung oleh job server. Selanjutnya permintaan tersebut akan diproses kemudian diurutkan berdasarkan FIFO (*Frist In Frist Out*) yang masuk lebih dahulu yang akan diproses. Dengan cara inilah SQL Server menghindari proses yang mungkin berbenturan.

2. Setiap terjadi perubahan pada database melalui perintah SQL (Insert, Update, Delete) SQL Server akan menyimpan transaksi dalam log file. Sehingga jika terjadi kegagalan mengubah database, sql server akan mengulangi proses yang sama.
3. Pada proses pembaharuan (*Update*) database, SQL Server akan membandingkan record yang akan di update dengan record yang ada pada database untuk setiap nilai kolomnya, jika ada perbedaan pada salah satunya sql server akan mengirim kesalahan bahwa record tersebut telah diubah oleh pengguna lainnya. Untuk mengubah record tersebut, operator harus menyegarkan kembali record tersebut dengan cara mengambil dari database kemudian melakukan perubahan yang diinginkan.

### **2.12.2 Metode Akses SQL Server**

untuk mengakses berbagai jenis data (format file) microsoft dan penyedia software menyediakan piranti file untuk keperluan tersebut. Pada saat ini terdapat dua kelompok provider/penyedia file yakni ODBC (*Open Database Conectivity*) dan OLE DB (*Object Link Embledded Database*).

Baik piranti ODBC maupun OLE DB berisi fungsi program API (*Aplication Program Interface*) untuk mengakses atau memanipulasi data/file. Ini berarti untuk mengakses menggunakan fungsi API sangat melelahkan dan merepotkan. Untuk mengatasi persoalan tersebut maka microsoft mengembangkan konsep/model pengaksesan data yang dikenal dengan nama AD (*Active X Data Objec*).

API ADO bekerja dengan cara memodelkan sebuah objek besar menjadi kelompok kecil. Lewat kelompok inilah sebuah file/database dimanipulasi sehingga anda tidak perlu dipusingkan oleh fungsi API karena kompleksitas pemrograman telah diambil alih oleh piranti (provider/penyedia) model ADO pada microsoft telah mengimplementasikan metode/model ADO pada microsoft Visual Basic 3.0 terdapat model DAO adalah model pengaksesan untuk file berformat Ms.Access, Text, foxpro, dan Dbase maupun SQL server.

Model DAO memanfaatkan API ODBC untuk mengakses atau memanipulasi file. Kelemahan dari model DAO ada pada keterbatasan jenis file yang akan diakses. File berformat email, brosur, dalam sumber daya, dan model DAO yang bertingkat membuatnya kurang fleksibel dibandingkan dengan model ADO.

### **2.13 Konsep Perancangan Basis Data**

Didalam suatu sistem pengolahan data membutuhkan suatu basis data yang terstruktur karena dari pengolahan data tersebut dapat diketahui bagaimana data didapatkan, diorganisasi, diakses, dipelihara, dikontrol dan diupdate atau dengan kata lain basis data merupakan komponen yang sangat penting dalam membentuk suatu sistem pengolahan data.

Sistem basis data terdiri dari empat komponen, yaitu data, perangkat keras, perangkat lunak, dan brainware.

- *Data* merupakan sekumpulan fakta yang akan diolah menjadi informasi. Data ini akan dikumpulkan pada suatu media penyimpan dalam bentuk basisdata.

- *Perangkat Keras* merupakan piranti komputer yang digunakan untuk menyimpan dan memproses data yang telah disimpan.
- *Perangkat Lunak* merupakan program-program yang digunakan untuk memberi perintah pada komputer untuk mengerjakan suatu pekerjaan tertentu.
- *Brainware* merupakan orang yang berkepentingan dengan data serta hasil pengolahan data.

### **2.13.1 Definisi Basis Data**

Dalam suatu program pengolahan data, basis data didefinisikan sebagai kumpulan berkas-berkas yang mempunyai kaitan antar suatu file dengan file yang lain sehingga membentuk suatu databases untuk menginformasikan kegiatan suatu perusahaan, instansi dalam batasan tertentu.

### **2.13.2 Kegunaan suatu Basis Data**

Penyusunan suatu basis data digunakan untuk mengatasi masalah-masalah pada penyusunan data, yaitu:

#### **a. Redudansi dan Inkonsistensi**

Penggandaan data pada berkas-berkas yang berbeda (redudansi) serta tidak konsistennya data merupakan hal-hal yang perlu dihindari, sehingga perlu disusun suatu database.

#### **b. Kesulitan Pengaksesan Data**

Apabila suatu saat dibutuhkan untuk mencetak suatu daftar tertentu sementara belum tersedia program untuk menulis data-data pada daftar tersebut maka terjadilah kesulitan dalam pengaksesan data.

c. Isolasi Data untuk standarisasi

Seharusnya data dalam suatu database dibuat suatu format sehingga mudah dibuat program aplikasinya.

d. Masalah keamanan data (*Security*)

Tidak semua pemakai sistem database diperbolehkan mengakses suatu data, oleh karena itu diperlukan keamanan suatu data. Keamanan ini dapat diatur lewat program yang dibuat oleh pemrograman.

e. Masalah Kesatuan

Databases berisi berkas-berkas yang saling berkaitan, masalah utama adalah bagaimana kaitan antara file tersebut terjadi atau bagaimana kesatuan dari berkas-berkas tersebut.

f. Masalah kebebasan data

Perubahan apapun dalam database sebaiknya dapat diatasi tanpa harus merubah program yang telah dibuat inilah yang disebut kebebasan data.

### 2.13.3 Struktur Basis Data

Setiap tabel terdiri atas baris (*row*) dan kolom (*Colum*). Baris sering disebut *record* atau *tupple*, kolom sering disebut field atau atribut. Setiap baris memuat satu kesatuan. Jumlah baris sering disebut cardinality atau kardinalitas. Jumlah field sering disebut degree atau arity.

Supaya data dapat diakses dengan lebih mudah dan cepat, maka data harus diurutkan berdasarkan suatu field tertentu. Field yang dijadikan dasar pengurutan tersebut disebut field kunci (*key field*). Ada beberapa katagori kunci :



1. *Primary Key* (Kunci Utama), kunci utama yang dijadikan dasar pengurutan.
2. *Secondary Key* (Kunci Sekunder), kunci kedua yang dijadikan dasar pengurutan.
3. *Foreign Key* (Kunci tamu), field yang dijadikan dasar hubungan dengan tabel lainnya.
4. *Candidate Key* (Kunci Kandidat), field yang mempunyai kemungkinan untuk dijadikan field kunci.

Biasanya field yang dijadikan kunci adalah field yang datanya bersifat unik. (artinya tidak mungkin ada duplikasinya).

#### **2.13.4 Normalisasi**

Normalisasi adalah teknik desain database dengan memecah atribut-atribut menjadi beberapa bagian kelompok/entity sehingga didapat tabel yang optimal atau normal. Dalam arti tidak mengalami anomaly data, memenuhi functional dependency dan tidak mengalami transitive dependency. Normalisasi dipilih jika kita mendapatkan struktur tabel lengkap dan jelas

Tahap Normalisasi, Yaitu :

1. Bentuk Tak Normal (*Unnormal Form*)

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan mengikuti suatu format tertentu. Dapat saja data tidak lengkap atau terduplikasi. Data dikumpulkan apa adanya sesuai dengan saat menginput.

Ciri-cirinya :

- Tidak harus mengikuti format tertentu
- Data tidak lengkap dan terduplikasi
- Disusun sesuai dengan kedatangan

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

Suatu relasi 1NF jika dan hanya jika sifat dan setiap relasi atributnya bersifat atomik. Atom adalah zat terkecil yang masih memiliki sifat induknya. Bila dipecah lagi maka ia tidak memiliki sifat induknya.

Ciri-cirinya :

- Setiap data dibentuk dengan flat file (*file datar/rata*)
- Data dibentuk dengan satu record demi record dan bersifat atomic value
- Tidak ada multivalued attribute

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

Bentuk normal kedua mempunyai syarat bentuk data telah memenuhi kriteria bentuk normal kesatu. Atribut bukan kunci haruslah bergantung secara fungsi pada *primary key*. Jadi untuk membentuk normal kedua haruslah sudah ditentukan kunci-kunci field. Kunci field haruslah unik dan dapat mewakili atribut lain yang menjadi anggotanya.

Ciri-cirinya :

- Memenuhi criteria 1NF
- Atribut bukan kunci (*non key*) bergantung fungsi pada kunci utama (*Functional Depedency*)

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

Untuk menjadi bentuk normal ketiga maka relasi haruslah dalam bentuk normal kedua dan semua atribut bukan primer tidak punya hubungan transitif. Dengan kata lain, setiap atribut bukan kunci harus bergantung hanya pada primary key dan pada primary key secara menyeluruh

Ciri-cirinya :

- Memenuhi criteria 2NF
- Tidak terdapat Transitive Dependency (*kebergantungan transitive*)

#### 5. *Boyce Codd Normal Form* (BCNF)

BCNF mempunyai paksaan lebih kuat dan bentuk normal ketiga. Untuk menjadi BCNF, relasi harus dalam bentuk normal kesatu dan setiap atribut harus bergantung fungsi pada atribut super key.

Ciri-cirinya :

- Relasi memenuhi kriteria 1NF
- Lebih kuat dari 3NF
- Setiap atribut harus bergantung fungsi pada atribut superkey

### **2.13.5 Entity Relationship**

Pemodelan data merupakan proses untuk merepresentasikan pandangan pengguna terhadap data. Bagian ini merupakan bagian yang penting dalam pengembangan aplikasi basis data. Salah satu pemodelan data adalah model E/R (*Entity Relationship*).

#### **2.13.5.1 Mengapa diperlukan model E/R**

1. Dapat menggambarkan hubungan antar entity dengan jelas.
2. Dapat menggambarkan batasan jumlah entity dan partisipasi antar entity.

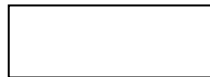
3. Mudah dimengerti oleh pemakai.
4. Mudah disajikan oleh perancang database.

### 2.13.5.2 Komponen-komponen yang terdapat dalam Entity Relationship

#### Model:

1. *Entity* merupakan sesuatu yang bisa diidentifikasi dalam lingkungan kerja dari pengguna.

Simbol yang digunakan dalam Entity adalah persegi panjang



2. *Atribut* mendeskripsikan karakteristik dari entitas yang bersangkutan.

Terdapat dua jenis atribut yaitu:

- a. Identifier(Key), yaitu untuk menentukan entitas secara unik.
- b. Descriptor(nonkey Atribut untuk menentukan karakteristik dari suatu entitas yang tidak unik.

Simbol yang digunakan adalah oval



3. *Identifier* merupakan atribut dari suatu entitas yang mewakili entitas tersebut, jadi bisa dikatakan bahwa identifier merupakan kunci.
4. *Relationship* merupakan hubungan antara satu entitas dengan entitas yang lain. Suatu relasi dapat mempunyai atribut. Suatu relasi dapat terdiri atas lebih dari dua entitas.

Ada tiga kemungkinan relasi antar entitas :

a. *One to One* (1 : 1)

Yaitu hubungan satu entitas lawan satu entitas

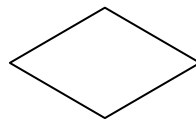
b. *One to Many* (1 : M)

Yaitu hubungan satu entitas dengan banyak entitas

c. *Many to Many* (N : M)

Yaitu hubungan banyak entitas lawan banyak entitas

Simbol yang digunakan dalam Relationship adalah belah ketupat



#### 5. Derajat Relationship

Derajat relationship menyatakan jumlah entitas yang berpartisipasi didalam suatu relationship.

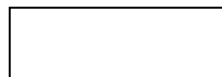
- a. *Unary degree* (derajat satu) adalah derajat yang memiliki satu relationship untuk satu entitas.
- b. *Binary degree* (derajat dua) adalah derajat yang memiliki satu relationship untuk dua entitas.
- c. *Tenary degree* (derajat tiga) adalah derajat yang memiliki satu relationship untuk tiga atau lebih entitas.

### 3.14 Diagram Alir Data

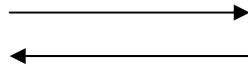
Diagram Alir Data (DAD) merupakan alat yang digunakan untuk mendokumentasikan proses didalam sistem. Dalam DAD akan ditunjukkan bagaimana data itu mengalir, menjelaskan darimana data itu berasal dan kemana data itu mengalir. DAD juga menunjukkan pengguna yang memproses data tersebut, akan diproses manjadi apa dan akan diberikan kemana hasil akhir proses tersebut.

DAD menggunakan notasi-notasi untuk menggambarkan arus dari dalam sistem yang tersrtuktur dan jelas. Notasi itu meliputi kesatuan luar (*Exsternal Entity*), arus data (*Data Flow*), Proses (*Process*), dan Untuk menyimpan Data (*Data Storage*). Pada data analisis, penggunaan notasi ini sangat membantu dalam mengkomunikasikan dengan pemakai sistem untuk memahami sistem secara logika. DAD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau untuk sistem yang baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut disimpan dan mengalir. Simbol atau lambang yang digunakan sebagai berikut:

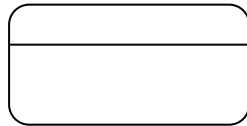
1. Kesatuan Luar (*External Entity*) melambangkan suatu entitas yaitu suatu bagian yang berperan dalam sistem, merupakan asal data dan tujuan kemana data akan mengalir.



2.. Arus Data (*Data Flow*) dilambangkan dengan anak panah yang menuju ke atau dari proses, yang mengalir diantara proses (*Procces*), simpanan data (*Data Storage*), dan kesatuan luar (*External Entity*)



3. Proses (*Procces*) proses menunjukkan transformasi dari masukan menjadi keluaran, dalam hal ini sejumlah masukan dapat menjadi hanya satu keluaran.



4. Menyimpan Data (*Data Srorage*) digunakan untuk memodelkan kumpulan data atau paket data. Dapat didefinisikan sebagai file atau basis data dalam bentuk tape magnetis, disk atau model dalam dbms lainnya.

