This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Sebuah sistem terdiri atas bagian-bagian yang bergabung untuk suatu tujuan tertentu, sebuah sistem bisa terdiri dari bagian-bagian yang saling berkaitan yang beroperasi bersama untuk mencapai sasaran atau maksud tertentu. Menurut JOGIYANTO [5] [1989, hal 1] adalah

> "Suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu".

Dari sistem tersebut dapat disimpulkan bahwa sistem adalah kumpulan dari elemen-elemen yang saling berhubungan dan bertanggung jawab memproses masukan atau input, sehingga menghasilkan keluaran atau output sesuai dengan keinginan.

Setiap sistem tidak peduli betapa kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem yang mana masing-masing menjalankan suatu fungsi dan mempengaruhi proses secara keseluruhan.

2.2 Konsep Dasar Informasi

Informasi sangat penting dalam suatu organisasi, sehingga suatu sistem yang kurang informasi akan menjadi luruh dan akhirnya berakhir. Informasi menurut JOGIYANTO [5] [1989, hal 8] dapat didefinisikan sebagai berikut :

> "Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya".

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian yang terjadi pada saat tertentu dan kesatuan nyata informasi yang diperoleh. Informasi yang diperoleh dari hasil pengolahan data harus mempunyai kualitas yang bergantung pada tiga hal berikut ini :

1. Akurat

   Berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan.

2. Tepat

   Berarti informasi yang datang kepada penerima tidak boleh terlambat. Informasi yang usang tidak memiliki nilai lagi karena informasi merupakan landasan bagi pengambilan keputusan.

3. Relevan

   Berarti informasi tersebut mempunyai manfaat yang sesuai bagi pemakainya.

2.3 Konsep Dasar Sistem Informasi

Informasi merupakan hal yang sangat penting bagi manajemen didalam pengambilan keputusan, sistem informasi menurut JOGIYANTO [5] [1989, hal 11] dapat didefinisikan sebagai berikut :

> "Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan".

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

Suatu sistem informasi memiliki komponen-komponen sistem informasi, yaitu :

1. Blok Masukan

   Input mewakili data yang masuk kedalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model

   Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran

   Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi

   Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan.

   Teknologi terdiri dari tiga bagian utama, yaitu teknisi (brainware atau humanware), perangkat lunak (software) dan perangkat keras (hardware).

5. Blok Basis Data

   Basis data (database) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

6. Blok Kendali

   Pengendalian yang dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.4 Metode Pengembangan Sistem

Metode pengembangan sistem menurut BAMBANG HARIYANTO [4] [2004, hal 38] ada empat macam, yaitu :

1. *Linear Sequential Model* atau *Waterfall*
2. *Prototyping Model*
3. *Spiral Model*
4. *Rapid Application Development* (RAD) *Model*

Penjelasan dari keempat motode diatas adalah sebagai berikut :

1. *Linear Sequential Model*

Model ini pertama kali dikemukakan oleh Royce. Model ini sering disebut model klasik atau *waterfall*. Model ini menyarankan pendekatan pengembangan secara sekuen dan sistematik untuk pengembangan perangkat lunak dimulai di level sistem, berlanjut ke analisis, lalu perancangan, pemrograman, pengujian dan pemeliharaan. Model ini merupakan model yang tertua.

Kelemahan model ini adalah :

a. Proyek-proyek nyata jarang mengikuti alur sekuen yang diusulkan model. Meskipun *linear model* dapat mengakomodasikan iterasi, namun model melakukan secara tidak langsung sebagai hasilnya, perubahan-perubahan dapat menyebabkan kebingungan saat tim pengembangan melakukannya.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

b. Sering *customer* sulit menyatakan semua kebutuhannya secara eksplisit. Model ini memerlukan pernyataan eksplisit itu dan sulit mengakomodasikan ketidakpastian yang terdapat diawal dari kebanyakan proyek.

c. *Customer* harus memiliki kesabaran. Versi yang dapat bekerja dari program tidak akan tersedia sampai akhir dari proyek. Kesalahan besar utama, jika tidak terdeteksi sampai program kerja dikaji ulang, maka kesalahan itu akan dapat mengakibatkan program sama sekali tidak dapat digunakan. Menyiakan banyak sumber daya.

Meskipun terdapat kelemahan, namun model ini memberikan daftar lengkap aktivitas minimal yang perlu terdapat di model pengembangan perangkat lunak. Umumnya, model-model pengembangan yang lain merupakan superset dari model pengembangan model *waterfall*, memuat aktivitas-aktivitas yang lebih banyak dari pada model *waterfall* ini.

2. Prototyping Model

Seringkali pemesan mendefinisikan kumpulan sasaran untuk diakomodasikan oleh perangkat lunak tanpa dapat mengidentifikasi rincian-rincian masukan, pengolahan dan keluaran yang diharapkannya. Model prototype cocok untuk situasi ini.

Pada model ini dimulai dengan pengumpulan informasi mengenai kebutuhan, dimana :

a. Pengembang dan pemesan bertemu dan mendefinisikan sasaran-sasaran umum.

b. Mengidentifikasi kebutuhan yang telah diketahui.

c. Mencari bidang-bidang yang masih memerlukan pendefinisian.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

Setelah itu pengembang melakukan "perancangan kilat" terhadap kebutuhan yang telah teridentifikasi pada pertemuan. Perancangan berfokus pada representasi yang tampak oleh pengguna. Perancangan ini menuntun pembangunan prototype perangkat lunak yang akan diberikan kepada pemakai atau pembeli. Prototype itu dievaluasi oleh pemakai dan digunakan sebagai landasan untuk memperbaiki spesifikasi kebutuhan. Proses ini berulang sampai prototype yang dikembangkan memenuhi kebutuhan pemakai, dan pengembang telah memahami permasalahan dengan lebih baik.

3. Spiral Model

Model ini diusulkan oleh Boehm. Model ini menggabungkan antara sifat alami iterasi dari *prototyping* dengan aspek sistematik dan terkendali dari *linear sequential* model. Model ini memberi peluang untuk pengembangan cepat.

Model pengembangan *spiral* adalah model pengembangan perangkat lunak yang memuat seluruh aktivitas di model *waterfall*, ditambah antara lain misalnya analisis resiko (tidak ada di model *waterfall*).

4. *Rapid Application Development* (RAD) *Model*

Rapid Application Development (RAD) *Model* adalah proses pengembangan perangkat lunak yang semakin meningkat (*incremental*) yang menekankan pada siklus pengembangan yang cepat. Model RAD merupakan adaptasi "berkecepatan tinggi" dari *linear sequential* model dimana pengembangan yang cepat dapat diperoleh dengan menggunakan pembangunan berbasis pada komponen.

RAD harus didukung oleh perangkat dan lingkungan pengembangan yang memadai, biasanya dikembangkan berdasarkan orientasi komponen. Lingkungan

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

pengembangan telah memiliki pustaka yang luar biasa besar dan lengkap seperti *Java Development Kit* beserta seluruh pustaka dari sun dan vendor-vendor pendukung lainnya.

Pemrograman biasanya berupa pembuatan antarmuka dan kode perekat untuk merekatkan beragam komponen-komponen yang telah terdapat di lingkungan pemrograman itu.

2.4.1 Analisis dan Desain Berorientasi Objek

Analisis beririentasi objek (OOA) menurut A. SUHENDAR [10] [2002, hal 11] merupakan metode analisis yang memeriksa *requirements* (syarat atau keperluan yang harus dipenuhi suatu sistem) dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup permasalahan.

Tujuan OOA adalah menentukan semua kelas (dan hubungan serta tingkah laku yang berkaitan dengannya) yang relevan dengan masalah yang akan dipecahkan. Sasaran OOA adalah mengembangkan sederetan model yang menggambarkan perangkat lunak komputer pada saat perangkat itu bekerja untuk memenuhi serangkaian persyaratan yang ditentukan oleh pelanggan.

Popularitas teknologi objek telah membengkitkan puluhan metode OOA. Masing-masing metode memperkenalkan proses untuk analisis produk atau sistem, serangkaian model yang berkembang keluar dari proses itu, dan notasi yang memungkinkan perekayasa perangkat lunak menciptakan masing-masing model dengan cara yang konsisten.

Desain berorientasi objek (OOD) adalah metode untuk mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek-objek sistem atau subsistem. Desain berorientasi objek (OOD) mentransformasi model analisis

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

yang dibuat dengan menggunakan OOA ke dalam suatu model desain yang berfungsi sebagai catak biru bangunan perangkat lunak. OOD menghasilkan desain yang mencapai sejumlah atau tingkat yang berbeda dari modularitas. Komponen sistem mayor dikumpulkan di dalam "modul" tingkat sistem yang disebut subsistem.

Kadang-kadang sulit untuk membuat perbedaan yang jelas di antara OOA dan OOD. Secara mendasar, OOA merupakan aktivitas klasifikasi di mana masalah dianalisis untuk menentukan kelas-kelas objek yang akan dapat diaplikasikan pada saat suatu solusi dikembangkan. Analisis juga menentukan hubungan dan tingkah laku objek. OOD memungkinkan perekayasa perangkat lunak untuk menunjukkan objek yang ditarik dari masing-masing kelas dan bagaimana objek-objek itu saling berhubungan satu dengan lainnya. OOD juga menggambarkan bagaimana hubungan di antara objek dicapai, bagaimana tingkah laku diimplementasi dan bagaimana komunikasi diimplementasi di antara objek.

Begitu model analisis yang sangat lengkap dikembangkan, perekayasa perangkat lunak berkonsentrasi pada desain sistem. Hal ini dilakukan dengan menggambarkan karakteristik subsistem yang diperlukan untuk mengimplementasi baik persyaratan pelanggan maupun lingkungan yang mendukung yang diperlukan bagi realisasinya.

Object Oriented Analisys & Design (OOAD) dapat didefinisikan sebagai pendekatan yang menekankan pada solusi *logic* berbasis objek.

Pemrograman *structural* seringkali tidak dapat mendesain apa yang akan terjadi dalam sistem yang telah selesai tanpa mengimplementasikan sistem terlebih dahulu. Jika ditemukan kesalahan dalam desain, maka desain harus

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

disusun ulang dari awal sampai akhir. Hal ini akan menjadi pemborosan biaya dan waktu. Perbedaan antara metode *structural* dan OOAD terletak pada bagaimana data dan fungsi disimpan. Dalam metode *structural*, data dan fungsi disimpan terpisah. Biasanya semua data ditempatkan sebelum fungsi ditulis. Seringkali, tidak terpikirkan sebelumnya untuk mengetahui data mana yang digunakan dalam suatu fungsi tertentu. Tetapi dalam OOAD, data dan fungsi yang berhubungan dalam suatu objek disimpan bersama-sama dalam satu kesatuan.

Keuntungan menggunakan OOAD bagi *programmer* adalah karena *programmer* dapat mendesain program dalam bentuk objek-objek dan hubungan antarobjek tersebut untuk kemudian dimodelkan dalam sistem nyata. Keuntungan lainnya adalah proses pembuatan *software* dapat dilakukan dengan lebih cepat karena *software* dibangun dari objek-objek standar, dapat menggunakan ulang model yang ada, dan dapat membuat model dengan cepat melalui metodologi.

2.5 Pengenalan UML

UML (*Unified Modeling Language*) menurut SOFTWARE ENGINEERING LABORATORY [9] [2004, hal 1] merupakan pengganti dari metode analisis berorientasi *object* dan *design* berorientasi *object* (OOA&D) yang dimunculkan sekitar akhir tahun 80-an dan awal tahun 90-an.

UML merupakan gabungan dari metode Booch, Rumbaugh (OMT) dan Jacobson. Tetapi UML ini akan mencakup lebih luas daripada OOA&D. Pada pertengahan pengembangan UML dilakukan standarisasi proses dengan OMG (*Object Management Group*) dengan harapan UML akan menjadi bahasa standar pemodelan pada masa yang akan datang.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed
after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

UML disebut sebagai bahasa pemodelan bukan metode. Kebanyakan metode terdiri paling sedikit prinsip, bahasa pemodelan dan proses. Bahasa pemodelan (sebagian besar grafik) merupakan notasi dari metode yang digunakan untuk mendesain secara cepat.

Bahasa pemodelan merupakan bagian terpenting dari metode. Ini merupakan bagian kunci tertentu untuk komunikasi. Jika ingin berdiskusi tentang desain dengan seseorang, maka dibutuhkan bahasa pemodelan bukan proses yang digunakan untuk mendapatkan desain.

UML merupakan bahasa standar untuk penulisan *blueprint software* yang digunakan untuk visualisasi, spesifikasi, pembentukan dan pendokumentasian alat-alat dari sistem perangkat lunak.

2.5.1 Pengertian Unified Modeling Language (UML)

UML adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan *artifacts* (bagian dari informasi yang digunakan atau dihasilkan oleh proses pembuatan perangkat lunak, *artifact* tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak. Selain itu, UML adalah bahasa pemodelan yang menggunakan konsep orientasi *object*. UML dibuat oleh Grady Booch, James Rumbaugh, dan Ivar Jacobson dibawah bendera *Rational Software Corp*. UML menyediakan notasi-notasi yang membantu memodelkan sistem dari berbagai perspektif. UML tidak hanya digunakan dalam pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

2.5.2 Bagian-bagian UML

Bagian-bagian utama dari UML adalah *view*, diagram, model element, dan *general mechanism*. Dari keempat bagian utama, penulis hanya menjelaskan dua bagian utama saja.

1. *View*

   View digunakan untuk melihat sistem yang dimodelkan dari beberapa aspek yang berbeda. *View* bukan melihat grafik, tapi merupakan suatu abstraksi yang berisi sejumlah diagram.

   Beberapa jenis *view* dalam UML antara lain: *use case view*, *logical view*, *component view*, *concurrency view*, dan *deployment view*.

   a. *Use case View*

      Mendeskripsikan fungsionalitas sistem yang seharusnya dilakukan sesuai yang diinginkan eksternal *actors*. *Actor* yang berinteraksi dengan sistem dapat berupa user atau sistem lainnya.

      View ini digambarkan dalam *use case diagrams* dan kadang-kadang dengan *activity diagrams*. *View* ini digunakan terutama untuk pelanggan, perancang (*designer*), pengembang (*developer*), dan penguji sistem (*tester*).

   b. *Logical View*

      Mendeskripsikan bagaimana fungsionalitas dari sistem, struktur statis (*class, object*, dan *relationship*) dan kolaborasi dinamis yang terjadi ketika *object* mengirim pesan ke *object* lain dalam suatu fungsi tertentu.

      View ini digambarkan dalam *class diagrams* untuk struktur statis dan dalam *state, sequence, collaboration*, dan *activity diagram* untuk model

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

dinamisnya. *View* ini digunakan untuk perancang (*designer*) dan pengembang (*developer*).

c. *Component View*

Mendeskripsikan implementasi dan ketergantungan modul. Komponen yang merupakan tipe lainnya dari *code module* diperlihatkan dengan struktur dan ketergantungannya juga alokasi sumber daya komponen dan informasi administratif lainnya. *View* ini digambarkan dalam *component view* dan digunakan untuk pengembang (*developer*).

d. *Concurrency View*

Membagi sistem kedalam proses dan prosesor. *View* ini digambarkan dalam diagram dinamis (*state, sequence, collaboration*, dan *activity diagrams*) dan diagram implementasi (*component* dan *deployment diagrams*) serta digunakan untuk pengembang (*developer*), pengintegrasi (*integrator*), dan penguji (*tester*).

e. *Deployment View*

Mendeskripsikan fisik dari sistem seperti komputer dan perangkat (*nodes*) dan bagaimana hubungannya dengan lainnya. *View* ini digambarkan dalam *deployment diagrams* dan digunakan untuk pengembang (*developer*), pengintegrasian (*integrator*), dan penguji (*tester*).

2. Diagram

Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu *view* tertentu dan ketika digambarkan biasanya dialokasikan untuk *view* tertentu. Adapun jenis diagram antara lain :

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

a. *Use case Diagram*

Menggambarkan sejumlah eksternal *actors* dan hubungannya ke *use case* yang diberikan oleh sistem. *Use case* adalah deskripsi fungsi yang disediakan oleh sistem dalam bentuk teks sebagai dokumentasi dari *use case symbol* namun dapat juga dilakukan dalam *activity diagrams*.

Use case digambarkan hanya yang dilihat dari luar oleh *actor* (keadaan lingkungan sistem yang dilihat user) dan bukan bagaimana fungsi yang ada di dalam sistem.

b. *Class Diagram*

Menggambarkan struktur statis *class* di dalam sistem. *Class* merepresentasikan sesuatu yang ditangani oleh sistem. *Class* dapat berhubungan dengan yang lain melalui berbagai cara : *associated* (terhubung satu sama lain), *dependent* (satu *class* tergantung/ menggunakan *class* yang lain), *specialed* (satu *class* merupakan spesialisasi dari *class* lainnya), atau *package* (grup bersama sebagai satu unit). Sebuah sistem biasanya mempunyai beberapa *class diagram*.

c. *State Diagram*

Menggambarkan semua *state* (kondisi) yang dimiliki oleh suatu objek dari suatu *class* dan keadaan yang menyebabkan *state* berubah. Kejadian dapat berupa objek lain yang mengirim pesan. *State class* tidak digambarkan untuk semua *class*, hanya yang mempunyai sejumlah *state* yang terdefinisi dengan baik dan kondisi *class* berubah oleh *state* yang berbeda.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

d. *Sequence Diagram*

Menggambarkan kolaborasi dinamis antara sejumlah objek. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antara objek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

e. *Collaboration Diagram*

Menggambarkan kolaborasi dinamis seperti *sequence diagram*. Dalam menunjukkan pertukaran pesan, *collaboration diagram* menggambarkan objek dan hubungannya (mengacu ke konteks). Jika penekanannya pada waktu atau urutan gunakan *sequence diagrams*, tapi penekanannya pada konteks gunakan *collaboration diagram*.

f. *Activity Diagram*

Menggambarkan rangkaian aliran dari aktifitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi.

g. *Component Diagram*

Menggambarkan stuktur fisik kode dari komponen. Komponen dapat berupa *source code*, komponen biner, atau *executable component*. Sebuah komponen berisi informasi tentang *logic class* atau *class* yang diimplementasikan sehingga membuat pemetaan dari *logical view* ke *component view.*

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

h. Deployment Diagram

Menggambarkan arsitektur fisik dari perangkat keras dan perangkat lunak sistem, menunjukkan hubungan komputer dengan perangkat (*nodes*) satu sama lain dan jenis hubungannya. Di dalam *nodes, executable component* dan objek yang dialokasikan untuk memperlihatkan unit perangkat lunak yang dieksekusi oleh *node* tertentu dan ketergantungan komponen.

2.6 Basis Data (Database)

Basis data (*database*) merupakan salah satu komponen yang paling penting di dalam sistem informasi karena merupakan dasar dalam menyediakan informasi bagi pemakai. *Database* terdiri dari berbagai macam data yang tersusun sehingga pemakai dapat dengan mudah untuk menambah, mengurangi, menghapus ataupun mengambil data yang dibutuhkan dalam sistem, sehingga informasi yang nantinya akan dihasilkan dapat terpenuhi dengan baik.

Dalam *database* terdiri dari entity, atribut, data value, record, dan file.

1. Entity

   Entity adalah objek yang diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks yang dirancang.

2. Atribut

   Setiap entity mempunyai atribut atau sebutan untuk mewakili suatu entity.

3. Data Value

   Data value adalah data aktual atau informasi yang disimpan pada tiap data elemen yang saling berkaitan menginformasikan tentang suatu entity secara lengkap.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

4. Record

   Record adalah kumpulan elemen-elemen yang saling berkaitan menginformasikan mengenai suatu entity secara lengkap.

5. File

   File merupakan kumpulan record-record sejenis yang mempunyai panjang elemen yang sama, record yang sama namun berbeda data value dan saling berhubungan. File dibuat bila diperlukan suatu kumpulan record tertentu. Data dapat dikumpulkan dari sumber-sumber dokumen yang baru atau mengambil record dari file yang sudah ada dan disusun kembali ke dalam file baru.

2.7 Arsitektur Aplikasi

2.7.1 Pengertian Jaringan Komputer

Jaringan komputer adalah sekumpulan komputer pribadi yang saling berhubungan antara satu dengan yang lainnya menggunakan *protocol* komunikasi sehingga dapat saling berbagi informasi, program penggunaan bersama perangkat keras seperti printer, *harddisk*. Selain itu jaringan komputer dapat diartikan sebagai kumpulan sejumlah terminal komunikasi yang berada di berbagai lokasi yang terdiri dari lebih dari satu komputer yang saling berhubungan. Dalam sebuah jaringan komputer biasanya terhubung sejumlah komputer ke sabuah/beberapa server.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

2.7.2 Jenis-jenis Jaringan Komputer

Jenis-jenis jaringan ada dua yaitu :

1 Model *peer to peer*

Menurut model ini, setiap *host* dapat menawarkan layanan ke *peer* lain dan juga mengambil layanan dari *peer* lain. Model ini cocok untuk jaringan kecil.

2 model *Client/server*

model ini memisahkan secara jelas, mana yang dapat memberikan layanan jaringan (*server*) dan mana yang hanya memberikan layanan (*client*). Beberapa komputer di*set-up* sebagai *server* yang memberikan sumber daya (*resource*) dari jaringan : printer, modem, dan saluran lainnya kepada komputer lain yang dikoneksi ke jaringan yang berfungsi sebagai *client*.

2.7.3 Topologi Jaringan Komputer

Yang dimaksud topologi jaringan itu adalah susunan fisik bagaimana *node-node* saling dihubungan. Ada tiga topologi jaringan komputer, yaitu :

1. Topologi Bus

Ethernet adalah contoh yang menggunakan topologi bus yang menggunakan satu kawat (kabel) yang berfungsi sebagai medium untuk mentransmisikan data. *Node* yang merupakan bagian dari jaringan dihubungkan seluruhnya ke kabel tersebut. *Node-node* yang terhubungkan mengirim dan menerima data jaringan malalui kabel sebagai pembawa sinyal dan melihat apakah data tersebut ditujukan buat dirinya.

2. Topologi Ring

Salah satu *node* dihubungkan dengan *node* yang ada di depan dan di belakangnya sehingga membentuk ring. Setiap *node* mendapatkan giliran

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

menggunakan jaringan dengan mengirim "*token*". *Node* yang mendapat giliran dapat mengirim data dan *node* lain menerima data, serta melihat apakah data ditujukan kepadanya. Bila ditujukan buat dirinya, datapun disimpan, bila tidak data diteruskan ke *node* di depannya.

3. Topologi Star atau Hub

   Susunan atau skema dari topologi ini mirip sebuah bintang. Topologi ini memiliki satu hub pusat dari mana data ditransmisikan ke seluruh *node* dalam jaringan. Skema ini mempunyai kelebihan dibandingkan dua skema sebelumnya, yaitu bila terjadi kerusakan pada kabel tidak membawa dampak bagi seluruh *node*, tapi hanya *node* yang bersangkutan saja sehingga aktivitas jaringan tidak terganggu secara total. Ini berbeda dengan skema bus dan ring, dimana kerusakan pada kabel berakibat pada seluruh jaringan.

2.7.4 Manfaat Jaringan Komputer

Manfaat jaringan komputer adalah sebagai berikut :

1. *Sharing Resources*

   Sharing Resources bertujuan agar seluruh program, peralatan/*peripheral* lainnya dapat dimanfaatkan oleh setiap orang yang ada pada jaringan komputer tanpa terpengaruh oleh lokasi maupun pengaruh dari pemakai. Dengan kata lain, seorang pemakai yang letaknya jauh sekalipun dapat memanfaatkan data maupun informasi yang lainnya tanpa mengalami kesulitan.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

2. Media Komunikasi

   Jaringan komputer memungkinkan terjadinya komunikasi antara pengguna, baik untuk *teleconference* maupun untuk mengirim pesan/informasi yang penting lainnya.

3. Integrasi Data

   Pembangunan jaringan komputer dapat mencegah ketergantungan pada komputer pusat. Setiap proses data tidak harus dilakukan satu komputer saja, melainkan dapat didistribusikan ke tempat lainnya oleh sebab inilah maka dapat terbentuk data yang terintegarasi sehingga dengan demikian memudahkan pemakai untuk memperoleh dan mengolah informasi setiap saat.

4. Pengembangan dan Pemeliharaan

   Dengan adanya jaringan komputer ini, maka perkembangan peralatan dapat dilakukan dengan mudah dan menghemat biaya. Jaringan komputer bisa memudahkan pemakai dalam merawat *harddisk* dan peralatan lainnya. Misalnya untuk memberikan perlindungan serangan virus maka pemakai cukup memusatkan perhatian pada *harddisk* yang ada di dalam komputer pusat.

5. Keamanan Data

   Sistem jaringan komputer memberikan perlindungan terhadap data jaminan keamanan data tersebut di berikan melalui pengaturan hak akses para pemakai dan *password*, serta teknik perlindungan terhadap *harddisk* sehingga data mendapatkan perlindungan yang efektif.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

6. Sumber Daya Lebih Efisien dan Informasi Terkini

   Dengan adanya pemakaian sumber daya secara bersama-sama maka pemakai bisa mendapatkan hasil dengan maksimal dan kualitas yang tinggi. Selain itu data atau informasi yang di akses selalu terbaru, karena setiap ada perubahan yang terjadi dapat secara langsung diketahui oleh setiap pemakai.

TCP/IP (*Transmission Control Protocol/Internet Protocol*) merupakan protokol standar internet yang digunakan untuk melakukan koneksi ke internet protokol. TCP/IP memiliki beberapa subyek protokol yang berbeda yang beroperasi pada lapisan yang berbeda dan mempunyai tugas masing-masing. Berkat adanya protokol ini setiap komputer dapat berhubungan secara fleksibel dengan *host-host* yang terkoneksi.

2.8 Pengertian Client/Server

Sistem *client/server* berjalan setidaknya pada dua sistem komputer yang berbeda, satu sebagai *server* dan satunya lagi sebagai *client*. *Client* umumnya berupa sebuah komputer *desktop* yang terhubung dalam suatu jaringan dimana biasanya komputer *client* akan mengeluarkan suatu permintaan kepada komputer *server*, dan *server* kemudian akan menjalankan permintaan tersebut dan mengirimkan kembali suatu informasi kepada *client* tersebut. Komputer pusat berfungsi sebagai penyelia data dan program disebut *server*. Sedangkan komputer-komputer mikro yang berfungsi sebagai terminal disebut *client*.

Maka dari itu komputer *server* setidaknya memiliki kemampuan yang tinggi untuk melayani keinginan komputer *client* dan biasanya pada suatu jaringan

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

komputer *client* tidak hanya satu. Dan sistem jaringan ini disebut dengan sistem *client/server*.

2.9 Perangkat Lunak Pendukung

2.9.1 Sekilas Tentang Visual Basic 6.0

Microsoft Visual Basic 6.0 adalah bahasa pemrograman yang bekerja dengan lingkup *microsoft windows.* M.S Visual Basic 6.0 dapat memanfaatkan kemampuan M.S *windows* secara optimal. Kemampuannya dapat dipakai untuk merancang program aplikasi yang berpenampilan seperti program aplikasi lainnya berbasis M.S *windows.*

Visual Basic merupakan *event-driven programming* (pemrograman terkendali kejadian) artinya program menunggu sampai adanya respon dari pemakai berupa *event*/kejadian tertentu (tombol, diklik, menu dipilih, dan lain-lain). Ketika *event* terdeteksi, kode yang berhubungan dengan *event* (prosedur *event*) akan dijalankan.

Visual Basic berawal dari sebuah ide untuk membuat *simple scripting language* untuk GUI *Windows*. Saat itu masih *Windows* 3.1.1. sebagai bahasa pemrograman yang tergolong sebagai RAD (*Rapid Application Development*) *tool*, perkembangan visual basic mendapat sambutan yang sangat baik dikalangan *programmer*.

Ledakan jumlah *developer* atau *programmer* yang sangat berarti terjadi saat munculnya Visual Basic 5, dimana Visual Basic mampu melakukan dobrakan yang sangat besar, yaitu kemampuan membuat *Active Control* beserta komponennya (*Active DLL* dan *Active EXE*). Hal ini mampu menyedot perhatian

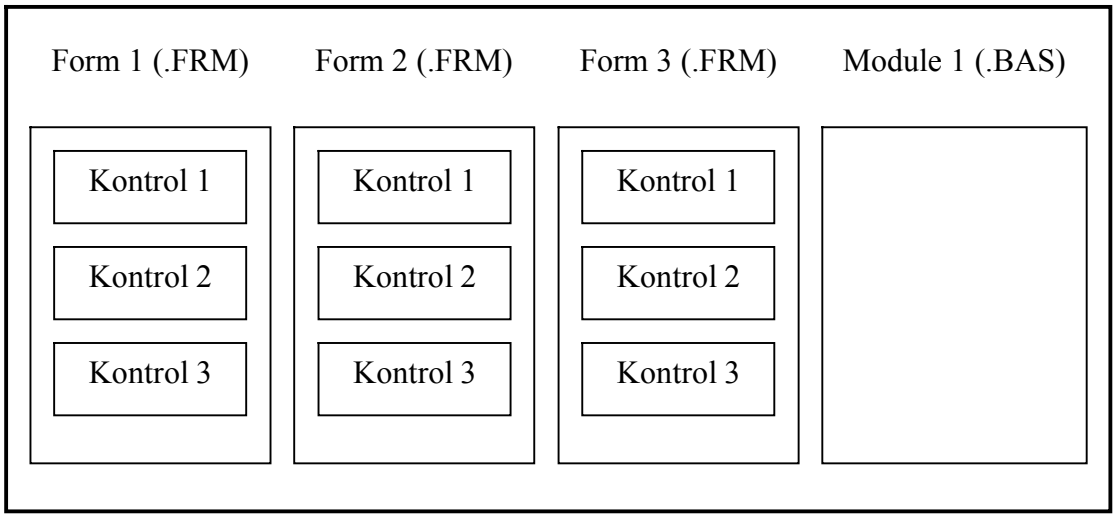
This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

developer dan mempengaruhi perubahan teknik pemrograman Visual Basic, terutama pada konsep OOP (*Object Oriented Programming*).

Pada oktober 1997 muncul Visual Basic 6.0 yang tidak jauh berbeda dengan Visual Basic 5. hanya saja, Visual Basic 6.0 sudah ditambah dengan kemampuan internet, programming, dan juga penyempurnaan pada tingkat keamanan saat menjalankan *thread*, *COM-based application*, yang kemudian membuat trend pemograman Visual Basic 6.0 mulai bergeser dari aplikasi *desktop* ke arah aplikasi *network* dan desain komponen. Hal itu terlihat dengan hadirnya *template* IIS *application project* dan DHTML *project* pada pilihan menu jenis *project*.

2.9.1.1 Struktur Aplikasi Visual Basic 6.0

Proyek (.VBP, .MAK)



Gambar 2.1 Struktur Aplikasi Visual Basic

[Sumber : Kusumo, A.S, 2000, "Buku Latihan Microsoft Visual Basic 6.0", hal: 2, PT. Elex Media Komputindo, Jakarta]

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

Aplikasi (proyek) terdiri dari :

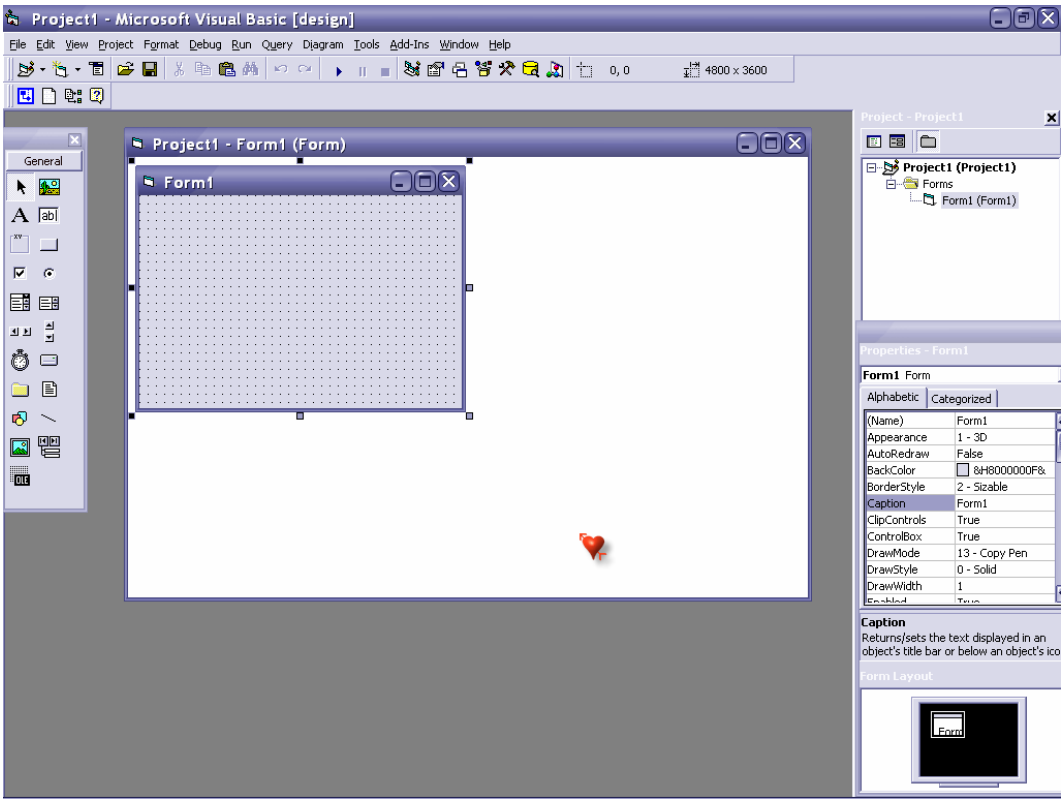
1. Form – *Windows*/jendela dimana anda akan membuat *user interface*/tampilan.
2. Kontrol/*Control* – Tampilan berbasis grafis yang dimasukkan pada form untuk membuat interaksi dengan pemakai (*text box, label, scroll bar*, tombol *command*).

   Form dan kontrol adalah objek.
3. Properti/*Properties* – Nilai/karakteristik yang dimiliki oleh sebuah objek Visual Basic. Contoh: *Name, Caption, Size, Color, Position* dan *Text*. Visual Basic menerapkan properti *default*/standar.
4. Metode/*Methods* – Serangkaian perintah yang sudah tersedia pada suatu objek yang dapat diminta untuk mengerjakan tugas khusus.
5. Prosedur Kejadian/*Event Procedures* – Kode yang berhubungan dengan suatu objek. Kode ini akan dieksekusi ketika ada respon dari pemakai berupa *event* tertentu.
6. Prosedur Umum/*General Procedures* – Kode yang tak berhubungan dengan suatu objek. Kode ini harus diminta oleh aplikasi.
7. Modul/*Module* – Kumpulan dari prosedur umum, deklarasi variabel dan definisi konstanta yang digunakan oleh aplikasi.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

2.9.1.2 Tampilan Layar Visual Basic

Tampilan layar Visual Basic seluruhnya seperti dibawah ini :

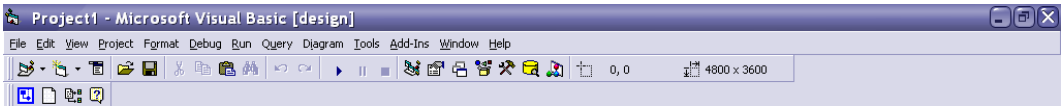


Gambar 2.2 Tampilan layar Visual Basic

[Sumber : Kusumo, A.S, 2000, "Buku Latihan Microsoft Visual Basic 6.0", hal : 3, PT. Elex Media Komputindo, Jakarta]

1. *Main Windows* (Jendela Utama) terdiri *title bar* (baris judul), *menu bar*, dan *toolbar*. Baris judul berisi nama proyek, mode operasi Visual Basic sekarang dan form yang aktif. *Menu bar* merupakan menu *drop-down* di mana dapat mengontrol operasi dari lingkungan Visual Basic. *Toolbar* 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.

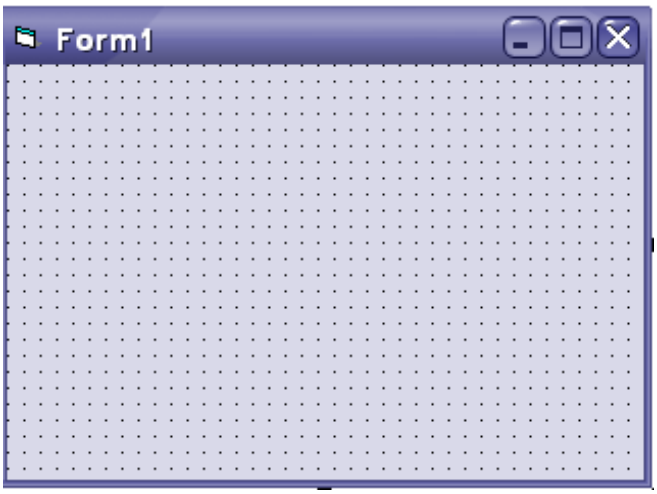
This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.



Gambar 2.3 Jendela utama Visual Basic

[Sumber : Kusumo, A.S, 2000, "Buku Latihan Microsoft Visual Basic 6.0", hal : 3, PT. Elex Media Komputindo, Jakarta]

2. Form *Windows* (Jendela Form) adalah pusat dari pengembangan aplikasi Visual Basic. Di sinilah tempat "menggambar" aplikasi.

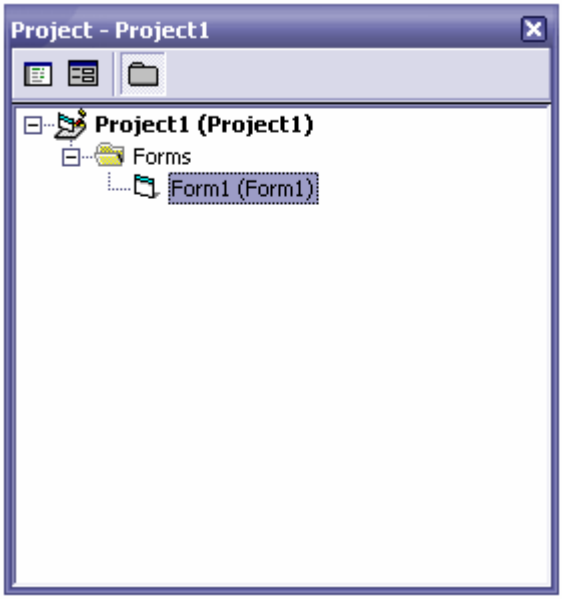


Gambar 2.4 Jendela Form Visual Basic

[Sumber : Kusumo, A.S, 2000, "Buku Latihan Microsoft Visual Basic 6.0", hal : 4, PT. Elex Media Komputindo, Jakarta]

3. *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.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.



Gambar 2.5 Project Explorer Visual Basic

[Sumber : Kusumo, A.S, 2000, "Buku Latihan Microsoft Visual Basic 6.0", hal : 4, PT. Elex Media Komputindo, Jakarta]

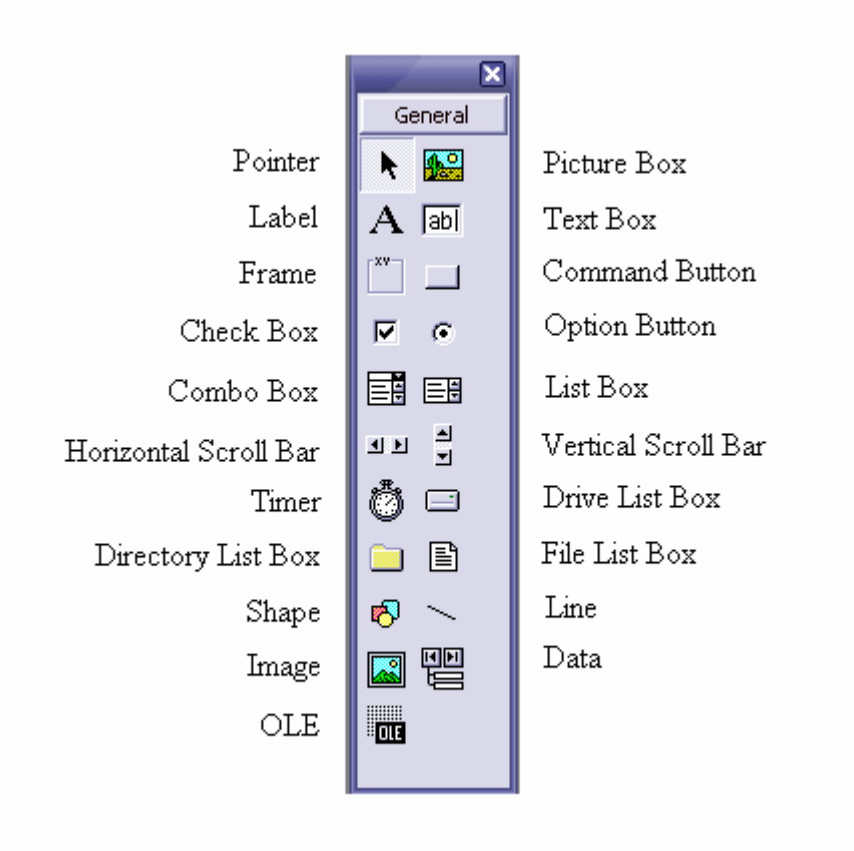
4. *Toolbox* adalah kumpulan dari objek yang digunakan untuk membuat *user interface* serta kontrol bagi program aplikasi.

   Ada dua cara untuk menempatkan kontrol pada suatu form:

   a. Klik ganda kontrol dalam *toolbox*. Selanjutnya anda dapat mengubah besar dan ukurannya serta memindahkannya dengan metode *drag and drop*.

   b. Klik kontrol dalam *toolbox*, kemudian pindahkan *pointer mouse* pada jendela form. Kursor berubah menjadi *crosshair*. Tempatkan *crosshair* pada sudut kiri atas di mana kontrol diletakkan sesuai keinginan, tekan tombol *mouse* kiri dan tahan ketika menyeret (*drag*) kursor ke arah kanan bawah.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

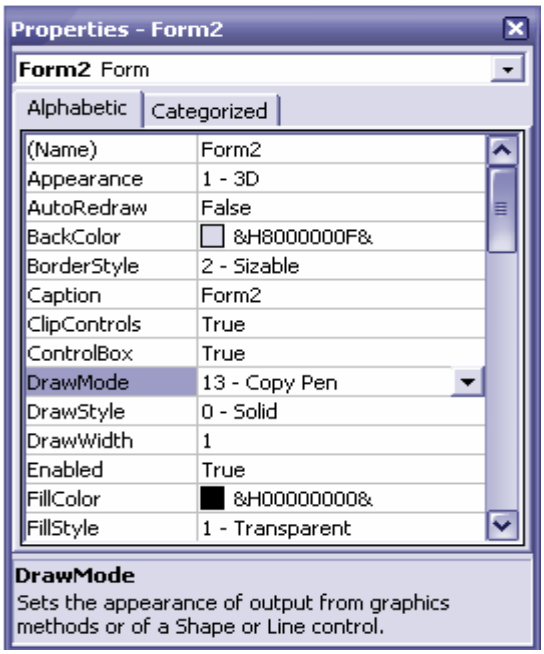


Gambar 2.6 Toolbox Visual Basic

[Sumber : Kusumo, A.S, 2000, "Buku Latihan Microsoft Visual Basic 6.0", hal : 5, PT. Elex Media Komputindo, Jakarta]

c. *Properties Windows* (Jendela Properti) berisi daftar struktur *setting property* 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 property dari objek terpilih.

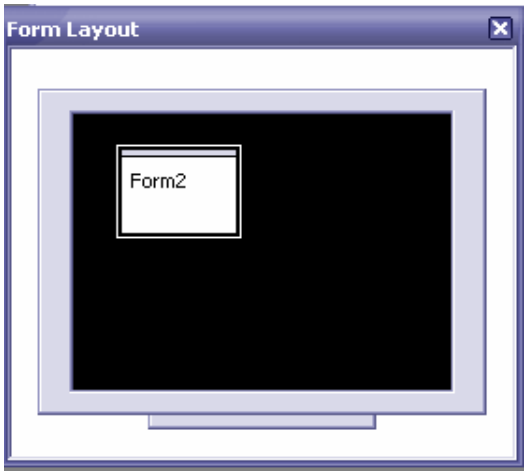
This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.



Gambar 2.7 Jendela Properti Visual Basic

[Sumber : Kusumo, A.S, 2000, "Buku Latihan Microsoft Visual Basic 6.0", hal : 6, PT. Elex Media Komputindo, Jakarta]

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



Gambar 2.8 Jendela Layout Form Visual Basic

[Sumber : Kusumo, A.S, 2000, "Buku Latihan Microsoft Visual Basic 6.0", hal : 6, PT. Elex Media Komputindo, Jakarta]

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

2.9.2 Sekilas Tentang MySQL

MySQL adalah multiuser database yang menggunakan bahasa *Structured Query Language* (SQL). MySQL dalam operasi *client-server* melibatkan *server daemon* MySQL disisi server dan berbagai macam program serta *library* yang berjalan disisi *client*.

SQL singkatan dari *Structured Query Language*. SQL sering dibaca sebagai "*sequel*" adalah bahasa terstruktur yang digunakan untuk query, mengupdate dan mengolah relasi antar database. User dapat menggunakan bahasa ini untuk memuat, mengurutkan, dan menyaring suatu data sehingga dihasilkan suatu data yang spesifik dari suatu *database*.

2.10 Pengertian Penjualan

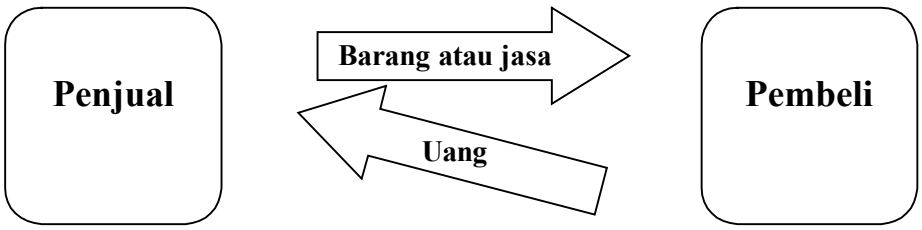
Menurut BASU SWASTHA [13] [1999, hal 8] Penjualan merupakan salah satu fungsi dari pemasaran atau merupakan bagian dari kegiatan pemasaran. Penjualan sangat penting dan menentukan, karena suatu perusahaan untuk dapat memasarkan produknya, yaitu dengan cara menjual produk tersebut, dan apabila penjualan tidak dapat dilaksanakan maka fungsi-fungsi lain dari pemasaran tidak akan berjalan.

Penjualan adalah proses terjadinya berkurangnya barang di gudang dengan memperhitungkan keuntungan dan kerugian yang akan diterima oleh penjual.

Sedangkan menjual adalah ilmu dan seni mempengaruhi pribadi yang dilakukan oleh penjual untuk mengajak orang lain agar bersedia membeli barang atau jasa yang ditawarkannya.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

Jadi, adanya penjualan dapat tercipta suatu proses pertukaran barang atau jasa antara penjual dengan pembeli. Di dalam perekonomian kita (ekonomi uang), seseorang yang menjual sesuatu akan mendapatkan imbalan berupa uang. Dengan alat penukar berupa uang, orang akan lebih mudah melakukan jual-beli. Jarang yang jauh tidak menjadi masalah bagi penjual. Secara sederhana, transaksi penjualan yang dilakukan oleh penjual dan pembeli dapat dilihat sebagai proses pertukaran.



Gambar 2.9 Proses Pertukaran Jual-Beli

(Sumber : Swastha, B, 1999, "Manajemen Penjualan Edisi 3", hal : 9, BPFE, Yogyakarta)

Semakin pandai seseorang untuk menjual akan semakin cepat pula mencapai sukses dalam melaksanakan tugas-tugasnya sehingga tujuan yang diinginkan akan segera terlaksana. Dalam segala bidang dan tingkatan, taktik penjualan harus digunakan agar pelayanan yang diberikan kepada orang lain dapat memberikan kepuasan. Dengan kepuasan ini diharapkan mereka dapat menjadi langganan atau sahabat yang baik.

Dalam melakukan penjualannya, penjual dituntut untuk memiliki bakat seni serta keahlian untuk mempengaruhi orang lain. Bakat inilah yang sering tidak dimiliki oleh setiap orang. Tidaklah mudah untuk mengarahkan kemauan calon pembeli dengan cara mengemukakan berbagai alasan serta pendapatnya.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

Ada definisi lain tentang penjualan yang dikemukakan oleh William G. Nickels yang menyebutnya dengan istilah penjualan tatap muka (*Personal Selling*).

Penjualan tatap muka adalah interaksi antar individu, saling bertemu muka yang ditunjukkan oleh menciptakan, memperbaiki, menguasai atau mempertahankan hubungan pertukaran yang saling menguntungkan dengan pihak lain.

Jadi, penjualan tatap muka merupakan komunikasi orang secara individual yang dapat dilakukan untuk mencapai tujuan seluruh usaha pemasaran pada umumnya, yaitu meningkatkan penjualan yang dapat menghasilkan laba dengan menawarkan kebutuhan yang memuaskan kepada pasar dalam jangka panjang.

Jenis-jenis penjualan :

1. *Trade Selling*

   Trade selling dapat terjadi bilamana produsen dan pedagang besar mempersilakan pengecer untuk berusaha memperbaiki distributor produk-produk mereka. Hal ini melibatkan para penyalur dengan kegiatan promosi, peragaan, persediaan dan produk baru. Jadi titik beratnya adalah pada "penjualan melalui" penyalur dari pada "penjualan ke" pembeli akhir.

2. *Missionary Selling*

   Dalam *missionary selling*, penjualan berusaha ditingkatkan dengan mendorong pembeli untuk membeli barang-barang dari penyalur perusahaan.

3. *Technical Selling*

   Technical selling berusaha meningkatkan penjualan dengan pemberian saran dan nasehat kepada pembeli akhir dari barang dan jasanya.

This document was created with free TRIAL version of eXPert PDF.This watermark will be removed after purchasing the licensed full version of eXPert PDF. Please visit www.bvrp.com for more details.

4. *New Business Selling*

   New Business selling berusaha membuka transaksi baru dengan merubah calon pembeli menjadi pembeli. Jenis penjualan ini sering dipakai oleh perusahaan asuransi.

5. *Responsive Selling*

   Setiap tenaga penjualan diharapkan dapat memberikan reaksi terhadap permintaan pembeli. Dua jenis penjualan utama di sini adalah *route driving* dan *retailing*. Jenis penjualan seperti ini tidak akan menciptakan penjualan yang terlalu besar meskipun layanan yang baik dan hubungan pelanggan yang menyenangkan dapat menjurus kepada pembelian ulang.

2.11 Pengertian Pembelian

Menurut SAEFUDDIN AZWAR [1] [1998, hal 176] pembelian adalah proses terjadinya pengadaan barang yang tergolong kurang memenuhi kebutuhan yang diperlukan". Perusahaan yang mengolah bahan setengah jadi dimana proses pembelian harus dilakukan secara berkesinambungan agar perusahaan tersebut dapat bertahan dalam persaingan, maka proses pembelian dan pengadaan barang bahan baku untuk dapat terus berproduksi sesuai dengan permintaan.

Pembelian terbagi menjadi dua cara yaitu :

1. Pembelian Tunai

   Yaitu pembelian barang secara langsung kepada supplier atau pemasok barang dengan pembayaran secara langsung saat itu juga.

2. Pembelian Kredit

   Yaitu pembelian barang secara langsung kepada supplier atau pemasok barang dengan pembayaran tidak langsung, melainkan berangsur sesuai kesepakatan.