

BAB II

LANDASAN TEORI

II.A. Pengertian Akuntansi

Pada dasarnya, akuntansi adalah sistem informasi. Tepatnya, akuntansi adalah penerapan dari teori umum informasi untuk masalah-masalah operasi ekonomi yang efisien. Akuntansi juga merupakan bagian besar dari informasi umum yang dinyatakan dalam bentuk kuantitatif. Dalam konteks ini akuntansi merupakan bagian dari sistem informasi umum suatu kesatuan operasional dan juga merupakan bagian dari bidang besar di bawah nama konsep informasi (Wilkinson, 1993:15).

Akuntansi adalah suatu aktivitas jasa (mengidentifikasi, mengukur, mengklasifikasikan dan mengikhtisarkan) kejadian atau transaksi ekonomi yang menghasilkan informasi kuantitatif terutama yang bersifat keuangan yang digunakan dalam pengambilan keputusan (Amin. W, 1997).

Akuntansi merupakan proses pencatatan (*recording*), pengelompokan (*classifying*), perangkuman (*summarizing*), dan pelaporan (*reporting*) dari kegiatan transaksi perusahaan (Jogiyanto, 1997). Sehingga Jogiyanto menyatakan bahwa akuntansi itu sendiri

walaupun dilakukan secara manual tetap merupakan suatu sistem informasi.

Sementara Wing Wahyu Winarno mendefinisikan akuntansi sebagai proses pencatatan, pengolahan data transaksi keuangan, serta menyajikan informasi tersebut kepada pihak-pihak yang berhak dan berkepentingan sehingga dapat digunakan sebagai dasar pengambilan keputusan. (Wing Wahyu Winarno, 2006).

II.B. Pengertian Sistem

Wing Wahyu Winarno (2006) memberikan pengertian Sistem sebagai sekumpulan komponen yang saling bekerja sama yang bertujuan untuk mencapai tujuan tertentu.

Sedangkan menurut John F, Nash dan Martin B. Roberts, 1984, yang dikutip dari Jogiyanto 1997 sistem adalah suatu kumpulan komponen yang berinteraksi membentuk suatu kesatuan dan keutuhan yang kompleks di tingkat tertentu untuk mengejar tujuan yang umum.

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran yang tertentu (Jogiyanto, 2005).

II.C. Pengertian Informasi

Informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi si penerima dan mempunyai nilai yang nyata yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau keputusan-keputusan yang akan datang (Gordon, 2005).

Wing Wahyu Winarno (2006) menjelaskan bahwa informasi adalah sebuah data yang sudah diolah sehingga akan berguna dalam pembuatan keputusan.

Informasi merupakan sebuah data yang telah diletakkan dalam konteks yang lebih berarti dan berguna, data tersebut nantinya dapat dikomunikasikan kepada penerima untuk digunakan dalam pengambilan keputusan (John Burch dan Gary Grudnitski, 1986, dikutip dari Jogyanto 1997).

II.D. Pengertian Sistem Informasi

Jogyanto HM (2001:11) mengatakan bahwa Sistem Informasi adalah suatu sistem yang berada didalam suatu organisasi yang berguna untuk memenuhi kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi. Selain itu juga dapat menyediakan laporan-laporan yang diperlukan oleh pihak luar.

II.E. Pengertian Sistem Informasi Akuntansi

Kumpulan dari sumber-sumber modal yang berada didalam suatu organisasi dan memiliki tanggungjawab untuk menyediakan informasi keuangan dan informasi-informasi lain yang didapat dari pengumpulan dan pengolahan data transaksi (Barry E. Cushing yang dikutip oleh Jogiyanto (1997:49)).

Sedangkan Wing Wahyu Winarno (2006) berpendapat bahwa sekumpulan sistem yang berfungsi untuk mencatat data transaksi, mengolah data, dan menyajikan informasi akuntansi kepada pihak-pihak internal(manajemen perusahaan) dan pihak-pihak eksternal(supplier, konsumen, kreditur, pemerintah, dan lainnya) disebut sebagai Sistem Informasi Akuntansi (SIA). Beberapa kegiatan yang saling berurutan yang berguna untuk menangani suatu transaksi disebut prosedur. Beberapa prosedur berkumpul menjadi satu dan saling berkaitan antara satu prosedur dengan prosedur lain disebut siklus. Lalu beberapa siklus yang terjadi dapat membuat suatu sistem akuntansi.

II.F. Komponen Utama Sistem Informasi

II.F.1. Komponen-komponen Sistem Informasi

Beberapa blok-blok membentuk sebuah sistem informasi. Blok-blok tersebut antara lain (Wing Wahyu Winarno (2006)) :

II.F.1.1. Blok Masukan (*Input Block*)

Masukan (input) adalah seluruh data yang dimasukkan ke dalam sistem informasi, termasuk seluruh metode dan media yang dipakai untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

II.F.1.2. Blok Model / Proses (*Model Block*)

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

II.F.1.3. Blok Keluaran (*Output Block*)

Informasi berharga dan dokumen yang berguna yang dihasilkan dari sebuah sistem informasi. Hampir semua tingkatan manajemen dan para pengguna sistem membutuhkan informasi dan dokumen tersebut.

II.F.1.4. Blok Teknologi (*Technology Block*)

“*tool box*” dalam sebuah sistem informasi adalah teknologi. Teknologi ini dapat digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan output dan mengirimkan output. Blok teknologi terdiri dari tiga bagian, yaitu: Teknisi (*Brainware*), Perangkat Lunak (*Software*), dan Perangkat Keras (*Hardware*).

II.F.1.5. Blok Basis Data (*Database Block*)

Basis data (*Database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di perangkat keras komputer, dan menggunakan perangkat lunak untuk memanipulasinya.

II.F.1.6. Blok Kendali (*Control Block*)

Blok kendali berguna untuk mengatasi beberapa hal yang mungkin dapat merusak sistem informasi. Hal-hal tersebut terkadang dapat terjadi dan tidak bisa diprediksi seperti gempa bumi, banjir, sabotase, kebakaran, dan lain sebagainya.

Dalam melindungi sistem yang ada maka perlu merancang dan menerapkan beberapa pengendalian agar sistem tetap terjaga.

II.G. Pengembangan Sistem Informasi

II.G.1. Proses Pengembangan Sistem Informasi

Menurut Whitten, 2001:75 proses pengembangan sistem berguna bagi *stakeholder* untuk mengembangkan dan meningkatkan manfaat sistem informasi dan perangkat lunak yang terdiri dari sekumpulan aktivitas, metode, praktek terbaik, proses penyampaian dan peralatan otomatis.

II.G.2. Faktor Pengembangan Sistem Informasi

Faktor – faktor yang menyebabkan timbulnya suatu pengembangan sistem informasi, antara lain:

II.G.2.1. Adanya masalah yang timbul pada sistem lama (Problem)

Problem yang berarti terjadi terjadi ketidak beresan pada sistem lama misalnya:

- Adanya kecurangan.
- Kesalahan yang tidak disengaja.

- Tidak efisien.
- Tidak ditaatinya manajemen.

II.G.2.2. Meraih kesempatan baru (*Opportunity*)

Opportunity adalah suatu kesempatan untuk mengembangkan organisasi walaupun terdapat suatu masalah tertentu.

II.G.2.3. Adanya instruksi (*Directive*)

Directive adalah suatu kebutuhan baru yang terpengaruhi dari manajemen, perintah, dan beberapa pihak luar lainnya.

II.G.3. Metode Pengembangan Sistem

Metode pengembangan sistem menurut Mulyadi ,2001, merupakan pengembangan sistem oleh seorang analisis melalui beberapa langkah tertentu.

Berikut merupakan tiga tahapan dalam pengembangan sistem akuntansi:

II.G.3.1. Analisis Sistem

Pemakai sistem pasti akan memerlukan beberapa informasi tertentu dalam melakukan

pekerjaannya oleh karena itu pada tahap ini analisis sistem membantu pemakai sistem menganalisa apa saja informasi yang dibutuhkan.

II.G.3.2. Desain Sistem

Pada tahap ini desainer sistem menterjemahkan informasi yang telah didapat oleh analisis dari tahapan sebelumnya menjadi sebuah rancangan sistem informasi. Rancangan tersebut nantinya akan diberikan kepada pemakai untuk mendapatkan persetujuan.

II.G.3.3. Implementasi Sistem

Tahapan terpenting dari sebuah pengembangan sistem adalah tahap implementasi. Pada tahap ini teknisi memberikan arahan dan pelatihan kepada pemakai agar pemakai dapat menjalankan sistem baru tersebut untuk mendapatkan informasi. Selain itu memberikan arahan bagi teknisi yang bertanggung jawab dan melakukan pengujian terhadap sistem baru agar membuat sistem informasi yang baru dapat dioperasionalkan dengan maksimal.

II.G.4. Strategi Pengembangan Sistem

II.G.4.1. *Model Driven Development (MDD)*

Menurut Whitten, 2001 model Driven Development (MDD) merupakan cara atau strategi dalam sebuah pengembangan suatu system. MDD lebih mengutamakan rancangan pada gambar model yang akan menghasilkan gambaran dalam memenuhi kebutuhan bisnis, mengidentifikasi permasalahan, dan akan diimplementasikan dalam mendesain suatu sistem informasi.

Berikut merupakan contoh dari kelebihan dan kekurangan MDD. Kelebihan yang dimiliki MDD yaitu memperoleh hasil desain sistem dengan lebih matang, hal tersebut disebabkan oleh lebih mendalam dan mendetailnya perencanaan sistem MDD. Sedangkan kekurangan dalam model MDD yaitu waktu yang dibutuhkan untuk merancang sebuah model yang diinginkan cukup lama.

II.G.4.2. *Rapid Application Development (RAD)*

Rapid Application Development (RAD) adalah strategi pengembangan sistem yang menekankan pada kecepatan proses pengembangan. Proses pengembangan yang cepat tersebut ditempuh melalui keterlibatan pemakai secara luas dalam proses konstruksi serangkaian *prototype* yang terus dikembangkan hingga pada akhirnya menjadi sistem yang telah jadi (*Final System*) yang dapat bekerja sepenuhnya.

Berikut merupakan salah satu contoh kelebihan serta kekurangan dari RAD. Kelebihan yang dimiliki RAD yaitu waktu yang relatif cepat dalam pengembangan sistemnya serta menunjang kebutuhan pemakai. Kekurangan dari RAD sendiri yaitu memungkinkan muncul *prototype* yang dihasilkan tidak menjadikan solusi atas masalah yang sebenarnya, hal tersebut disebabkan oleh analisis masalah yang cenderung diabaikan.

II.G.4.3. Commercial Off the Shelf Package Software(COTS)

Commercial Off the Shelf Package Software (COTS) adalah strategi pengembangan sistem yang memanfaatkan sebuah paket perangkat lunak (*software*) yang dibeli untuk diterapkan pada suatu perusahaan dalam mendukung operasional sehari-hari.

Berikut merupakan kelebihan dan kekurangan dari COTS. Kelebihan dari COTS yaitu tidak memerlukan waktu untuk pemrograman sehingga waktu yang dibutuhkan dalam pengimplementasian sistem relative cepat. Sedangkan kelemahan dari COTS itu sendiri yaitu paket perangkat lunak yang dibeli harus disesuaikan baik dari perusahaan maupun pemakai.

II.H. Analisis Sistem

II.H.1. Pengertian Analisis Sistem

Analisis sistem adalah penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi

permasalahan, kesempatan, hambatan yang terjadi dan kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan (Jogiyanto, HM, 2001).

II.H.2. Tahap-Tahap Analisis dan Desain Sistem dengan RAD

II.H.2.1. Analisis Masalah

Analisis masalah dibutuhkan untuk mengetahui kebutuhan-kebutuhan apa saja yang diperlukan untuk suatu perusahaan apabila ingin mengembangkan suatu program guna kepentingan bisnisnya.

II.H.2.2. Identifikasi Kebutuhan

Identifikasi kebutuhan ini terkait dengan desain formulir yang diinginkan pemilik yang sesuai dengan kebutuhan perusahaan dalam menjalankan kegiatan operasionalnya.

II.H.2.3. Desain Sistem

II.H.2.3.1. Data

Data adalah bahan baku informasi yang berguna untuk keperluan informasi akuntansi. Dalam proses ini alur data sering

digambarkan dengan *Entity-Relationship Diagram (ERD)*.

II.H.2.3.2. Proses

Proses adalah suatu cara pelaporan akuntansi mulai dari data masuk (*input*) sampai hasil (*output*) dan pembuatan laporan keuangan. Dalam proses ini alur proses sering digambarkan menggunakan Diagram Alir Data atau *Data Flow Diagram (DFD)*.

II.H.2.3.3. Interface


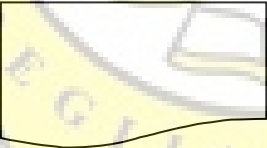
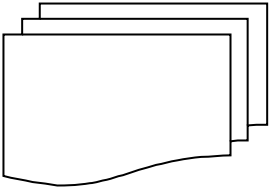
Interface adalah bentuk atau desain dari formulir yang diinginkan oleh pemilik dalam program yang akan dikembangkan berdasarkan hasil dari analisis masalah dan identifikasi kebutuhan. *Interface* hanya berupa desain formulir bukan hasil pengembangan sistem.

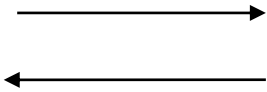
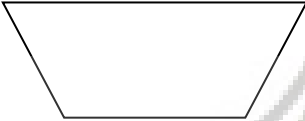
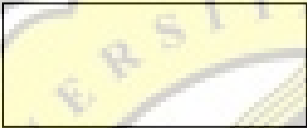
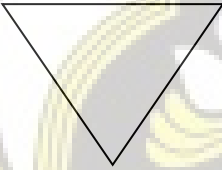
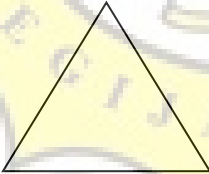
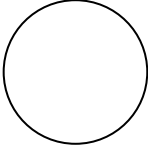
II.H.3. Alat Bantu dalam Analisis Sistem

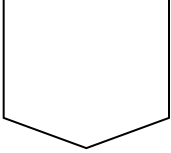
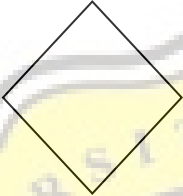




II.H.3.1. Diagram Alir Dokumen (*Document Flowchart*)


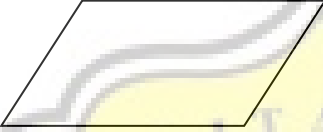
Diagram alir menunjukkan kontrol dari sebuah sistem aliran dokumen yang menunjukkan aliran data dan serangkaian tahapan dalam sebuah system (Whitten, Jeffrey L, et. Al., 2001).

Simbol-simbol diagram alir dokumen adalah sebagai berikut:

Simbol	Keterangan
	<p>1. Terminal</p> <p>Menunjukkan kegiatan awal dan akhir sebuah proses.</p>
	<p>2. Dokumen</p> <p>Menunjukkan dokumen dalam proses manual maupun sudah terkomputerisasi pada bagian <i>input</i> (formulir) dan <i>output</i> (laporan).</p>
	<p>3. Multi Dokumen</p> <p>Menunjukkan beberapa jenis dokumen yang digunakan dalam suatu waktu.</p>

	<p>4. Garis Alir</p> <p>Menunjukkan bagan pekerjaan manual.</p>
	<p>5. Proses Manual</p> <p>Menunjukkan pekerjaan manual yang sedang dikerjakan.</p>
	<p>6. Proses Berbasis Komputer</p> <p>Menunjukkan proses yang dilakukan oleh komputer.</p>
	<p>7. Arsip Sementara</p> <p>Menunjukkan tanda penyimpanan sebuah arsip sebagai back-up dalam pembuatan laporan keuangan maupun bahan audit suatu saat nanti.</p>
	<p>8. Arsip Permanen</p> <p>Menunjukkan penyimpanan arsip permanen</p>
	<p>9. Penghubung/Konektor antar halaman</p> <p>Menunjukkan penghubung bagan alir lain di halaman yang sama.</p>

	<p>10. Penghubung/ konektor kelain halaman</p> <p>Menunjukkan penghubung pada bagian lain di halaman yang sama.</p>
	<p>11. Kondisi</p> <p>Menunjukkan alternative pilihan dalam suatu proses.</p>
	<p>12. File Master</p> <p>Menunjukkan tempat penyimpanan untuk data master.</p>
	<p>13. File Transaksi</p> <p>Menunjukkan penyimpanan data tetapi bukan merupakan data master. Data tersebut dapat berupa file transaksi, referensi, dan lain sebagainya.</p>
	<p>14. Pemasukan Data</p> <p>Menunjukkan data masukan.</p>
	<p>15. Prosedur yang tidak terdefiniskan</p> <p>Menunjukkan prosedur lain yang sebenarnya bukan sistem prosedur yang telah dibuat.</p>

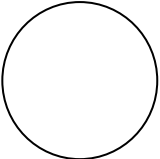
	<p>16. Display</p> <p>Menunjukkan <i>output</i> dalam bentuk tampilan layar monitor.</p>
	<p>17. Input / Output</p> <p>Menunjukkan data yang digunakan untuk proses masukan (<i>input</i>) dan menghasilkan laporan (<i>output</i>) dari proses tersebut.</p>

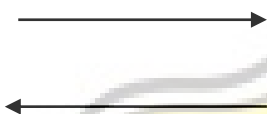
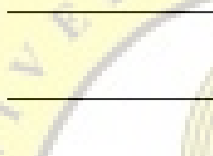

Tabel II.1 Simbol-Simbol *Flowchart*

II.H.3.2. *Data Flow Diagram (DFD)*

Menurut Mulyadi (2001:58) Simbol-simbol yang digunakan untuk menggambarkan sebuah aliran data melalui suatu proses yang saling berhubungan yang nantinya akan membentuk sebuah gambaran grafis disebut dengan *Data Flow Diagram (DFD)*.

Simbol-simbol yang digunakan dalam *Data Flow Diagram (DFD)*:

Simbol	Keterangan
	<p>1. Proses</p> <p>Menunjukkan proses transformasi dari sebuah data masukan (<i>input</i>) menjadi sebuah data</p>

	keluaran (<i>output</i>).
	2. Aliran Data Aliran yang digambarkan dengan dua ujung panah menunjukkan adanya dialog antar informasi data.
	3. Penyimpanan Menunjukkan sebuah tempat penyimpanan data.
	4. External Entity Lingkungan diluar sistem seperti orang atau sistem-sistem yang lain yang membentuk suatu kesatuan / entity.

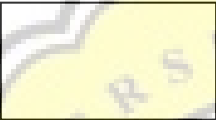
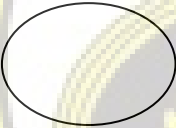

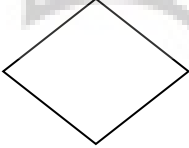

Tabel II.2 Simbol-Simbol *Data Flow Diagram (DFD)*

II.H.3.3. *Entity Relationship Diagram (ERD)*

Entity Relationship Diagram (ERD) merupakan suatu teknik grafis yang menggambarkan skema database. Diagram ini disebut sebagai diagram E-R karena diagram tersebut menunjukkan berbagai entitas

yang dimodelkan, serta hubungan antara entitas tersebut (Romney, 2004:134).

Simbol-simbol yang digunakan dalam Entity Relationship Diagram (ERD):

Simbol	Keterangan
	<p>1. Entitas (<i>Entity</i>)</p> <p>Menunjukkan suatu objek yang mengidentifikasi dalam lingkungan pemakai.</p>
	<p>2. <i>Attribute</i></p> <p><i>Entity</i> yang mempunyai elemen yang berfungsi mendiskripsikan karakter <i>entity</i>.</p>
	<p>3. <i>Associative Entity</i></p> <p>Sebuah <i>relationship</i> antara entitas yang memiliki hubungan <i>many to many</i>.</p>
	<p>4. Hubungan Relasi</p> <p><i>Entity</i> dapat berhubungan satu sama lain. Hubungan ini disebut <i>relationship</i>.</p>
	<p>5. Garis</p> <p>Simbol garis digunakan untuk menghubungkan <i>entity</i> dengan atributnya.</p>

Tabel II.3 Simbol-Simbol *Entity Relationship Diagram (ERD)*

II.I. Microsoft Visual Basic.Net

II.I.1. Pengertian Program Microsoft Visual Basic

Menurut Cushing, Barry E. 1992, *Visual Basic* adalah salah satu *development tools* untuk membangun aplikasi dalam lingkungan windows. Pada pemrograman, visual pengembangan aplikasi dimulai dengan pembentukan *user interface*, kemudian mengatur *property* dari objek-objek yang digunakan dalam *user interface* dan baru dilakukan penulisan kode program untuk menangani kejadian-kejadian (event).

II.I.2. Komponen Microsoft Visual Basic.Net

Langkah awal dari belajar Visual Basic adalah mengenal IDE (*Integrated Development Environment*). Penguasaan IDE yang baik akan sangat membantu dalam mengefektifkan tugas-tugasnya sehingga dapat bekerja secara efisien. Menurut Cushing, Barry E. 1992, komponen yang terdapat dalam IDE Visual Basic adalah sebagai berikut:

II.I.2.1. Menu Bar

Menu Bar berfungsi untuk memilih tugas-tugas tertentu, seperti memulai, memilih, mengakhiri suatu aplikasi, dan lain sebagainya. Dalam menu bar ini terdapat tiga belas menu utama, yaitu : *File, Edit, View, Project, Format, Debug, Run, Query, Diagram, Tools, Add-Ins, Windows, dan Help.*

II.I.2.2. Toolbar

Fungsi *toolbar* tidak jauh berbeda dengan menu bar, hanya saja pada *toolbar* masing-masing fungsi dinyatakan dalam bentuk simbol-simbol (*icon*) sehingga penggunaannya lebih praktis dan cepat. Fungsi-fungsi yang terdapat pada *toolbar* antara lain: *Add Project, Add Form, Menu Editor, Open, Save, Cut, Copy, Find, Undo, Redo, Start,* dan lain sebagainya.

II.I.2.3. Jendela Toolbox

Toolbox berisi komponen-komponen yang dapat digunakan untuk mendesain *user interface* pada suatu form. *Pointer, Testbox, Label, CheckBox, Frame, CommandButton, ListBox, ComboBox, Line, Shape,*

Image, dan lainnya merupakan beberapa komponen yang ada pada Jendela Toolbox.

Secara garis besar masing-masing *intrinsic control* memiliki fungsi sebagai berikut:

II.I.2.3.1. *Pointer* digunakan saat kita ingin memiliki kontrol-kontrol yang ada pada form. *Pointer* sendiri sebenarnya bukan merupakan suatu kontrol.

II.I.2.3.2. *Picture Box* adalah kontrol yang digunakan untuk menampilkan image dengan format: BMP, DIB (*bitmap*), ICO (*icon*), CUR (*cursor*), WMF (*metafile*), GIF dan JPEG.

II.I.2.3.3. *Label* adalah kontrol yang digunakan untuk menampilkan teks yang tidak dapat diperbaiki oleh pemakai.

II.I.2.3.4. *TextBox* merupakan kontrol yang mengandung *string* yang dapat diperbaiki oleh pemakai. *TextBox* dapat dapat berupa banyak baris atau hanya berupa satu baris tunggal.

II.I.2.3.5. *Frame* adalah kontrol yang digunakan *container* bagi kontrol lainnya.

II.I.2.3.6. *CommandButton* merupakan kontrol yang hampir ditemukan pada setiap *form* dan digunakan untuk membangkitkan *event* proses tertentu ketika pemakai melakukan klik padanya.

II.I.2.3.7. *CheckBox* digunakan untuk pilihan yang isinya bernilai *yes / No, True / False*.

II.I.2.3.8. *OptionButton* menampilkan beberapa opsi yang nantinya *user* hanya dapat dipilih satu pilihan dari beberapa opsi tersebut.

II.I.2.3.9. *ListBox* pada listbox ini *user* dapat memilih beberapa item karena disediakan beberapa item tetapi hal itu tetap bergantung pada *propertyMultiselect*.

II.I.2.3.10. *ComboBox* merupakan kombinasi dari *TextBox* dan suatu *ListBox* dimana pemasukan data dapat dilakukan dengan pengetikan maupun pemilihan.

II.I.2.3.11. *HscrollBar* dan *VscrollBar* digunakan untuk membentuk *scrollbar* berdiri sendiri.

II.I.2.3.12. *Timer* digunakan untuk proses background yang diaktifkan berdasarkan *interval* waktu tertentu.

II.I.2.3.13. *DriveListBox*, *DirListBox*, dan *FileListBox* biasanya digunakan untuk membentuk sebuah *dialog box* yang berkaitan dengan *file*.

II.I.2.3.14. *Shape* dan *Line* digunakan untuk membuat berbagai bentuk seperti persegi, garis, oval, dan lingkaran.

II.I.2.3.15. *Image* memiliki fungsi yang hampir sama seperti *imagebox* tetapi tidak dapat digunakan sebagai *container* bagi control yang lainnya.

II.I.2.3.16. *Data* digunakan untuk data binding.

II.I.2.3.17. *OLE* dapat digunakan sebagai tempat bagi program eksternal seperti *Microsoft Excel*, *Word*, dll.

II.I.2.4. Jendela Form Designer

User interface dirancang dan didesain pada Jendela Form Designer ini dan komponen-komponen pada *toolbox* diletakkan agar dapat membentuk sebuah *form* tertentu.

II.I.2.5. Jendela Kode Editor

Dalam pembuatan suatu aplikasi Jendela Kode Editor dapat digunakan untuk menuliskan *listing* program yang ada. Pada Jendela Kode Editor ini perintah-perintah yang dibutuhkan dituliskan dalam bahasa pemrograman oleh programmer.

II.I.2.6. Jendela Project

Form-form apa saja yang berada dalam project tertentu dapat dilihat melalui Jendela Project.

II.I.2.7. Jendela Properties

Kontrol-kontrol yang sudah ada dalam *form* dapat dimanipulasi oleh Jendela Properties ini.

II.I.2.8. Jendela Layout

Menurut Jaja Jamaludin Malik tahun 2007 tata letak form pada proyek ditampilkan pada jendela layout.

II.J. MySQL

MySQL merupakan *Relationship Database Management System* (RDBMS) didistribusikan secara gratis (open source) dan

berada di bawah naungan lisensi GPL (General Public License). Setiap orang bebas menggunakan MySQL dapat digunakan secara bebas oleh setiap orang tetapi tidak boleh dijadikan produk turunan yang bertujuan komersial atau demi keuntungan pribadi.

MySQL tergolong sebagai database server yang lebih unggul dalam *query* data bila dibandingkan database server lainnya. Hal ini terbukti dari kecepatan *query* MySQL dapat lebih cepat sepuluh kali dibanding PostgreSQL dan MySQL lebih cepat lima kali bila dibandingkan dengan *Interdatabase*. Kecepatan *query* tersebut berguna untuk *query* yang dilakukan oleh single user.

II.J.1. Structured Query Language (SQL)

Standar bahasa basis data relasional disebut dengan SQL. SQL yang menggunakan kombinasi kalkulus relasional dan aljabar relasional telah dikenal memiliki bahasa *query* yang *marketable* dan *user-friendly*. Selain itu SQL memiliki beberapa kemampuan lain diluar melakukan *query* basis data seperti memodifikasi data dalam basis data, mendefinisikan struktur data, dan menemukan konstrain sekuriti (Kusrini, M.Kom dan Andri Koniyo, 2007).

Bahasa SQL dibagi menjadi empat bagian:

II.J.1.1. Data Definition Language (DDL)

DDL digunakan untuk mendefinisikan data yang berhubungan dengan pembuatan dan penghapusan objek seperti *table*, *indeks*, bahkan hingga basis datanya sendiri. Contoh dari DDL adalah *DROP*, *CREATE*, *ALTER*.

II.J.1.2. Data Manipulation Language (DML)

DML digunakan untuk memproses manipulasi data pada label atau record. Contoh dari DML adalah *UPDATE*, *INSERT*, *DELETE*, dan *SELECT*.

II.J.1.3. Data Control Language (DCL)

DCL digunakan untuk mengendalikan akses ke database, contoh DCL adalah *GRANT* dan *REVOKE*.