

BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Dalam mendefinisikan sistem dapat menggunakan dua kelompok pendekatan sistem, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya.

Menurut Jerry FitzGerald, Ardra F. FitzGerald, Warren D. Stallings dalam Jogiyanto HM (1999, Hal 1-2) "Suatu sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu "

Merupakan salah satu definisi sistem dengan menggunakan pendekatan pada prosedur.

Sedangkan prosedur itu sendiri oleh Richard F. Neuschel didefinisikan sebagai suatu urutan operasi klerikal (tulis-menulis), biasanya melibatkan beberapa orang di dalam satu atau lebih departemen yang ditetapkan untuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis yang terjadi.

Pendekatan sistem yang menekankan pada komponen atau elemennya menurut Davis Gordon B. (1974, hal 81) bahwa, " sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu".

Komponen atau elemen merupakan subsistem-subsistem dalam suatu sistem yang tidak dapat berdiri lepas sendiri-sendiri namun akan saling membentuk suatu kesatuan.

2.1.1 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu yang saling berkaitan dan saling terintegrasi untuk mencapai tujuan. Karakteristik atau sifat - sifat tertentu itu adalah :

a. Komponen sistem (*Component*)

Komponen sistem dapat berupa suatu subsistem atau bagian dari sistem yang akan menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

b. Batas Sistem (*Boundary*)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan dengan sistem yang lainnya atau dengan lingkungan luarnya, yang memungkinkan suatu sistem dipandang sebagai satu kesatuan yang menunjukkan ruang lingkup dari sistem tersebut.

c. Lingkungan Luar Sistem (*Environment*)

Lingkungan luar dari suatu sistem adalah segala sesuatu diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat pula merugikan.

d. Penghubung Sistem (*Interface*)

Penghubung merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Dimana melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya.

e. Masukan Sistem (*Input*)

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan sistem dapat berupa masukan perawatan, yaitu energi yang dimasukkan supaya sistem dapat beroperasi, dan masukan sistem dapat pula berupa masukan sinyal, yaitu energi yang diproses untuk mendapatkan keluaran

f. Keluaran Sistem (*Output*)

Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat dijadikan masukan untuk subsistem yang lain atau kepada supra sistem.

g. Pengolah Sistem (*Proses*)

Suatu bagian sistem yang mengolah atau merubah masukan menjadi keluaran.

h. Sasaran Sistem (*Goal*)

Suatu sistem akan mempunyai tujuan (goal) atau sasaran karena apabila sistem tidak memiliki tujuan maka operasi tersebut tidak akan ada gunanya.

2.1.2 Klasifikasi Sistem

a. Sistem abstrak dan sistem fisik

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Sistem fisik adalah sistem yang ada secara fisik.

b. Sistem alamiah dan sistem buatan manusia

Sistem alamiah adalah sistem yang terjadi melalui proses alam, tidak dibuat oleh manusia. Sistem buatan manusia adalah sistem yang dirancang oleh manusia.

c. Sistem tertentu dan sistem tidak tentu

Sistem tertentu adalah sistem yang operasinya sudah dapat diprediksi. Sistem tidak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

d. Sistem tertutup dan sistem terbuka

Sistem tertutup adalah sistem yang tidak berhubungan dan tidak terpengaruh dengan lingkungan luarnya. Sistem terbuka adalah sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.

2.1.4 Pengertian Informasi

a. **Data**

Dalam Jogiyanto (1999) “ Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata ”.

Sedangkan kejadian-kejadian adalah sesuatu yang terjadi pada saat tertentu.

Informasi

Dalam Jogiyanto (1999, Hal 8) “ Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya ”.

2.1.5 Pengertian Sistem Informasi

Robert A. Leitch dan K. Roscoe Davis dalam Jogiyanto (1991, Hal 11) mendefinisikan bahwa sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Menurut John dan Gary Grudnitski, sistem informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan. Blok bangunan tersebut yaitu :

- a) Blok Masukan (*input block*)

Blok masukan mewakili data yang masuk ke dalam sistem informasi yang termasuk di dalamnya metode-metode dan media untuk menangkap data yang dapat berupa dokumen-dokumen dasar.

b) Blok Model

Blok model terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data.

c) Blok Keluaran

Produk keluaran sistem informasi adalah informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen.

d) Blok Teknologi

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

e) Blok Basis Data

Basis data adalah kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan dalam perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

f) Blok Kendali

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal yang dapat merusak sistem dapat dicegah ataupun apabila terdapat kesalahan maka akan cepat terselesaikan.

2.1.6 Pengertian Sistem Informasi Manajemen

Dalam Jogiyanto (1999, Hal 14 -15) “ SIM adalah kumpulan dari interaksi-interaksi sistem-sistem informasi yang menyediakan informasi baik untuk kebutuha manajerial maupun kebutuhan operasi ”.

2.1.7 Peranan Sistem Informasi Manajemen bagi manajemen

Manajemen membutuhkan informasi untuk mendukung pengambilan keputusan yang akan dilakukan. Sumber informasi untuk pengambilan keputusan manajemen bisa didapatkan dari informasi internal dan eksternal. Agar informasi yang dihasilkan oleh sistem informasi dapat mengena dan berguna, maka analisis sistem harus mengetahui kebutuhan-kebutuhan yang dibutuhkan oleh manajemen.

2.1.8 Pengertian Analisa Sistem Informasi

Analisa sistem dapat didefinisikan sebagai penguraian dari suatu sistem informasi yang utuh kedalam bagian-bagian komponeenya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan-hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

2.2 Peralatan Pendukung (Tools System)

2.2.1 Diagram Alir Data

Diagram Alir Data (DAD) atau Data Flow Diagram (DFD) digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau lingkungan fisik dimana data tersebut akan disimpan.

Dalam menggambarkan suatu DAD dengan menggunakan simbol-simbol DAD. Simbol-simbol DAD tersebut yaitu :

1. Kesatuan Luar (*External Entity*)

Merupakan kesatuan dilingkungan luar sistem yang dapat berupa orang, organisasi, atau sistem lainnya yang akan memberikan input atau pun menerima output dari sistem.

2. Arus Data (*Data Flow*)

Arus data menunjukkan arus dari data yang dapat berupa masukan untuk sistem atau pun hasil dari proses sistem.

3. Proses (*Process*)

Proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses

4. Simpanan Data (*Data Store*)

Simpanan data merupakan simpanan dari data yang dapat berupa suatu file atau database, arsip atau catatan manual kotak tempat penyimpanan, sistem dan agenda.

2.2.2 Bagan Alir

Bagan alir adalah bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi.

Pedoman-pedoman yang digunakan pada saat menggambar suatu alir data yaitu :

1. Bagan alir sebaiknya digambar dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
2. Kegiatan di dalam bagan alir harus ditunjukkan dengan jelas.
3. Harus ditunjukkan dari mana kegiatan akan dimulai dan dimana akan berakhir.
4. Masing-masing kegiatan di dalam bagan alir sebaiknya digunakan suatu kata yang mewakili suatu pekerjaan. ★
5. Masing-masing kegiatan di dalam bagan alir harus di dalam urutan yang semestinya.
6. Kegiatan yang terpotong dan akan disambung ditempat lain harus ditunjukkan dengan jelas menggunakan simbol penghubung.
7. Gunakan simbol-simbol bagan alir yang standar.

2.2.3 Kamus Data

Kamus data adalah katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi.

Isi kamus data terdiri dari :

1. Nama Arus Data

Arus data perlu dicatat karena kamus data dibuat berdasarkan arus data yang mengalir di DAD.

2. Alias

Alias perlu dituliskan karena data yang sama mempunyai nama yang berbeda untuk orang atau departemen satu dengan yang lainnya.

3. Bentuk Data

Bentuk data perlu dicatat dikamus data karena dapat digunakan untuk mengelompokkan kamus data kedalam kegunaannya sewaktu perancangan sistem

4. Arus Data *

Arus data menunjukkan dari mana data mengalir dan kemana data akan menuju.

5. Penjelasan

Untuk memperjelas lagi tentang makna dari arus data yang akan dicatat di kamus data, bagian ini juga dapat berisi dengan keterangan-keterangan tentang arus data tersebut.

6. Periode

Periode menunjukkan kapan terjadinya arus data ini.

7. Volume

Volume yang dicatat adalah volume tentang rata-rata dan volume puncak dari arus data.

8. Struktur Data

Struktur data menunjukkan arus data yang dicatat di kamus data terdiri dari *item-item* data apa saja.

Tabel 2.1. Notasi Struktur Data

Notasi	Keterangan
=	Terbentuk dari (<i>is composed</i>) atau terdiri dari (<i>consist of</i>) atau sama dengan (<i>is equivalent of</i>)
+	AND
[]	Salah satu dari (memilih dari salah satu dari elemen - elemen data didalam kurung braket ini)
	Sama dengan symbol [].
N{ }M *	Iterasi (elemen data didalam kurung brase beriterasi mulai minimum N kali dan Maksimum M kali).
()	Opsional (elemen data didalam kurung parenthesis sifatnya <i>optional</i> , dapat ada dan dapat tidak ada.
*	Keterangan setelah tanda ini adalah komentar.

Tabel 2.2. Notasi Tipe Data

Notasi	Keterangan
X	Untuk setiap karakter
9	Untuk setiap angka numerik
A	Karakter alfabet
Z	Angka 0 yang ditampilkan dalam spasi kosong
.	Sebagai pemisah ribuan.
,	Sebagai pemisah pecahan
-	Sebagai tanda penghubung
/	Sebagai tanda pembagi

2.2.4 Pengkodean

a. Konsep Dasar

Kode biasa digunakan untuk mengklasifikasikan data, memasukkan data kedalam komputer dan untuk mengambil bermacam-macam informasi yang berhubungan dengannya. Kode dapat dibentuk dari kumpulan angka, huruf, dan karakter-karakter khusus. Kode Numerik menggunakan 10 macam kombinasi angka dalam kode, kode alphabet menggunakan 26 kombinasi huruf dalam kode, dan kode alphanumeric menggunakan gabungan dari angka dan huruf dalam kode.

b. Jenis Kode

1) Kode Mnemonik

Bertujuan agar mudah diingat. Kode mnemonik dibuat dengan dasar singkatan atau mengambil sebagian karakter dari item yang akan diwakili dengan kode ini.

2) Kode Urut

Kode urut dapat disebut juga dengan kode seri (*serial kode*) merupakan kode yang nilainya urut antara satu kode dengan kode berikutnya.

3) Kode Blok

Kode blok mengklasifikasikan item kedalam kelompok blok tertentu yang mencerminkan satu klasifikasi tertentu atas dasar pemakaian maksimum yang diharapkan. ★

4) Kode Group

Kode group merupakan kode yang berdasarkan field-field dan tiap-tiap field kode mempunyai arti.

5) Kode decimal

Kode decimal mengklasifikasikan kode atas dasar 10 unit angka decimal dimulai dari angka 0 sampai dengan angka 9 atau dari 00 sampai dengan 99 tergantung dari banyaknya kelompok.

2.2.5 File

a. Jenis File

1. File Induk

Merupakan file yang sangat penting dan tetap terus ada selama hidup dari sistem informasi.

a. File induk acuan (*referenc maste file*), yaitu file induk yang recordnya relatif statis jarang berubah nilainya.

b. File induk dinamika (*dynamic master file*), yaitu file induk yang nilai dari record-recordnya sering berubah atau sering dimutakhirkan sebagai akibat dari suatu transaksi

2. File Transaksi

File yang digunakan untuk merekam data hasil dari suatu transaksi yang terjadi.

3. File Laporan

File yang berisi dengan informasi yang akan ditampilkan dan juga berguna untuk mempersiapkan pembuatan suatu laporan dan biasanya masih dilakukan oleh proses yang lainnya.

4. File Sejarah

File yang berisi dengan data masa lalu yang sudah tidak aktif lagi, tetapi perlu disimpan untuk keperluan mendatang.

5. File Pelindung

Merupakan salinan dari file-file yang masih aktif di database pada suatu saat tertentu.

6. File Kerja

Dibuat oleh suatu proses program secara sementara karena memori komputer tidak mencukupi atau untuk menghemat pemakaian memori selama proses dan akan dihapus bila proses telah selesai

b. Akses File

Suatu metode yang menunjukkan bagaimana suatu program komputer akan membaca record-record dari suatu file.

Metode akses file dibagi menjadi dua cara, yaitu :

1. Metode akses secara urut (*sequential access*)

Dilakukan dengan membaca atau menulis suatu record di file dengan membaca terlebih dahulu mulai dari record pertama, urut sampai dengan record yang diinginkan.

2. Metode akses secara langsung (*direct access / random access*)

Dilakukan dengan cara langsung membaca record pada posisinya di file tanpa membaca dari record pertama terlebih dahulu.

c. Organisasi File

Pengaturan dari record secara logika di dalam file dihubungkan satu dengan yang lainnya.

Organisasi File secara tradisional :

1. Organisasi file urut

Merupakan file dengan organisasi urut serta pengaksesan urut.

2. Organisasi fileurut berindex

Merupakan file dengan organisasiurut serta pengaksesan langsung.

3. Organisasi file akses langsung

Merupakan file dengan organisasi acak serta pengaksesan langsung.

Organisasi file database :

1. Struktur data berjenjang

Menunjukkan hubungan antara data membentuk suatu jenjang.

2. Struktur data jaringan

Menunjukkan suatu hubungan antara data membentuk jenjang dan setiap node (simpul) mempunyai lebih dari satu orangtua.

3. Struktur data hubungan

Menunjukkan hubungan antara bentuk record yang didasarkan pada nilai field kunci.

2.3 Microsoft Visual Basic.NET

- Sejarah Singkat Visual Basic.NET

Sebelum versi *Visual Basic.NET* diluncurkan, Microsoft terlebih dahulu meluncurkan *Visual Basic 6* pada tahun 1998. Setelah beberapa tahun kesuksesan dari *Visual Basic 6*, Microsoft akhirnya mempresentasikan mengenai pengembangan Microsoft .NET dalam PDC (*Professional Developers Conference*) di Orlando, Florida, Amerika Serikat pada bulan Juli 2000.

Setelah melalui masa pengembangan yang cukup lama pada bulan februari 2002 secara resmi meluncurkan versi beta dari *Visual Studio.NET* dimana salah satu bahasa pemrogramannya adalah *Visual Basic.NET*. Setahun setelah *Visual Basic.NET* diluncurkan, Microsoft merilis *Visual Studio.NET 2003*.

- Keistimewaan Visual Basic.Net

Berikut ini adalah beberapa alasan kenapa *Visual Basic.NET* lebih baik dari Visual Basic 6 :

- a. Menyederhanakan Pengembangan (*Deployment*)

Visual Basic.NET mengatasi masalah seputar *deployment* dari aplikasi berbasis Windows yaitu masalah pada DLL (*Dynamic Link Library*) yang lebih dikenal dengan *DLL Hell*, dan *registrasi COM* (*Component Object Model*). Masalah ini dapat terjadi karena ketika kita menginstall suatu aplikasi baru, aplikasi tersebut akan menyebabkan beberapa aplikasi yang digunakan komponen sama seperti *file*, DLL atau komponen COM lainnya.

- b. Menyederhanakan Pengembangan Perangkat Lunak

Visual Basic.NET memiliki fitur *compiler* yang bekerja secara *background* bersifat *realtime* dan daftar tugas untuk penanganan kesalahan pada program, sehingga pengembang *software* dapat langsung memperbaiki kesalahan kode yang terjadi. Selain itu, *Visual Basic.NET* juga menyediakan akses langsung ke *.NET Framework*

yang memungkinkan pengembang secara cepat mengakses *registry*, *log event* dan *alias file*.

c. Mendukung Penuh Objek Berorientasi Pemrograman

Visual Basic.NET memiliki fitur bahasa pemrograman berorientasi objek termasuk implementasinya secara penuh seperti pewarisan (*inheritance*), pembungkusan (*encapsulation*) dan banyak bentuk (*polymorphism*).

d. Mempermudah Pengembangan Aplikasi Berbasis Web

Untuk mengembangkan aplikasi web disediakan *designer form web*, dimana digunakan mekanisme "*Drag and Drop*" untuk membangun *form*. Klik ganda dan tulis kod untuk menyatakan suatu kejadian atau *event*.

- Struktur Program Dalam *Visual Studio.NET*

Platform .NET merupakan satu set kumpulan teknologi yang memungkinkan teknologi internet ditransformasikan ke dalam *platform distributed computing* dengan skalabilitas dan kompatibilitas tinggi. Secara teknikal, *.NET Platform* menyediakan konsep pemrograman dengan *library* dan modul-modul baru yang konsisten, terlepas dari jenis bahasa pemrograman yang digunakan. *Microsoft .NET* merupakan *software* yang menghubungkan informasi, orang, sistem dan perangkat yang dapat menjangkau *client*, *server* dan *tool* pengembang. *Microsoft .NET* disusun oleh beberapa komponen, diantaranya :

a. *.NET Framework (Dot Net Framework)*

Ada dua komponen inti *.NET Framework*, kedua komponen tersebut adalah *Common Language Runtime (CLR)* yang menyediakan *runtime environment* untuk aplikasi yang *Visual Studio.NET*, terlepas dari jenis bahasa pemrogramannya. dan komponen yang kedua yaitu *.NET Framework Class Library* (kumpulan-kumpulan kelas dalam *.NET Framework*).

Secara sederhana *.NET Framework* adalah *platform* tunggal dimana semua orang dapat mengembangkan aplikasi menggunakan suatu *platform* yang mirip dengan *Java Virtual Machine (JVM)*, sehingga aplikasi yang dikembangkan menggunakan bahasa pemrograman seperti VB, C++, C# dan J# serta bahasa pemrograman lainnya yang kompatibel dengan *.NET Framework*. Bagi seorang yang mengembangkan teknologi *.NET*, pemahaman terhadap konsep dan objek model *.NET Framework* adalah jauh lebih penting daripada bahasa pemrograman itu sendiri. Berikut ini adalah beberapa kegunaan dari *.NET Framework* :

- 1) Menyediakan lingkungan pemrograman berorientasi objek, apakah kode objek dijalankan dan disimpan secara local atau dijalankan melalui jarak jauh (*remote*).
- 2) Menyediakan lingkungan untuk menjalankan suatu kode yang meminimalkan konflik saat *software* didistribusikan.
- 3) Menyediakan lingkungan untuk menjalankan suatu kode yang dapat mengurangi masalah performa.

4) Membuat pengembang memiliki pengalaman yang konsisten dalam berbagai tipe aplikasi seperti aplikasi berbasis *Windows* dan aplikasi berbasis *Web*.

b. Common Language Runtime (CLR)

Common Language Runtime (CLR) merupakan komponen dasar dari *.NET Framework*. CLR bertugas mengendalikan dan menyediakan sejumlah layanan (*service*) dari sebuah program, seperti berikut ini :

- Mengelola dan mengendalikan kode baik dalam keadaan *loading* maupun pada saat program tersebut telah berjalan.
- Mengelola memori dalam *□ias□m*.
- Mengisolasi memori aplikasi.

Ketika sebuah program berbasis *.NET (dot net)* dijalankan, maka kode dalam program tersebut harus melalui CLR terlebih dahulu. Kode yang sesuai dengan ketentuan dalam CLR disebut dengan *managed code* (kode terkelola), sedangkan kode yang tidak sesuai dengan ketentuan CLR disebut dengan *unmanaged code* (kode yang tidak terkelola).

c. .NET Framework Class Library (dot net framework class library)

.NET Framework Class Library atau pustaka *Class .NET Framework* merupakan koleksi dari tipe yang dapat digunakan secara berulang-ulang dan terintegrasi dengan CLR. Kelas pustaka sifatnya berorientasi objek, menyediakan tipe-tipe dimana *managed code* dapat mewariskan fungsionalitasnya. *Inheritance* atau pewarisan menyediakan cara bagi pengembang untuk memakai ulang fungsi yang ada tanpa harus

menulis kembali kodenya. Tipe-tipe *.NET Framework* memungkinkan anda untuk menyelesaikan sejumlah tugas pemrograman umum termasuk pengelolaan *string*, koleksi data, konektivitas *database* dan akses *file*.

2. Macam-macam Proyek Dalam *Visual Studio.NET*

Apabila kita hendak membuat sebuah aplikasi baru, pada menu *New Project* terdapat beberapa pilihan *templates* yang menunjukkan macam-macam proyek yang bias dibuat baik menggunakan bahasa pemrograman *Visual Basic*, *C#*, *C++* dan *J#* diantaranya yaitu :

a. *Windows Application*

Tipe proyek ini untuk membuat aplikasi dengan antarmuka *Windows*.

b. *Class Library*

Komponen untuk membuat kode dasar, dimana tidak ada antarmuka yang terlihat. *Class* adalah kumpulan dari fungsi yang dapat digunakan dalam proyek lainnya. *Class Library* sama dengan tipe proyek *ActiveX DLL* dan *ActiveX EXE* pada program *Visual Basic 6*.

c. *Windows Control Library*

Kontrol windows seperti *textbox* dan *label* adalah elemen dasar untuk membuat antarmuka. Jika control yang disediakan dalam *Visual Basic.NET* tidak dapat memenuhi kebutuhan *programmer*, anda dapat membuat control sendiri menggunakan *Windows Control Library*. *Windows Control Library* sama dengan tipe proyek *ActiveX* pada program *Visual Basic 6*.

2.4 Microsoft Access

Pengertian Microsoft Access atau Microsoft Office Access adalah sebuah program aplikasi basis data komputer relasional., dan juga menggunakan tampilan grafis yang intuitif sehingga memudahkan pengguna. Para pengguna/programmer yang telah berpengalaman dapat menggunakannya untuk mengembangkan perangkat lunak aplikasi yang kompleks, sementara para programmer yang kurang berpengalaman dapat menggunakannya karena Microsoft Access merupakan program yang telah disetting sedemikian rupa agar para penggunanya baik dari para programmer yang handal atau tidak. Access juga menawarkan teknik-teknik pemrograman berorientasi objek.

2.5 Persediaan Barang

2.5.1 Jenis-jenis persediaan barang menurut fungsinya

- Batch stock/Lot size inventory

Persediaan yang dilakukan karena kita membeli atau membuat bahan-bahan atau barang-barang dalam jumlah yang lebih besar dari jumlah yang dibutuhkan saat ini.

Keuntungannya :

- Potongan harga pada harga pembelian
- Efisiensi produksi
- Penghematan biaya angkutan

- Fluctuation stock

Persediaan yang diadakan untuk menghadapi fluktuasi permintaan konsumen yang tidak dapat diramalkan.

- Anticipation stock

Persediaan yang diadakan untuk menghadapi fluktuasi permintaan yang dapat diramalkan, berdasarkan pola musiman yang terdapat dalam satu tahun dan untuk menghadapi penggunaan atau penjualan atau permintaan yang meningkat.

2.5.2 Jenis-jenis persediaan menurut jenis dan posisi barang

- Persediaan bahan baku
- Persediaan barang produk/komponen yang dibeli
- Persediaan bahan-bahan pembantu/penolong
- Persediaan barang-barang setengah jadi/dalam proses
- Persediaan barang jadi

2.5.3 Biaya-biaya yang timbul dari persediaan

- Biaya pemesanan
- Biaya yang terjadi dari adanya persediaan
- Biaya kekurangan persediaan
- Biaya yang berhubungan dengan kapasitas

2.5.4 Cara mengukur jumlah persediaan

- Periodik sistem
- Perpetual sistem

2.5.5 Pengawasan persediaan

Tujuannya :

- Menjaga jangan sampai kehabisan persediaan
- Supaya pembentukan persediaan stabil
- Menghindari pembelian kecil-kecilan
- Pemesanan yang ekonomis

2.5.6 Catatan penting dalam sistem pengawasan persediaan

- Permintaan untuk dibeli
- Laporan penerimaan
- Catatan persediaan
- Daftar permintaan bahan
- Perkiraan pengawasan

2.5.7 Macam-macam biaya persediaan

- Holding cost atau carrying cost

Adalah biaya yang timbul karena perusahaan menyimpan persediaan

- Ordering cost atau set-up cost

Adalah biaya yang berhubungan dengan pemesanan dan pengadaan barang

- Stock-out cost

Adalah biaya yang timbul akibat perusahaan kehabisan persediaan

2.5.8 Fungsi-fungsi persediaan

- **Fungsi decoupling**

Adalah persediaan yang memungkinkan perusahaan dapat memenuhi permintaan langganan tanpa tergantung pada supplier, atau untuk menghadapi fluctuation stock.

- **Fungsi economic lot sizing**

Persediaan lot size ini perlu mempertimbangkan penghematan-penghematan atau potongan pembelian, biaya pengangkutan per unit menjadi lebih murah dan sebagainya. Hal ini disebabkan karena perusahaan melakukan pembelian dalam kuantitas yang lebih besar dibandingkan dengan biaya-biaya yang timbul karena besarnya persediaan (biaya sewa gudang, investasi, resiko dan sebagainya).

- **Fungsi antisipasi**

Apabila perusahaan menghadapi fluktuasi permintaan yang dapat diramalkan berdasarkan pengalaman atau data-data masa lalu, yaitu permintaan musiman. Dalam hal ini perusahaan dapat mengadakan persediaan musiman (*seasonal inventories*).