

BAB II

LANDASAN TEORI

II.1 Teori Sistem Informasi

II.1.1 Definisi

Sistem Informasi, kata tersebut mungkin sudah tidak asing lagi ditelinga kita. Sebagian dari kita mungkin tidak tahu apa itu Sistem Informasi. Sistem Informasi terdiri dari 2 kata yaitu **sistem** dan **informasi**. Sumber dari informasi adalah data. Data adalah fakta dan angka yang tidak sedang digunakan pada proses keputusan, dan biasanya berbentuk catatan historis yang dicatatkan dan diarsipkan tanpa maksud untuk segera diambil kembali untuk pengambilan keputusan.

Definisi dari **Sistem** adalah beberapa set komponen yang bekerja bersama - sama untuk menampilkan suatu tugas. Definisi dari **Informasi** adalah data yang diambil kembali, diolah, atau sebaliknya digunakan sebagai dasar untuk peramalan atau pengambilan keputusan. Jika kita lihat definisi dari masing - masing kata penyusunnya, sistem informasi dapat didefinisikan sebagai :

- a. Suatu set komponen yang bekerja bersama - sama untuk mengatur penambahan, penyimpanan, manipulasi dan distribusi dalam informasi.
- b. Sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan atau untuk mengendalikan organisasi.

II.1.2 Komponen

Komponen dari sistem informasi terdiri dari :

- a. Perangkat keras, meliputi seluruh perlengkapan fisik yang membentuk suatu komputer.
- b. Perangkat lunak, adalah instruksi - instruksi yang digunakan untuk menjalankan perangkat keras tersebut.
- c. Orang - orang, seperti kita, dapat menciptakan produk, pemberian pelayanan, menyelesaikan masalah dan membuat keputusan. Orang yang menggunakan *hardware* komputer dan perangkat lunak untuk menampilkan suatu tugas kadang terhubung sebagai *end user* atau *user* saja. Sebagai *end user* dalam sistem informasi kita harus mengetahui apa yang dapat dilakukan untuk kita, dan bagaimana menggunakannya secara efektif untuk menyelesaikan informasi yang kita butuhkan.
- d. Data juga memiliki peranan yang sangat penting dalam sistem informasi, data menyediakan dasar - dasar dari informasi.
- e. Prosedur, merupakan instruksi yang membantu *user* dalam pengoperasian dan berguna sebagai sistem informasi.

II.1.3 Desain Sistem Informasi

Sistem informasi didesain untuk digunakan oleh banyak *user* yang biasa disebut dengan *multiuser*. Hal tersebut dapat ditemukan dalam banyak bisnis dan organisasi dan merupakan peranan yang sangat penting dalam kesuksesan operasi. Sistem informasi yang digunakan oleh *user* secara individu biasa disebut dengan *personal*

sistem informasi. Kita dapat men - set up sistem informasi *personal* dalam komputer rumah untuk mengatur *portfolio* keuangan kita.

Sistem informasi memiliki tiga fungsi dasar :

1. Menerima data (*input*)
2. Merubah data menjadi informasi (*proses*)
3. Untuk memproduksi dan mengkomunikasikan informasi ke dalam *timely fashion* bagi *user* untuk membuat keputusan (*output*). Sebagai contoh, banyak bank dan institusi keuangan yang menggunakan sistem informasi untuk membantu menentukan apakah nasabah disetujui untuk melakukan pinjaman.

Input	Proses	Output
Data	Data ke informasi	Informasi
Spesifikasi nasabah	Algoritma mengubah data menjadi informasi	Transformasi dalam bentuk formulir yang digunakan
Kebijakan peminjaman	yang diinginkan oleh karyawan bank	
Tingkat bunga	(c/o : kredit yang diinginkan nasabah, rekomendasi untuk bisa melakukan pinjaman atau tidak)	(c/o : kwitansi)

Tabel 2.1 Contoh kasus

Data yang berisi tentang nasabah, kebijakan peminjaman, dan tingkat bunga dimasukkan ke dalam *input*

menjadi sistem informasi. Data kemudian diproses dengan menggunakan prosedur yang berkelanjutan untuk membuat keputusan yang berharga. Akhirnya, informasi dikomunikasikan kepada petugas bank (*user*) sebagai rekomendasi untuk menerima atau menolak pinjaman. Hal ini penting untuk dicatat karena bagaimanapun output yang dihasilkan, namun *user* dari sistem informasi dapat menciptakan keputusan yang aktual.

Produk - produk sekarang banyak yang tidak dapat diproduksi tanpa adanya kegunaan secara efektif dalam sistem informasi dalam tingkat produksi. Aerospace, otomotif dan industri manufaktur menggunakan sistem informasi untuk mengotomatisasikan dan mempersingkat produksi, kecepatan perkembangan waktu, pengurangan biaya dan tetap berkompetisi. Industri - industri ini membutuhkan komitmen yang kuat dan investasi dalam sistem informasi untuk tetap sukses bersaing dalam biaya, kualitas dan pengiriman.

Selama beberapa dekade, manajemen informasi dan komputer serta sistem informasi telah meningkat menjadi lebih penting dalam pencapaian tujuan organisasi. Informasi merupakan bagian penting dalam organisasi. Sebagian aktivitas manajer dalam organisasi - seperti identifikasi masalah, solusi, kontrol dan pembuatan keputusan - berdasarkan pada keputusan. Manajer membutuhkan penerimaan yang akurat dan informasi setiap waktu untuk menyelesaikan segala aktivitas secara efektif.

Komputer dan sistem informasi memperbolehkan informasi untuk memperoleh, memproses dan mendistribusi secara efisien. Mereka memperbolehkan informasi untuk mendapatkan persaingan yang kompetitif karena secara akurat, tepat pada waktunya dan secara lengkap informasi menyediakan keputusan yang lebih baik untuk dibuat.

Komunikasi dengan kenaikan angka dalam departemen lain, manajer dan *staff* berhubungan dengan teknologi baru dan perawatan dalam pertumbuhan angka dalam kompetitor merupakan seluruh faktor yang dapat menambah kompleksitas manajer dalam bekerja. Suatu sistem informasi dapat membantu manajer dalam mengontrol peningkatan kompleksitas dengan cara menjamin ketepatan informasi yang dikomunikasikan secara akurat dan dengan cara yang tepat pada waktunya.

Ketika didesain, dikembangkan dan digunakan secara benar, sistem informasi merupakan peralatan yang sangat penting dalam meningkatkan efisiensi dan efektivitas dalam organisasi untuk memperoleh, memproses, mengkomunikasikan dan menggunakan informasi. Sistem informasi yang efektif dapat memungkinkan suatu organisasi untuk mengambil keputusan yang tepat dan meraih tujuan mereka.

II.1.4 Pengertian Basis Data dan DBMS

Basis data atau *database* adalah kumpulan informasi yang tersimpan secara elektronik pada sebuah *file*. Suatu *database* merupakan koleksi dari informasi yang berhubungan dengan subjek atau fungsi tertentu. Dengan

menggunakan suatu *Database Management System* (DBMS) kita dapat menempatkan data-data dan mengorganisasikannya menjadi tabel-tabel.

DBMS adalah desain software untuk membantu dalam memelihara dan memanfaatkan dari sekumpulan / koleksi data yang besar. Atau dapat dikatakan bahwa DBMS adalah program yang menyediakan akses ke *database*. DBMS mengizinkan organisasi untuk memusatkan data, menata secara efisien dan menyediakan akses untuk data yang disimpan dengan program aplikasi. DBMS juga bertindak sebagai antarmuka antara program aplikasi dan *file* data secara fisik dan menyediakan *user* dengan *tools* untuk menambah, menghapus, memelihara, menampilkan, mencetak, mencari, memilih, mengurutkan dan meng-*upgrade* data.

Keuntungan dari DBMS yaitu :

1. Data independence : program aplikasi dapat dimungkinkan menjadi *independent* dari detail representasi data dan penyimpanan data. DBMS dapat menyediakan tampilan abstrak dari data untuk mengisolasi kode aplikasi dari detail.
2. Akses data yang efisien : DBMS memanfaatkan beragam teknik yang menggunakan untuk menyimpan dan mencari kembali data secara efisien. Kondisi ini akan sangat penting jika data disimpan pada peralatan penyimpanan *external*.
3. Integritas dan keamanan data : jika data selalu diakses melalui DBMS maka DBMS dapat memaksa batasan/kekangan integritas dari data.

4. Administrasi data : ketika beberapa *user* membagi data, pemusatan administrasi data dapat menawarkan peningkatan yang signifikan.
5. Akses bersama dan *crash recovery* : DBMS menjadwalkan akses bersama pada data dimana *user* dapat memandang data dapat diakses hanya oleh 1 *user* pada 1 waktu. Lebih jauh DBMS melindungi *user* dari efek kegagalan sistem.
6. Pengurangan waktu pengembangan aplikasi : DBMS mendukung banyak fungsi penting yang umum pada setiap aplikasi mengakses penyimpanan data pada DBMS.

II.2 Sistem Informasi Berbasis Mobile

Dalam beberapa tahun terakhir, banyak perhatian ditujukan pada pengembangan *mobile computing* di dalam industri TI. Berbagai perangkat *mobile* yang berhasil dikembangkan mencakup handphone, PDAs (Personal Digital Assistants), *communicators*, dan komputer portable. Mobilitas telah menjadi sebuah paradigma komputasional yang baru, dimana proses-proses yang terjadi berpindah bersama dengan *user* atau pengguna.

Tentunya ada beberapa penjelasan mengapa mobilitas dan *mobile computing* berkembang dengan pesat. Beberapa alasan dapat ditemukan dari lingkungan atau masyarakat, organisasi, dan *tools*. Corak dasar masyarakat modern adalah mobilitas dimana jika dibandingkan 50 tahun yang lalu kebanyakan orang jarang meninggalkan rumahnya. Dalam konteks organisasi, alasan utama untuk melakukan mobilitas adalah karena kebanyakan pekerjaan dalam perusahaan modern

beralih ke arah pekerjaan yang dikerjakan secara bersama atau tim. Pada bentuk organisasi sebelumnya umumnya tidak mempromosikan bentuk pekerjaan secara bersama-sama atau tim. Adapun penjelasan lainnya adalah adopsi *mobile phones*, dimana didesain untuk memungkinkan orang-orang dapat berhubungan satu dengan lainnya seperti tanpa ada jarak yang memisahkan. Hal inilah yang kemudian menjadi cara baru yang dapat diterapkan dalam berbagai organisasi. Dalam perkembangannya, sebuah sistem informasi jarak jauh yang memberikan hak akses khusus anggota mulai mencoba menggunakan media perangkat *mobile* sebagai sarana untuk menghubungkan antar pengguna dalam sebuah sistem. Saat ini banyak diantara kita bergantung pada sistem dan aplikasi yang menggunakan media perangkat *mobile* yang harus berjalan dengan baik dan terpercaya, misalnya saja sistem *mobile banking*.

Sistem informasi berbasis *Mobile* adalah suatu sistem informasi yang menggunakan media perangkat *mobile* sebagai perantara. Sistem informasi itu sendiri memiliki pengertian suatu set komponen yang bekerja bersama - sama untuk mengatur penambahan, penyimpanan, manipulasi dan distribusi dalam informasi atau sekumpulan prosedur organisasi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan atau untuk mengendalikan organisasi. Jadi pada sistem informasi berbasis *mobile* ini, berbagai proses yang terdapat pada sistem informasi dapat dilakukan dan diakses melalui berbagai perangkat *mobile* yang telah kompatibel.

II.3 Teknologi SMS (Short Messaging Service)

SMS (Short Messaging Service) merupakan teknologi yang memungkinkan untuk mengirimkan dan menerima pesan dalam bentuk *alphanumeric* ke dan dari handphone, Personal Digital Assistants (PDAs) dan PC (Personal Computers).

Isu SMS pertama kali muncul di belahan Eropa pada sekitar tahun 1991 bersama sebuah teknologi komunikasi *wireless* yang saat ini cukup banyak penggunaannya, yaitu *Global System for Mobile Communication* (GSM). Dipercaya bahwa pesan singkat pertama yang dikirimkan menggunakan SMS dilakukan pada bulan Desember 1992, dikirimkan dari sebuah Personal Computer (PC) ke telepon *mobile* (bergerak) dalam jaringan GSM milik perusahaan Vodafone Inggris. Perkembangannya kemudian merambah ke benua Amerika, dipelopori oleh beberapa operator komunikasi bergerak berbasis digital seperti BellSouth Mobility, PrimeCo, Nextel, dan beberapa operator lain. Teknologi digital yang digunakan bervariasi dari yang berbasis GSM, *Time Division Multiple Access* (TDMA), hingga *Code Division Multiple Access* (CDMA).

Untuk dapat menggunakan *Short Message Service*, pengguna perlu melakukan pendaftaran atau langganan ke operator jaringan selular yang menyediakan layanan SMS dan memerlukan perangkat komunikasi *mobile* yang mendukung SMS.

Sebuah *short message* tunggal dapat mencapai panjang 160 karakter dari suatu teks ketika menggunakan alfabet Latin dan mencapai 70 karakter ketika menggunakan alfabet non-Latin seperti Arabic dan Chinese. Panjang 160 karakter tersebut dapat meliputi kata atau angka atau kombinasi

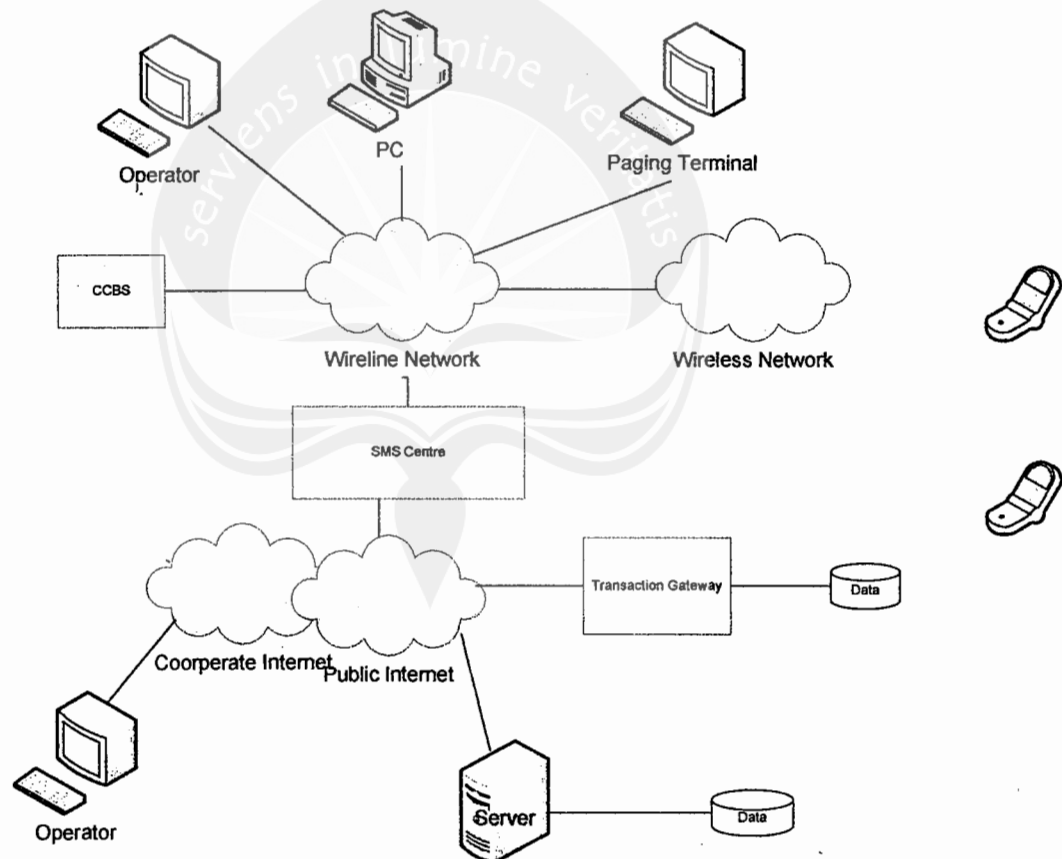
phanumeric. Selain itu *short message* juga ada yang tidak berbasis teks (misalnya, dalam format binary). Jenis *short message* ini digunakan untuk *ringtone* dan layanan logo.

Dalam sistem SMS, mekanisme utama yang dilakukan adalah melakukan pengiriman *short message* dari satu terminal pelanggan ke terminal yang lain. Hal ini dapat dilakukan berkat adanya sebuah entitas dalam sistem SMS yang bernama *Short Message Service Centre (SMSC)*, disebut juga *Message Centre (MC)*. SMSC merupakan sebuah perangkat yang melakukan tugas *store and forward traffic short message*. Di dalamnya termasuk penentuan atau pencarian rute tujuan akhir dari *short message*.

Layanan SMS merupakan sebuah layanan yang bersifat *nonreal time* dimana sebuah pesan singkat dapat di-*submit* ke suatu tujuan, tidak peduli apakah tujuan tersebut aktif atau tidak. Bila dideteksi bahwa tujuan tidak aktif, maka sistem akan menunda pengiriman ke tujuan hingga tujuan aktif kembali. Pada dasarnya sistem SMS akan menjamin *delivery* dari suatu *short message* hingga sampai tujuan. Kegagalan pengiriman yang bersifat sementara seperti tujuan tidak aktif akan selalu teridentifikasi sehingga pengiriman ulang *short message* akan selalu dilakukan kecuali bila diberlakukan aturan bahwa *short message* yang telah melampaui batas waktu tertentu harus dihapus dan dinyatakan gagal terkirim.

Karakteristik utama SMS adalah SMS merupakan sebuah sistem pengiriman data dalam paket yang bersifat *out-of-band* dengan bandwidth kecil. Dengan karakteristik ini, pengiriman *short message* dapat dilakukan dengan efisiensi

yang tinggi. Pada awalnya SMS diciptakan untuk menggantikan layanan *paging* dengan menyediakan layanan serupa yang bersifat *two-way-messaging* ditambah dengan *notification service* untuk voice mail. Pada perkembangan selanjutnya, muncul jenis-jenis layanan lain seperti email, fax, *interactive banking*, *information service*, *Location Based Services* (LBS) dan integrasi dengan aplikasi berbasis internet. Integrasi dengan aplikasi berbasis internet kemudian mendorong timbulnya layanan seperti *web-based messaging*, *gaming*, dan *chatting*.



Gambar 2.1 Arsitektur Jaringan SMS

II.4 Mobile FBUS 1.5

FBUS merupakan suatu metode komunikasi diantara produk handphone Nokia dan port serial pada PC yang digunakan untuk melakukan transmisi data, layanan, dan penyesuaian.

Mobile FBUS 1.5 adalah produk dari Software Cave dan merupakan *ActiveX Control* yang bersifat *freeware* dimana dapat digunakan untuk membuat perangkat lunak atau *software* untuk handphone, seperti mengirim SMS, memanipulasi logo operator, mengakses phonebook, dan lain-lain. Mobile FBUS merupakan tool yang ideal untuk menghubungkan perangkat komunikasi seperti handphone dengan aplikasi Visual Basic. Mobile FBUS dapat bekerja pada produk handphone GSM Nokia, dimana dalam hal ini Software Cave telah melakukan uji coba pada produk handphone Nokia seri 51xx/61xx dan 3210. Berikut ini adalah beberapa acuan dalam pemrograman Mobile FBUS 1.5 yaitu :

1. Properties

Kebanyakan *properties* dari *ActiveX Control* Mobile FBUS hanya dapat digunakan ketika kontrol tersebut terhubung dengan handphone.

a. ACDCPower property

Boolean-Read only

Properti ini mengembalikan nilai True/Benar ketika handphone dihubungkan dengan sumber tenaga dari luar, misalnya baterai pembangkit.

b. BatteryLogo property

Single-Read only

Properti ini mengembalikan tingkat/level baterai dari handphone yang terhubung. Unit yang digunakan tergantung pada model handphone.

c. CellId property

String-Read only

Properti ini mengembalikan *Cell Id* dari node jaringan pada handphone yang sedang terhubung. *Cell Id* bersifat unik untuk setiap transmitter atau pemancar jaringan GSM, walaupun *Location Area Code (LAC)* berada di antara sejumlah pemancar pada area yang sama.

d. Connected property

Boolean-Read only

Properti ini akan mengembalikan nilai jika kendali terhubung ke handphone.

e. DateTime property

Date-Read/Write

Properti ini mengembalikan atau men-set tanggal dan waktu pada jam yang ada dalam handphone yang terhubung.

Catatan :

- Anda tidak dapat men-set detik karena detik yang menjadi bagian dari format VB Date akan diabaikan ketika Anda men-set waktu alarm.
- Pada handphone Nokia seri 51xx Anda tidak dapat men-set tanggal melalui antarmuka yang ada pada handphone, sehingga kemungkinan dapat berisi nilai random ketika Anda melakukan query. Meskipun demikian, Anda dapat men-set tanggal

melalui Mobile FBUS dan setelah itu handphone Anda akan tetap terset pada tanggal yang benar.

f. KeyboardLocked property

Boolean-Read only

Properti ini mengembalikan nilai True jika papan tombol pada handphone yang terhubung berada dalam posisi tidak dapat digunakan.

g. ProviderCode property

String-Read only

Properti ini mengembalikan kode provider untuk provider jaringan yang ada dalam format "ccc nn", dimana ccc merupakan kode negara sebanyak 3 digit dan nn adalah kode jaringan. Sebagai contoh, provider Belanda Ben memiliki kode provider "204 16".

h. ProviderCountry property

String-Read only

Properti ini mengembalikan nama negara provider selular dari handphone yang terhubung.

i. ProviderName property

String-Read only

Properti ini mengembalikan nama provider selular dari handphone yang terhubung.

j. ProviderSMSC property

String-Read only

Properti ini mengembalikan nomor handphone dari Short Message Service Centre (SMSC) provider selular pada handphone yang terhubung.

k. RfLevel property

Single-Read only

Properti ini mengembalikan kekuatan sinyal GSM yang terdeteksi oleh handphone. Unit yang digunakan tergantung pada model handphone.

1. SMSClass property

Integer-Read/Write

Properti ini mengembalikan atau men-set kelas pesan SMS yang akan digunakan dalam metode SendSMSMessage. Kelas pesan mendefinisikan bagaimana pesan ditampilkan oleh handphone penerima. Beberapa kelas yang biasa digunakan antara lain :

fbSmsClassFlash pesan tertampil sebagai pesan sekejap (terlihat dengan cepat pada layar handphone tanpa konfirmasi pengguna).

fbSmsClassNormal pesan diterima sebagai pesan SMS biasa.

fbSmsClass2 pengaruhnya sama dengan fbSmsClass Normal pada handphone Nokia.

fbSmsClass3 pengaruhnya sama dengan fbSmsClassNormal pada handphone Nokia.

2. Method

a. Connect method

Sub Connect [sComport As String="COM1"]

Metode ini digunakan untuk menghubungkan kendali pada handphone. Hal ini merupakan hal yang mendasar yang harus dilakukan, seolah-olah tidak ada metode lain yang akan bekerja apabila metode ini tidak dipanggil.

Parameter :

sComport-Optional-Port yang digunakan untuk menghubungkan. Jika diabaikan, maka yang digunakan adalah COM1. Tuliskan nama comport sebagai berikut "COM<nomor>" (misalnya : COM1, COM2, COM3 dan lain-lain).

Nilai :

Tidak ada

b. Disconnect method

Sub Disconnect

Metode ini akan mengakhiri hubungan dengan handphone dan dengan metode ini akan menonaktifkan seluruh metode yang ada dalam kontrol.

Parameter :

Tidak ada

Nilai :

Tidak ada

3. Collections

a. Phonebook

Phonebook(iLocation As Integer, [iMemoryType As fbMemory]) As FBPhonebookEntry

Collection ini dapat digunakan untuk mengatur phonebook yang tersimpan dalam SIM atau memory handphone.

Parameter :

iLocation - Mandatory - Lokasi memory untuk dibaca atau ditulisi.

iMemoryType - Optional - Jenis memory yang akan diakses. Jika parameter ini diabaikan, maka akan digunakan memory SIM.

Properties :

Group - Read/Write Integer - Grup ID untuk entri phonebook. 0 jika entri bukan bagian dari grup.

Last Error - Read only Integer - Kesalahan terakhir dari penggunaan phonebook. 0 jika penggunaan sukses.

Name - Read/Write String - Nama untuk entri phonebook.

Number - Read/Write String - Nomor telepon untuk entri phonebook.

Method :

WriteData-Gunakan perintah ini untuk menuliskan Nama, Nomor dan Grup ID per satu kali penggunaan. Perintah ini lebih cepat daripada memasukkan tiga properties tadi satu per satu.

4. SMS Objek

Obyek ini dapat digunakan untuk membaca dan menulis pesan SMS. *SMS Object* memiliki beberapa properties, metode dan sub-objek sebagai berikut.

Properti	Deskripsi
Last Error	Menghasilkan kode kesalahan terakhir
TotalMessage	Menghasilkan jumlah total SMS yang disimpan
UnreadMessage	Menghasilkan jumlah pesan yang belum dibaca
Method	Deskripsi
Refresh	Mengambil pesan yang tersimpan dalam handphone
SendMessage	Mengirim pesanSMS
Object	Deskripsi
DeliveryNotification	Message box berisi notifikasi pengiriman
Inbox	Message box berisi SMS yang diterima
Outbox	Message box berisi SMS yang

	terkirim
FBSmsMessage	Sub-Object yang mengacu pada satu pesan di dalam kotak objek

Tabel 2.2 Properties dan Method SMS Object

4.1 Properties

a. SMS.LastError property

Integer-Read only

Properti ini akan mengembalikan error internal terakhir yang terjadi saat memanggil properti atau metode SMS. Last Error akan bernilai 0 ketika operasi telah terakhir selesai.

b. SMS.TotalMessages property

Integer-Read only

Properti ini akan mengembalikan total SMS yang ada di dalam SIM. Properti ini dapat berjalan meskipun SMS.Refresh tidak dijalankan.

c. SMS.UnreadMessages property

Integer-Read only

Properti ini akan mengembalikan jumlah SMS yang tidak terbaca dalam memory.

4.2 Methods

a. SMS.Refresh method

Sub Refresh()

Metode ini akan me-refresh daftar pesan yang tersimpan dalam SMS.Inbox, SMS.OutBox dan SMS.Delivery-Notification.

Parameter :

Tidak ada

Nilai :

Tidak ada

b. SMS.SendMessage method

*Function SendMessage(sDest As String, sMessage As String, [iSmsClass As fbSmsClass=fbSmsClassNormal])
As Boolean*

Metode ini akan mengirimkan SMS ke sDest, dengan isi sMessage. Parameter iSMSClass dapat digunakan untuk mengontrol tipe SMS yang dikirimkan. Tipenya adalah :

1. fbSmsClassFlash : pesan tertampil sebagai pesan sekejap (terlihat dengan cepat pada layar handphone tanpa konfirmasi pengguna).
2. fbSmsClassNormal : pesan diterima sebagai pesan SMS biasa.
3. fbSmsClass2 : pengaruhnya sama dengan fbSmsClassNormal pada handphone Nokia.
4. fbSmsClass3 : pengaruhnya sama dengan fbSmsClassNormal pada handphone Nokia.

Nilai :

Bernilai True jika fungsi sukses.

4.3 Sub-Objek

a. SMS.DeliveryNotifications/inbox/Outbox objects

Objek ini merupakan enkapsulasi dari 3 kotak SMS utama, dengan masing-masing kotak memiliki properti *Count*, yang merupakan jumlah pesan di dalam inbox. Pesan-pesan memiliki indeks di antara 1 dan Count.

b. FBSmsMessage object

Setiap kotak berisi sebuah *Collection* dari objek pesan yang bertipe *FBSmsMessage*, setiap pesan memiliki properties dan method seperti berikut ini.

Properti	Deskripsi
DateTime	Waktu dari pesan (read-only Date)
Destination	Tujuan Pesan (read-only String)
LastError	Error yang terakhir terjadi, bernilai 0 jika sukses (read-only Integer)
Sender	Pengirim Pesan (read-only String)
SentRead	Akan bernilai True jika pesan telah terkirim atau terbaca (read-only Boolean)
Text	Isi dari pesan (read-only String)
Method	Deskripsi
Delete	Menghapus pesan dari SIM atau memori telepon

Tabel 2.3 Properties dan Method FBSmsMessage object

II.5 HTML Parsing

II.5.1 Pengertian HTML Parsing

HTML *Parsing* adalah suatu metode untuk menguraikan teks HTML untuk mencari informasi - informasi berharga di dalam suatu teks HTML. Secara khusus tidak ada penggunaan metode yang benar - benar spesifik dan signifikan dalam proses *Parsing* HTML tersebut. Namun secara tidak langsung ada 2 cara yang banyak digunakan untuk *memparsing* teks HTML itu sendiri, 2 cara tersebut antara lain :

1. Dengan membuat fungsi *parsing* sendiri. Ada beberapa orang yang membuat fungsi *Parsing* HTML sendiri dengan menggunakan langkah - langkah yang sederhana,

namun di balik kesederhanaan tersebut seringkali hasil yang didapat malah lebih baik dibanding dengan menggunakan software *parsing* buatan orang atau perusahaan tertentu. Salah satu cara untuk membuat fungsi *parsing* sendiri adalah dengan menggunakan *regular expression*.

2. Dengan menggunakan fungsi *parsing* yang telah dibuat oleh vendor atau perusahaan tertentu. Sebagai contoh adalah fungsi MSHTML pada aplikasi Microsoft Visual Basic 6.0, dengan menggunakan fungsi *parsing* tersebut kita dapat *memparsing* suatu teks HTML dengan lebih cepat dan lebih mudah walaupun hasil yang didapat belum tentu sesuai dengan keinginan kita.

Kegunaan HTML *Parsing* itu sendiri adalah untuk mencari informasi berharga yang ada disuatu teks HTML. Dengan HTML *parsing* ini informasi lebih cepat didapat karena kita tidak harus membuka halaman HTML yang hendak *diparsing*, namun cukup dengan langsung mengambil source HTML tersebut dan dengan menggunakan HTML *Parsing* kita ambil informasi - informasi penting yang kita butuhkan.

II.5.2 Proses HTML Parsing

Secara garis besar proses *parsing* adalah sebagai berikut. HTML adalah suatu bahasa pemrograman yang dibangun atas dasar tag - tag yang sudah didefinisikan dari awal. Tiap-tiap tag tersebut itulah yang nantinya akan *diparsing*. Tag - tag yang hendak *diparsing* harus mempunyai suatu tag awal yang sesuai dengan tag akhir (<

pagetitle> ... </pagetitle>). Hal ini bisa terjadi karena dalam informasi - informasi yang terdapat di dalam suatu teks HTML tersebut biasanya berada di dalam suatu tag - tag khusus di mana tag tersebut harus mempunyai tag awal dan tag akhir, informasi yang berada di antara tag awal dan tag akhir itulah yang nanti akan kita ambil. Tag akhir adalah tag yang sama dengan tag awal namun yang membedakannya adalah tag akhir memiliki karakter tanda slash (/) setelah karakter "<". Contohnya :

```
<table>
  <table>
    .....
  </table>
</table>
```

Dengan melihat hal tersebut maka proses *parsing* dapat lebih mudah dilakukan. Jika kita menggunakan metode *parsing* sederhana maka kita hanya perlu mencari karakter akhir ">" dari tag awal dan karakter awal "</" dari tag akhir. Lalu informasi yang berada di antara 2 karakter itulah yang akan kita ambil untuk diolah informasinya. Proses tersebut kita lakukan berulang kali/secara rekursif untuk mengecek dan membandingkan setiap karakter yang ada di suatu teks HTML dari awal sampai akhir teks HTML sambil mengambil informasi yang kita butuhkan.

Secara tidak langsung proses *parsing* HTML sederhana ini hanya akan mengumpulkan informasi yang berada di antara tag awal dan tag akhir saja. Untuk dapat mencari informasi yang berada di dalam suatu tag HTML (Semisal informasi link, ... , atau informasi lainnya) kita kadang perlu membutuhkan waktu dan kompleksitas pemrograman yang lebih tinggi.

Namun untuk memperoleh informasi tersebut kita tidak perlu khawatir, karena saat ini sudah banyak vendor atau perusahaan yang sudah membuat fungsi - fungsi *Parsing* HTML yang dilengkapi dengan kemampuan - kemampuan tersebut. Keunggulannya adalah kita tidak perlu tahu bagaimana cara/proses *parsingnya*, kita hanya perlu tahu bagaimana cara menggunakan method - method atau fungsi - fungsi yang sudah mereka sediakan, jika kita sudah mengetahui hal tersebut maka proses *parsing* tersebut akan lebih mudah dilakukan dan informasi yang didapat akan jauh lebih lengkap.

II.6 Saham

II.6.1 Pengertian Saham

Saham adalah salah satu bukti kepemilikan atau penyertaan seseorang atau badan dalam suatu perusahaan. Saham terdiri dari 2 jenis (Kamus Investasi:<http://www.e-bursa.com/referensi/investasi/default.php>) :

1. Saham preferen : Saham yang mempunyai hak likuidasi lebih tinggi dibanding saham biasa.
2. Saham biasa : Saham yang umum diperdagangkan baik di pasar perdana/sekunder.

Possible return suatu saham datang dari *fluktuasi* harga yang terjadi di pasar (*gain*)/dalam bentuk *corporate action* yang dilakukan perusahaan, seperti pembagian *dividen* tunai, bonus saham, dan sebagainya (*Impact* dan kalkulasinya dapat dilihat di bagian *corporate action*).

Selain *possible return*, saham juga mempunyai *posible loss* baik karena *fluktuasi* harga, kepailitan *emiten*

ataupun penutupan saham yang dilakukan pengadilan/pemerintah. Banyak saham di tahun 1998-1999 yang mengalami penutupan khususnya saham - saham perbankan. Secara umum memang jarang terjadi *delisting* (penghapusan keikutsertaan suatu perusahaan dari daftar saham) di bursa, tapi kemungkinan itu tetap ada.

II.6.2 Pengertian Pasar Modal/Bursa

Pasar modal ataupun bursa merupakan tempat pembeli dan penjual efek/surat berharga yang terdaftar dalam suatu lembaga bertemu bursa beserta pernak pernik peraturannya untuk menjamin berlangsungnya proses jual beli tersebut secara tertib dan adil (Kamus Investasi:<http://www.ebursa.com/referensi/investasi/default.php>). Berdasarkan UU No.8 th 1990 tentang pasar modal, definisi bursa efek adalah pihak yang menyelenggarakan dan menyediakan sistem dan atau sarana untuk mempertemukan penawaran jual dan beli efek pihak - pihak lain dengan tujuan memperdagangkan efek diantaranya. Dengan kata lain bursa efek adalah pihak yang menyediakan media perdagangan efek, antara lain saham. Media tersebut dipergunakan untuk memperdagangkan efek oleh anggota - anggotanya (perusahaan efek). Agar perdagangan efek berjalan dengan aman, teratur dan efisien maka bursa mengatur dan mengawasi tata cara perdagangan di bursa melalui suatu peraturan bursa efek.

Dengan demikian bursa ini mirip dengan pengertian pasar pada umumnya. Perbedaannya adalah yang diperdagangkan efek (surat berharga tenda kepemilikan)

terhadap perusahaan yang *listing*/terdaftar di pasar modal dan khusus melalui anggotanya dengan aturan yang ditetapkan untuk *fairness* suatu transaksi.

Di Indonesia sampai saat ini dikenal 2 bursa saham, Bursa Efek Jakarta (BEJ) dan Bursa Efek Surabaya (BES). Bursa Efek Jakarta (BEJ) fokus pada perdagangan saham dan waran untuk industri skala menengah ke atas. Sedangkan Bursa Efek Surabaya (BES) lebih pada saham skala kecil menengah, produk *derivatif* dan *obligasi*.

II.6.3 Proses dan Aktivitas Perdagangan Saham di BEJ

Perdagangan saham dan waran di pasar reguler satuannya adalah 1 *lot*. 1 *lot* sama dengan 500 lembar saham, kecuali untuk industri perbankan, di mana 1 *lot* sama dengan 5000 lembar saham. Sedangkan *right* diatur secara keseluruhan dalam 5000 lembar perlotnya.

Investor dapat melakukan perdagangan sesuai dengan jam *trading* yang ada di BEJ. Waktu perdagangan dibagi 2, sesi pertama dan sesi kedua.

- Sesi pertama dimulai jam 09.30 WIB - 12.00 WIB
- Sesi kedua dimulai jam 13.30 WIB - 16.00 WIB

Selama hari kerja kecuali Jumat di mana sesi pertama ditutup jam 11.30 dan sesi kedua dimulai jam 14.00 WIB. Untuk hari Sabtu dan Minggu libur.

II.7 Kurs

II.7.1 Kurs Mata Uang

Menurut Beams dan Jusuf (2000 : 468), mata uang fungsional dari suatu entitas merupakan mata uang yang

berlaku di wilayah operasi utama perusahaan. Jadi mata uang fungsional dari sebuah entitas luar negeri adalah mata uang yang digunakan perusahaan tersebut untuk menghasilkan dan membelanjakan uang kas mereka. Secara sederhana, PSAK No. 10 mendefinisikan mata uang asing sebagai mata uang selain mata uang pelaporan suatu perusahaan. Sedangkan mata uang pelaporan sendiri didefinisikan sebagai mata uang yang digunakan dalam menyajikan laporan keuangan.

Beams dan Jusuf (2000 : 469) melanjutkan penjelasannya dengan mengatakan bahwa tujuan dari suatu mata uang adalah menyediakan suatu standar nilai, alat pertukaran, serta unit pengukuran. Pada dasarnya semua mata uang berfungsi sebagai unit pengukuran bagi kegiatan ekonomi di negara-negara yang bersangkutan. Suatu transaksi dikatakan dinilai dengan mata uang tertentu jika besarnya dinyatakan dalam mata uang tersebut. Dari pengertian ini kemudian muncul istilah kurs mata uang. Beberapa definisi mengenai kurs mata uang adalah sbb. :

1. Menurut FASB, yang dimaksud dengan nilai tukar mata uang (kurs) adalah : Rasio antara satu unit mata uang dengan sejumlah mata uang lain yang dapat ditukar pada waktu tertentu.
2. Sedangkan menurut Jamli (1998 : 22) menyebutkan bahwa : Kurs mata uang merupakan harga dari suatu mata uang dalam mata uang yang lain

Dari definisi-definisi tersebut dapat disimpulkan bahwa kurs mata uang adalah perbandingan nilai antar mata uang yang menunjukkan harga suatu mata uang jika

dipertukarkan dengan mata uang lain pada waktu tertentu. Jamli (1998 : 22) menuturkan, kurs mata uang asing atau valuta asing juga menunjukkan jumlah satuan valas yang dipersiapkan oleh pembeli dan penjual untuk pertukaran dengan mata uang domestik atau valas lainnya. Secara lebih spesifik, kurs ini ditunjukkan sebagai jumlah satuan mata uang domestik yang dipersiapkan oleh pembeli dan penjual untuk dipertukarkan dengan satu unit valuta asing, atau dengan kata lain nilai tukar valuta asing digunakan untuk mentranslasikan transaksi mata uang sendiri dengan mata uang asing.

Yuliati dan Prasetyo (1998 : 69) mengungkapkan, bila ditinjau secara teoritis, dalam kondisi tanpa intervensi pemerintah harga suatu mata uang juga ditentukan oleh permintaan dan penawaran terhadap mata uang itu. Kurs terbentuk pada saat jumlah dan harga mata uang yang diminta sama dengan jumlah dan harga mata uang yang ditawarkan. Kondisi ini disebut sebagai kondisi keseimbangan atau ekuilibrium. Kondisi keseimbangan dapat berubah setiap saat, jika faktor-faktor yang mempengaruhi dan /atau penawaran berubah. Faktor-faktor tersebut antara lain :

1. Laju inflasi relatif.
2. Tingkat bunga relatif.
3. Tingkat pendapatan relatif.
4. Kontrol pemerintah.
5. Pengharapan pasar.

II.7.2 Transaksi Dalam Mata Uang Asing

Dalam dunia perdagangan (bisnis) yang semakin lama semakin kompleks ini, terutama dengan adanya persetujuan perdagangan dunia, transaksi yang dilakukan antar perusahaan di negara yang berbeda telah menjadi sebuah pemandangan umum dalam dunia bisnis. Keadaan ini menimbulkan dua macam transaksi (Beams & Jusuf, 2000 : 472), antara lain :

1. Transaksi luar negeri, yaitu transaksi yang terjadi antar negara atau antar perusahaan dari negara yang berbeda
2. Transaksi mata uang asing, yaitu transaksi yang nilai tukarnya dinyatakan dalam mata uang selain mata uang fungsional suatu entitas.

Lebih lanjut, Beams dan Jusuf menjelaskan bahwa sebuah transaksi luar negeri tidak otomatis merupakan transaksi mata uang asing. Jenis transaksi luar negeri yang paling sering dilakukan adalah ekspor impor barang dan jasa. Transaksi ini bukanlah transaksi mata uang asing kecuali jika nilai tukarnya didenominasi dalam suatu mata uang asing, yang berarti mata uang yang digunakan adalah mata uang selain mata uang fungsional entitas.¹ Dalam situasi seperti ini, perusahaan/entitas tersebut harus menjual atau membeli mata uang asing agar transaksi tersebut dapat segera terselesaikan.

Sementara itu PSAK No. 10 menyebutkan bahwa suatu perusahaan dapat melakukan aktivitas yang menyangkut valuta asing (*foreign activities*) dalam dua cara, yaitu melakukan transaksi dalam mata uang asing, atau memiliki kegiatan usaha di luar negeri (*foreign operations*). Untuk

memasukkan transaksi dalam valuta asing pada laporan keuangan suatu perusahaan, transaksi harus dinyatakan dalam mata uang pelaporan perusahaan.

PSAK No. 10 dengan gamblang juga menjelaskan bahwa yang dimaksud dengan kegiatan usaha luar negeri (*foreign operation*) adalah suatu anak perusahaan (*subsidiary*), perusahaan asosiasi (*associates*), usaha patungan (*joint venture*) atau cabang perusahaan pelapor yang aktivitasnya dilaksanakan di suatu negara di luar negara perusahaan pelapor. Sedangkan transaksi dalam mata uang asing dinyatakan sebagai suatu transaksi yang didenominasi atau membutuhkan penyelesaian dalam suatu mata uang asing, termasuk transaksi yang timbul ketika perusahaan :

1. Membeli atau menjual barang atau jasa yang harganya didenominasi dalam suatu mata uang asing
2. Meminjam (hutang) atau meminjamkan (piutang) dana yang didenominasi dalam suatu mata uang asing
3. Menjadi suatu pihak untuk suatu perjanjian dalam valuta asing yang belum terlaksana
4. Memperoleh atau melepaskan aktiva, menimbulkan atau melunasi kewajiban yang didenominasi dalam suatu mata uang asing

Menurut Hidayati (1999) dalam Sumardana (2004), transaksi dalam mata uang asing sering menimbulkan keuntungan dan kerugian karena adanya perbedaan kurs tukar, baik itu keuntungan dan kerugian transaksi maupun keuntungan dan kerugian penjabaran, dalam kasus penyusunan laporan konsolidasi untuk perusahaan anak di luar negeri. Sebagai contoh, pada tahun 1994 sebuah

perusahaan Indonesia meminjam uang kepada sebuah perusahaan di Amerika Serikat sebesar US\$ 100,000. Pada waktu itu nilai tukar Dollar AS terhadap Rupiah adalah 1 US\$ = Rp. 5.000,-. Bila dinyatakan dalam Rupiah maka jumlah hutang perusahaan Indonesia tersebut adalah sebesar Rp. 500.000.000, (100,000 x Rp. 5.000,-). Hutang itu akan dilunasi pada tahun 1997.

Namun pada tahun 1997 nilai tukar Dollar AS menguat 200 % terhadap mata uang Rupiah, sehingga 1 US\$ = Rp. 10.000,-. Jika diasumsikan perusahaan Indonesia tidak melakukan tindakan apa-apa untuk melindungi nilai hutangnya, maka perusahaan Indonesia tersebut harus mengembalikan hutangnya sebesar Rp.1.000.000.000,- (100,000 x Rp.10.000,-). Ini artinya, perusahaan Indonesia mengalami kerugian selisih kurs mata uang sebesar Rp. 500.000.000,-. Hal inilah yang sebenarnya terjadi pada sebagian perusahaan Indonesia pada tahun 1997-1998, yang mengakibatkan perusahaan-perusahaan tersebut harus gulung tikar untuk membayar hutangnya, atau bahkan harus meminta bantuan pemerintah untuk melunasi hutang-hutangnya.

Keuntungan atau kerugian yang terjadi dalam transaksi mata uang asing memang suatu hal yang biasa terjadi dalam sebuah perdagangan global. Namun hal itu tetap saja akan menjadi sebuah masalah besar ketika secara tidak terduga terjadi kerugian yang sampai mempengaruhi operasi perusahaan bahkan membuatnya bangkrut.