

BAB II

LANDASAN TEORI

2.1 Sistem Informasi

Secara umum informasi adalah suatu pengetahuan yang bermakna dan berguna bagi pengguna informasi itu. Informasi memberikan nilai keuntungan tersendiri bagi perusahaan dan manajer perusahaan, karena informasi sangat dibutuhkan untuk mengambil keputusan dan melakukan tindakan yang tepat sesuai keputusan yang telah diambil. Informasi biasanya diperoleh dari data yang telah diproses, dimana informasi ini mewakili hasil akhir dari suatu sistem informasi. Data adalah fakta, gambar, maupun simbol yang belum diolah yang secara bersama-sama membentuk masukan bagi suatu sistem informasi.

Informasi merupakan sebuah sumber daya penting dalam sebuah organisasi. Informasi menjadi dasar, energi atau mesin penggerak bagi organisasi penggunanya. Dengan informasi yang tepat, sebuah organisasi dapat meningkatkan kinerja dan kualitas hasil kerjanya.

Namun, informasi tidak akan bisa membantu penggunanya jika tidak ditangani dengan baik. Penanganan informasi membutuhkan sebuah sistem yang mampu mengolah data-data menjadi informasi yang berguna bagi organisasi.

Suatu sistem pada dasarnya adalah sekelompok unsur yang erat berhubungan satu dengan yang lainnya, yang berfungsi bersama-sama untuk

mencapai tujuan tertentu. Setiap sistem memiliki suatu batasan tersendiri yang memisahkannya dari lingkungan sekitarnya. Kebanyakan sistem merupakan sistem yang terbuka, dimana sistem-sistem ini menerima masukan dari luar dan menghasilkan keluaran ke lingkungan luarnya tersebut. Sistem ini juga merupakan sistem yang nyata, dalam arti sistem menggunakan sumber daya fisik seperti alat-alat tertentu dan juga manusia.

Setiap sistem terdiri dari unsur-unsur. Unsur-unsur suatu sistem terdiri dari subsistem yang lebih kecil, yang terdiri pula dari kelompok unsur yang membentuk subsistem tersebut. Unsur sistem tersebut bekerja sama untuk mencapai tujuan sistem. Setiap sistem mempunyai tujuan tertentu. Setiap sistem dibuat untuk menangani sesuatu yang berulang kali atau yang secara rutin terjadi. Suatu sistem merupakan bagian dari sistem lain yang lebih besar.

Pengolah informasi atau yang saat ini lebih dikenal dengan nama sistem informasi tidak memiliki definisi secara pasti. Pengertian sistem informasi tidak bisa dilepaskan dari pengertian sistem dan informasi. Paulus Mudjihartono dalam bukunya yang berjudul *Sistem Informasi : Teori, Metodologi, dan Tool* menyatakan bahwa secara lugas sistem informasi didefinisikan sebagai kumpulan orang, prosedur, *hardware*, *software* yang saling berinteraksi yang pada saat dilaksanakan akan memberikan informasi bagi pengambil keputusan dan/atau untuk mengendalikan organisasi maupun

memberikan suatu pelayanan informasi bagi *user*. Berbeda dengan pengertian sistem, sistem informasi merupakan kumpulan dari objek yang jelas dan tujuan yang jelas pula.

Dalam buku *Sistem Akuntansi* yang ditulis oleh Mulyadi dinyatakan bahwa seperti halnya suatu bangunan rumah, sistem informasi memiliki komponen utama yang membentuk struktur bangunan sistem informasi. Komponen bangunan sistem informasi terdiri dari enam blok yang disebut dengan *information system building block*. Keenam blok tersebut yaitu :

1. Blok masukan (*Input Block*)

Masukan adalah data yang dimasukkan ke dalam sistem informasi beserta metode dan media yang digunakan untuk menangkap dan memasukkan data tersebut ke dalam sistem. Masukan terdiri dari transaksi, permintaan, pertanyaan, perintah, dan pesan. Umumnya masukan harus mengikuti aturan dan bentuk tertentu mengenai isi, identifikasi, otorisasi, tata letak, dan pengolahannya. Cara untuk memasukkan masukan ke dalam sistem dapat berupa tulisan tangan, formulir kertas, pengenalan karakteristik fisik seperti sidik jari, papan ketik (*keyboard*) dan lain-lain.

2. Blok Model (*Model Block*)

Blok model terdiri dari *logico-mathematical models* yang mengolah masukan dan data yang disimpan, dengan berbagai macam cara, untuk

memproduksi hasil yang dikehendaki atau keluaran. *Logico-mathematical model* dapat mengkombinasi unsur-unsur data untuk menyediakan jawaban atas suatu pertanyaan, atau dapat meringkas atau menggabungkan data menjadi suatu laporan ringkas.

3. Blok Keluaran (*Output Block*)

Produk suatu sistem informasi adalah keluaran yang berupa informasi yang bermutu dan dokumen untuk semua tingkat manajemen dan semua pemakai informasi, baik pemakai intern maupun pemakai luar organisasi. Keluaran suatu sistem merupakan faktor utama yang menentukan blok-blok lain suatu sistem informasi. Jika keluaran suatu sistem informasi tidak sesuai dengan kebutuhan pemakai informasi, perancangan blok masukan, keluaran, model, teknologi, basis data, dan pengendalian tidak akan ada manfaatnya. Mutu yang harus melekat dalam keluaran sistem informasi adalah : ketelitian, ketepatan waktu, dan relevansi. Media yang dipakai untuk menyajikan keluaran sistem informasi dapat berupa : layar monitor, mesin pencetak (*printer*), alat pendengar (*audio*), atau *microfilm*.

4. Blok Teknologi (*Technology Block*)

Teknologi ibarat mesin untuk menjalankan sistem informasi. Teknologi menangkap masukan, menjalankan model, menyimpan dan mengakses data, menghasilkan dan

menyampaikan keluaran, serta mengendalikan seluruh sistem. Dalam sistem informasi berbasis komputer, teknologi terdiri dari tiga komponen : komputer dan penyimpan data di luar (*auxiliary storage*), telekomunikasi, dan perangkat lunak (*software*).

5. Blok Basis Data (*Database Block*)

Basis data merupakan tempat untuk menyimpan data yang digunakan untuk melayani kebutuhan pemakai informasi. Basis data dapat diperlakukan dari dua sudut pandang yaitu secara fisik dan secara logis. Basis data secara fisik berupa media untuk menyimpan data, seperti kartu buku besar, pita magnetik, disk, disket, kaset, kartu magnetik, *chip*, dan *microfilm*. Basis data secara fisik merupakan tempat yang sesungguhnya suatu data disimpan. Namun yang lebih penting bukan dalam bentuk fisik apa data disimpan, melainkan bagaimana mencari, menggabungkan, dan mengambil data yang disimpan untuk memenuhi kebutuhan khusus pemakai. Oleh karena itu, basis data dapat dipandang dari sudut pandang logis yang bersangkutan dengan bagaimana struktur penyimpanan data sehingga menjamin ketepatan, ketelitian, dan relevansi pengambilan informasi untuk memenuhi kebutuhan pemakai.

6. Blok Pengendalian (*Control Block*)

Semua sistem informasi harus dilindungi dari bencana dan ancaman, seperti bencana alam, api, kecurangan, kegagalan sistem, kesalahan dan penggelapan, penyalahgunaan, ketidakefisienan, sabotase, orang-orang yang dibayar untuk melakukan kejahatan. Beberapa cara yang perlu dirancang untuk menjamin perlindungan, integritas, dan kelancaran jalannya sistem informasi adalah :

- a. Penggunaan sistem pengelolaan catatan.
- b. Penerapan pengendalian akuntansi.
- c. Pengembangan rancangan induk sistem informasi.
- d. Pembuatan rencana darurat dalam hal sistem informasi gagal menjalankan fungsinya.
- e. Penerapan prosedur seleksi karyawan.
- f. Pembuatan dokumentasi lengkap tentang sistem informasi yang digunakan oleh perusahaan.
- g. Perlindungan dari bencana api dan putusnya aliran listrik.
- h. Pembuatan sistem penunjang untuk mengantisipasi kegagalan sistem informasi yang sekarang digunakan dan pembuatan tempat penyimpanan data di luar perusahaan sebagai cadangan (*backup*).
- i. Pembuatan prosedur pengamatan dan penggunaan alat-alat pengamanan serta

pengendalian akses ke dalam sistem informasi.

Terlepas dari organisasi yang dilayani oleh sistem informasi atau cara yang dipakai untuk merancang dan mengembangkan sistem informasi, setiap sistem informasi selalu terdiri dari enam blok bangunan seperti yang disebutkan di atas.

Sistem informasi terdiri dari bermacam-macam jenisnya, salah satunya adalah sistem informasi yang digunakan untuk membangun atau mengembangkan sebuah web atau yang disebut sistem informasi berbasis web.

2.2 Sistem Informasi Berbasis Web

Sistem informasi berbasis web secara umum digunakan sebagai sistem informasi yang menjadi standar penyimpanan data, mendapatkan informasi, memformat data, serta menampilkan informasi via *client/server architecture*. Sistem informasi ini menangani seluruh tipe informasi digital meliputi *text, hypermedia, graphics*, dan suara yang sangat mudah digunakan oleh pengguna karena menerapkan *Graphical User Interface (GUI)*.

Sistem Informasi berbasis web sama seperti sistem informasi lainnya, bedanya terletak pada bahasa pemrograman yang digunakan yaitu bahasa pemrograman web dan juga sistem informasi berbasis web ini bersifat *online*. Adapun komponen-komponen yang menyusun sebuah sistem informasi berbasis web seperti pada tugas akhir ini sebagai berikut :

2.2.1 Internet

Internet mengimplementasikan suatu mekanisme untuk menghubungkan jaringan-jaringan komputer. Internet dapat dianggap sebagai lem yang merekatkan semua jaringan komputer yang ada di seluruh dunia, untuk membentuk satu jaringan tunggal komputer. Perkembangan internet telah membawa perubahan yang dramatis dalam hal keberadaan dan kegunaan jaringan komputer baik untuk komunikasi di dalam maupun antar organisasi.

Internet mengintegrasikan jaringan-jaringan komputer ini dengan mengimplementasikan protokol-protokol komunikasi standar yang digunakan jaringan komputer untuk berkomunikasi. Protokol komunikasi adalah sekumpulan peraturan yang mengatur komunikasi yang terjadi antar komputer. Peraturan ini termasuk mengatur tata cara pengkodean pesan, dan menentukan jenis pesan yang dapat dikirim dan pesan yang harus dikirim sebagai jawaban. Jaringan komputer yang berbeda menggunakan protokol yang berbeda pula. Internet mengimplementasikan suatu protokol standar yang disebut TCP/IP (*Transmission Control Protocol/Internet Protocol*) yang membuat jaringan komputer dapat berkomunikasi satu sama lain.

Internet menjadi penting karena internet menyediakan infrastruktur global yang siap digunakan dengan biaya rendah untuk melakukan komunikasi baik di dalam suatu badan organisasi maupun antar organisasi berbeda. Aplikasi internet

yang terkenal dan paling banyak digunakan, yaitu *electronic mail* atau *email*, menggambarkan kekuatan dan penyebaran luas penggunaan internet.

Teknik yang sama seperti yang digunakan oleh internet, contohnya protokol TCP/IP, sering juga digunakan di dalam organisasi untuk menciptakan jaringan pribadi perusahaan. Jaringan ini disebut dengan intranet. Demikian juga organisasi yang berbeda menciptakan jaringan pribadi untuk komunikasi antar organisasi yang juga menggunakan teknologi internet. Jaringan ini disebut dengan ekstranet. Keberadaan intranet dan ekstranet ini semakin menguatkan keberadaan teknologi internet itu sendiri.

2.2.2 World Wide Web (WWW)

World Wide Web (WWW) atau web adalah suatu aplikasi dari internet dimana diberikan cara mudah untuk mengakses informasi dan menjalankan program-program yang tersimpan pada komputer-komputer yang terhubung oleh internet. Web juga memiliki memori di dalamnya dimana informasi dapat disajikan, disimpan, dan diakses. Informasi yang tersimpan ini dapat dihubungkan satu sama lain melalui internet dalam bentuk *link*.

Web menggunakan internet dalam menyediakan fasilitas untuk :

- menyimpan informasi
- mencari dan memberikan informasi
- menyimpan dan mengeksekusi program-program komputer

- menerima masukan dan memanipulasi informasi.

Yang membuat fasilitas web tersebut menjadi sangat berguna dan revolusioner adalah kemampuan untuk memperoleh fasilitas tersebut dari lokasi manapun oleh siapapun dengan biaya sistem komputer dan sumber komunikasi yang rendah. Hal ini dikarenakan web dapat digunakan oleh siapa saja yang melakukan akses ke internet.

Informasi yang tersimpan di dalam web disebut dengan dokumen web dimana dokumen ini dapat berisi *link-link*. *Link* ini digunakan untuk mengalamatkan bagian lain dari dokumen atau suatu dokumen lain. Dengan akses informasi pada web yang sebagian besar tidak terikat aturan ketat, siapa saja yang melakukan akses ke internet dapat menambahkan dokumen apapun ke dalam web dan *me-link* atau menghubungkan dokumen tersebut ke dokumen lain yang ada. Sehingga web menciptakan sumber informasi yang luas dan terhubung satu sama lain (*inter-linked*), dimana siapapun dapat ikut berpartisipasi di dalamnya.

2.2.3 Web Server

Web menyediakan sarana untuk menyimpan dan mengakses informasi dan layanan yang dapat diakses dari *web client* menggunakan *web browser*. Ruang yang digunakan untuk menyimpan informasi dan layanan ini disebut dengan *web site*. *Website* mirip dengan penyimpan berkas konvensional tetapi dalam

hal ini yang disimpan adalah berkas yang berisi dokumen web dan sumber informasi lain. Dokumen yang disimpan ini umumnya berbentuk *form* atau berkas HTML, tetapi juga dapat berbentuk berkas teks sederhana (*plain text*), berkas *pdf* atau format dokumen lain yang biasa disimpan. Sumber informasi yang disimpan dapat berupa kode yang dapat dieksekusi, yang biasa disebut dengan skrip, yang berguna untuk mengakses program atau jenis server lain.

Web site ini diimplementasikan pada suatu *web server*. *Web server* juga bisa memfasilitasi klien untuk mengakses ke lingkungan luar, seperti ke suatu sistem basis data, dengan menjalankan program yang disebut *gateway*. Mekanisme dimana suatu *web server* mengakses suatu *gateway* disebut *Common Gateway Interface (CGI)*. Dengan kata lain *web server* akan menyediakan suatu rute atau *gateway* untuk dapat mengakses informasi yang tersimpan di sistem lain.

Web server dijalankan di komputer induk dan menunggu *HTTP request* masuk. *Web server* bisa dianggap sebagai pihak yang menyediakan informasi dan layanan yang akan diakses oleh klien melalui web.

2.2.4 Web Browser

Web Browser digunakan untuk mengakses dan menampilkan dokumen web atau halaman *World Wide Web (WWW)* dan mengikuti link yang terdapat di

dalamnya untuk mengakses dokumen lain atau untuk berpindah antara bagian-bagian lain di dalam dokumen web tersebut. Dokumen yang ditampilkan bisa disimpan di suatu *website* sebagai suatu dokumen web, atau dokumen tersebut dibentuk dengan mengeksekusi skrip yang tersimpan di server.

Suatu *web server* dapat diakses secara bersamaan oleh banyak *browser* sekaligus. Koneksi antara satu *browser* dan satu server hanya akan bertahan untuk satu pertukaran pesan saja. Setelah *browser* menerima dokumen web yang diinginkan, koneksi akan diputuskan dan harus diciptakan kembali jika *browser* ingin menerima dokumen web yang lain. Oleh karena itu server dibuat secara sederhana saja karena tidak dibutuhkan pemeliharaan saluran komunikasi yang banyak. Juga tidak diciptakan memori ketika suatu koneksi antara *browser* dan server dibangun. Hal ini berarti harus dilakukan langkah lain oleh *browser* dan server agar dapat mendukung transaksi yang melibatkan banyak interaksi antara *browser* dan server.

Beberapa contoh *web browser* adalah *Netscape Navigator*, *Microsoft Internet Explorer*, *Opera*, dan *Mozilla Firefox*. *Web Browser* yang disebut juga dengan *web client* memberikan kemudahan navigasi dari halaman web pada internet dan menampilkan dokumen yang ditulis dalam bahasa HTML. Dengan kata lain *web browser* adalah sebuah perangkat lunak untuk menerjemahkan kode-kode HTML dari dokumen menjadi halaman web yang terdiri dari font

berwarna, garis, perataan text, dan *style* lainnya sesuai dengan dokumen HTML.

2.2.5 *HyperText Transfer Protocol (HTTP)*

HyperText Transfer Protocol (HTTP) adalah protokol komunikasi yang dibangun dan digunakan dalam web ketika terjadi koneksi antara *browser* dan *server*.

Secara umum pelayanan kepada *web client* adalah pelayanan *HyperText Transfer Protocol (HTTP)* yang memberikan akses ke tempat penyimpanan dokumen HTML. Dengan kata lain *browser* dan *web server* berkomunikasi dengan HTTP.

Dalam buku *Web Database Systems* karya Barry Eaglestone dan Mick Ridley tertulis cara kerja HTTP adalah sebagai berikut :

- a. *Client* mengontak *server* dengan melakukan permintaan akan suatu dokumen web. Permintaan ini terdiri dari suatu metode yang digunakan, alamat dokumen yang diminta, dan versi HTTP yang digunakan. Dua macam metode yang biasa digunakan adalah metode GET dan POST. Kedua metode ini pada dasarnya memiliki prinsip yang sama dalam proses *request*, bedanya adalah pada metode GET parameter yang digunakan tergabung di dalam *querystring* sedangkan pada metode POST parameter terpisah-pisah. Dengan kata lain pada metode POST ditangani lebih banyak parameter.
- b. *Client* kemudian mengirimkan *header information* yang berisi keterangan tentang pihak klien

terdiri dari *User-Agent header* (jenis browser dan sistem operasi yang digunakan) serta tipe dokumen yang dapat diterima oleh klien.

- c. *Client* lalu dapat mengirimkan informasi tambahan, misalnya parameter-parameter yang dikirimkan ke skrip pada server melalui metode POST. Pada metode GET, informasi ini dikirimkan pada bagian pencarian dari URL. Sampai pada tahap ini, klien telah menyelesaikan fase protokol pertama. Tanggung jawab proses sekarang beralih ke server yang harus melakukan tanggapan terhadap permintaan yang dikirimkan klien.
- d. Pertama-tama server menanggapi dengan mengirimkan status yang berisi versi HTTP dan kode status, apakah *request* berhasil atau tidak. Seringkali status ini dikirimkan ke klien tanpa pengguna menyadarinya. Kode yang mungkin dikenali oleh pengguna antara lain kode kesalahan seperti "404 Not Found".
- e. Sama halnya dengan klien yang melakukan pengiriman informasi *header*, server juga melakukan pengiriman informasi *header server* yang berisi tipe server, format dokumen yang dikirimkan, ukuran dokumen, dan sebagainya.
- f. Ketika permintaan berhasil dilakukan, maka server akan mengirimkan data yang diminta tersebut.

2.2.6 *HyperText Mark Up Language (HTML)*

HyperText Mark Up Language (HTML) adalah bahasa pemrograman yang digunakan untuk membuat suatu halaman web atau yang biasa disebut sebagai dokumen HTML. Dokumen web berformat HTML ini ditransfer dari web server ke web browser menggunakan *HyperText Transfer Protocol (HTTP)* yang merupakan protokol untuk mentransferkan data dari web server ke web browser.

Agar file teks dalam dokumen HTML dapat ditampilkan sebagai suatu halaman web maka di dalamnya ditambahkan perintah-perintah HTML yang dituliskan di dalam tanda kurung < dan >. Pasangan dari sebuah tag ditandai dengan tanda /, misalnya pasangan tag <HTML> adalah </HTML>.

Dengan perintah/tag HTML ini dapat dilakukan berbagai fungsi antara lain :

- a. Menentukan jenis font yang digunakan beserta ukurannya.
- b. Mengkombinasikan antara teks dan gambar.
- c. Membuat link ke halaman web atau situs lainnya.
- d. Membuat suatu form interaktif.

2.2.7 *Web Database*

Sistem basis data dapat dikoneksikan ke internet untuk digunakan melalui web sehingga dapat diakses dari berbagai tempat secara jarak jauh, tetapi juga dapat diintegrasikan dengan

sistem lain. Beberapa tipe koneksi yang dapat digunakan antara lain :

- *Remote connections* atau koneksi jarak jauh : sistem basis data yang dapat diakses melalui web menjadi dapat digunakan dari belahan bumi manapun.
- *Client-server architectures* atau arsitektur klien-server : sudah merupakan hal yang biasa jika program aplikasi basis data dan basis data itu sendiri dijalankan di komputer yang berbeda. Aplikasi dan basis data ini dapat berkomunikasi melalui internet. Keuntungannya adalah tiap komputer dapat dikonfigurasi dengan peranan tersendiri, seperti sebagai penyimpan data, komputer grafis, komputasi, dll.
- *Distributed databases* atau basis data terdistribusi : beberapa DBMS memiliki fasilitas yang memungkinkan bagian-bagian berbeda dari suatu basis data untuk disimpan di komputer yang berbeda. Data didistribusikan dengan cara-cara tertentu sehingga pengguna tidak menyadarinya. Keuntungannya adalah data dapat disimpan dimana data tersebut digunakan.
- *Multidatabases* atau basis data majemuk : ada teknologi penyatuan basis data yang disebut *multidatabase technology* yang memungkinkan sejumlah basis data yang berdiri sendiri dapat dikombinasikan sehingga mampu menyediakan akses terpadu ke data yang disimpan.

Penggunaan basis data dan teknologi web di dalam suatu sistem tunggal memiliki keuntungan dalam hal sebagai berikut :

- Akses yang lebih luas ke basis data. Dengan mengkoneksikan sistem basis data ke web akan tercipta kemungkinan akses ke basis data dari seluruh dunia.
- Distribusi sistem. Web memungkinkan suatu basis data dan aplikasinya untuk didistribusikan. Data dapat disimpan dimana data tersebut dipakai dan/atau aplikasi dapat diletakkan dimana aktivitas yang memerlukannya terjadi. Keuntungannya adalah kemampuan untuk mengkonfigurasi komputer dengan tugas-tugas yang lebih spesifik.
- Proses *querying*, manipulasi, dan administrasi dari data web menjadi lebih baik. DBMS menyediakan sarana untuk *querying*, manipulasi, dan administrasi data. Dengan demikian aplikasi web yang harus menggunakan sejumlah besar data akan memperoleh keuntungan dari menggunakan DBMS untuk menyimpan data.

2.3 Basis Data

Keberhasilan suatu sistem informasi sangat dipengaruhi oleh sistem basis data yang merupakan salah satu elemen penyusun sistem tersebut. Semakin lengkap, akurat, dan mudah dalam menampilkan kembali data-data yang termuat dalam

sistem basis data akan semakin meningkatkan kualitas sistem informasi tersebut.

Ada banyak pendapat mengenai pengertian basis data. Dalam buku *Sistem Basis Data Konsep dan Peranannya dalam Sistem Informasi Manajemen* yang ditulis oleh Edhy Sutanta, terdapat terjemahan pengertian basis data dari James Martin dalam buku berjudul *Database Organization* sebagai berikut :

"Basis data adalah suatu kumpulan data terhubung (*interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa mengatap satu sama lain atau tidak perlu suatu kerangkapan data (*controlled redundancy*) dengan cara-cara tertentu sehingga mudah untuk digunakan atau ditampilkan kembali; dapat digunakan oleh satu atau lebih program aplikasi secara optimal; data disimpan tanpa mengalami ketergantungan pada program yang akan menggunakannya; data disimpan sedemikian rupa sehingga penambahan, pengambilan, dan modifikasi data dapat dilakukan dengan mudah dan terkontrol."

Dari pengertian tersebut dapat disimpulkan bahwa basis data mempunyai beberapa kriteria penting, yaitu :

1. bersifat *data oriented* bukan *program oriented*.
2. dapat digunakan oleh beberapa program aplikasi tanpa perlu mengubah basis datanya.
3. dapat berkembang dengan mudah, baik volume maupun strukturnya.

4. dapat memenuhi kebutuhan sistem-sistem baru secara mudah.
5. dapat digunakan dengan cara-cara yang berbeda.
6. kerangkapan data (*data redundancy*) minimal.

Menurut James Martin dalam bukunya yang berjudul *Database Organization*, ada beberapa hal yang harus diperhatikan pada *file* basis data agar dapat memenuhi kriteria sebagai suatu basis data. Hal-hal tersebut berhubungan dengan masalah berikut :

1. *data redundancy*, yaitu munculnya data-data yang sama secara berulang-ulang pada *file* basis data yang semestinya tidak diperlukan.
2. *data inconsistency*, yaitu munculnya data yang tidak konsisten pada medan yang sama untuk beberapa *file* dengan kunci yang sama.
3. *data terisolasi*, disebabkan oleh pemakaian beberapa basis data. Program aplikasi yang digunakan tidak dapat mengakses *file* tertentu dalam sistem basis data tersebut, kecuali bila program aplikasi diubah/ditambah sehingga seolah-olah ada *file* yang terpisah/terisolasi terhadap *file* yang lain.
4. *security problem*, berhubungan dengan masalah keamanan data dalam sistem basis data. Harus ada perlindungan terhadap basis data agar hanya pengguna yang berwenang dapat melakukan akses, serta basis data harus dilindungi dari

kerusakan akibat bencana alam atau bahaya lainnya.

5. *integrity problem*, berhubungan dengan unjuk kerja sistem agar dapat melakukan kendali/kontrol pada semua bagian sistem sehingga sistem selalu beroperasi dalam pengendalian yang penuh.

Database Management System (DBMS) merupakan salah satu elemen dalam basis data. Sesuai dengan pernyataan James F. Courtney Jr. dan David B. Paradiance dalam buku berjudul *Database System for Management*, DBMS adalah perangkat lunak yang memberikan fasilitas (yang tersedia dan dapat digunakan) untuk melakukan fungsi pengaturan, pengawasan, pengendalian/kontrol, pengolahan, dan koordinasi terhadap semua proses/operasi yang terjadi pada sistem basis data.

Relational data model atau model data relasional menjelaskan kepada pengguna tentang hubungan logik antardata dalam basis data dengan memvisualisasikannya ke dalam bentuk tabel-tabel yang terdiri dari sejumlah baris dan kolom yang menunjukkan atribut tertentu. Relasi dalam model basis data relasional mempunyai beberapa karakteristik yang harus dipenuhi, yaitu sebagai berikut :

1. semua *entry*/elemen data pada suatu baris dan kolom tertentu harus mempunyai nilai tunggal (*single value*), atau suatu nilai yang tidak

dapat dibagi lagi (*atomic value*), bukan suatu larik atau grup perulangan.

2. semua entry/elemen data pada suatu kolom tertentu dalam relasi yang sama harus mempunyai jenis yang sama.
3. masing-masing kolom dalam suatu relasi mempunyai suatu nama yang unik (meskipun kolom-kolom dalam relasi yang berbeda diijinkan mempunyai nama yang sama).
4. pada suatu relasi/tabel yang sama tidak ada dua baris yang identik.

Dalam buku *Barry Eaglestone* dan *Mick Ridley* yang berjudul *Web Database Systems*, keuntungan dari model data relasional antara lain adalah

- menggambarkan data secara logik sehingga terlihat tidak sekompleks model data biasa yang merujuk ke penampilan data secara fisik.
- sederhana, model ini memiliki struktur dan tabel yang sederhana serta mudah dalam mengambil dan memanipulasi data.
- model relasional merupakan model yang terdefinisi secara matematis sehingga sesuai dengan teori basis data juga sistem basis data aktual.
- berorientasi pada *set* (sekelompok *record*) sehingga beroperasi dengan banyak data pada suatu waktu.

Agar suatu data mentah dari dunia luar dapat dijadikan basis data dengan model relasioanl, maka

diperlukan suatu proses normalisasi. Proses normalisasi adalah sebuah teknik untuk menghasilkan himpunan relasi dengan sifat-sifat yang diinginkan yaitu bersifat *reversible*, berarti hasil dari normalisasi selalu dapat dikembalikan ke keadaan semula.

2.4 .NET

Microsoft telah melakukan suatu revolusi dalam dunia pemrograman untuk aplikasi berbasis Windows sekaligus menjadi pemain utama dalam pengembangan aplikasi web dengan memperkenalkan .NET Framework dan Visual Studio .NET. Kedua penemuan baru ini membawa perubahan-perubahan yang signifikan bagi pengembangan program yang menggunakan Visual Basic. Sebagai satu kesatuan Microsoft.NET adalah perubahan paling penting pada *platform* pengembangan Microsoft sejak pergeseran dari 16-bit ke 32-bit. Bukan hanya mengimplementasikan bahasa pemrograman berorientasi obyek, .NET juga membawa penemuan hebat dalam hal pengembangan proyek yang kompatibel untuk berbagai *platform*.

Dua bagian penting dari .NET adalah *Microsoft .NET Framework Software Development Kit (SDK)* dan *Visual Studio Integrated Development Environment (IDE)*. IDE digunakan untuk mengembangkan program dan Framework digunakan untuk menjalankan program yang telah dibuat.

.NET Framework adalah teknologi mendasar untuk pengembangan ASP.NET. Teknologi ini menyediakan suatu *platform* untuk mengembangkan dan menjalankan aplikasi dan XML Web Services yang ditulis dalam berbagai macam bahasa pemrograman pada berbagai jenis *platform*. Framework ini tersusun atas *Common Language Runtime* (CLR) atau *runtime* bahasa umum, *class libraries* atau pustaka kelas, dan ASP.NET yang merupakan versi terbaru dari *Active Server Pages* (ASP).

Common Language Runtime (CLR) menyediakan lingkungan yang mengatur pengeksekusian kode program. CLR menyediakan layanan untuk tugas-tugas seperti mengintegrasikan komponen-komponen yang dikembangkan dalam berbagai bahasa, menangani kesalahan dalam berbagai bahasa, menangani keamanan, dan mengatur penyimpanan dan penghapusan obyek-obyek pemrograman.

Kelas-kelas dan antarmuka yang menjadi bagian dari bahasa pemrograman .NET disimpan di dalam suatu *library* atau pustaka yang dikenal dengan *.NET Framework class library*. *Library* ini dibagi menjadi bagian-bagian atau kelompok-kelompok yang disebut dengan *namespace*. Setiap *namespace* terdiri dari kelas-kelas, struktur, enumerasi, delegasi, dan/atau antarmuka-antarmuka yang dapat digunakan dalam program. Contoh *namespace* antara lain *System* yang berisi kelas-kelas dasar dan fundamental untuk tipe-tipe data, event, dan event handler, *namespace System.Data* yang berisi arsitektur ADO.NET, *System.IO* yang berisi tipe untuk membaca

dan menulis file-file maupun menampilkan ke layar, dan lain sebagainya.

2.4.1 ASP.NET

Bagian lain dari .NET yang patut diperhitungkan adalah peningkatan di dalam pengembangan aplikasi web. ASP.NET merupakan penerus dari Active Server Pages (ASP). ASP.NET tidak sekedar versi *upgrade* dari ASP. ASP.NET menyediakan *platform* pengembangan web terdepan yang diciptakan dewasa ini. Terlebih lagi, ASP.NET dibangun dari dasar untuk membuat infrastruktur pengembangan web yang sama sekali baru dan lebih fleksibel. Yang membuat ASP.NET menjadi sebuah revolusi ialah pembuatannya yang didasarkan pada *platform* baru Microsoft.NET, atau lebih tepatnya .NET Framework.

ASP.NET merupakan perangkat pengembang web yang mampu mengkompilasi aplikasi yang ditulis dalam berbagai bahasa pemrograman yang mendukung kompatibilitas .NET termasuk Visual Basic. Hal ini berarti keunggulan dari CLR dan kode terkelola (*managed code*) berlaku pula untuk pengembangan aplikasi web. ASP.NET menjadikan pengembangan web lebih mudah dengan menyediakan dukungan *debugging* yang sama untuk *Web Forms* dan *Web Services* seperti halnya pada aplikasi Windows.

Pada arsitektur ASP.NET terdapat banyak perbaikan dan fitur-fitur baru. Walaupun demikian tetap terdapat banyak hal yang tidak berbeda dari fitur pada ASP. Obyek *Response* dan *Request* tetap

ada, seperti juga obyek-obyek Aplikasi, Sesi, dan Server, meskipun dengan metode dan properti yang baru. ASP.NET dirancang untuk jalan berdampingan dengan ASP klasik sehingga sambil bekerja dengan ASP.NET yang baru, aplikasi ASP yang ada masih bisa dijalankan.

Sesuai dengan yang tertulis di buku *Microsoft ASP.NET Step By Step* yang ditulis oleh G. Andrew Duthie, beberapa fitur baru ASP.NET antara lain :

- **Web Forms**

Merupakan model pemrograman yang baru pada Web Form ASP.NET yang menggabungkan aplikasi yang terbaik pada ASP dengan kemudahan pengembangan dan produktivitas Visual Basic.

- **Kontrol Server**

Merupakan komponen besar pada model pemrograman Web Form. Kontrol server ASP.NET memetakan ke elemen HTML (dan kontrol tambahan lainnya) serta menyediakan kemampuan memrogram *server-side* yang handal. Kontrol server dijalankan pada server dan bisa membuat output HTML yang dirancang untuk browser tingkat tinggi, seperti Internet Explorer 5.x atau yang lebih baru, atau untuk semua browser yang sesuai dengan HTML 3.2.

- **Layanan Web**

Merupakan kunci ASP.NET yang memungkinkan pengembang membuat layanan program yang bisa dipakai pengembang lain melalui internet (atau *local internet*). Layanan web didasarkan pada tumbuhnya standar SOAP (*Simple Object*

Access Protocol), sehingga memungkinkan hubungan yang mudah di antara operasi lewat *platform* yang berbeda.

- **Caching**

ASP.NET memiliki mesin *caching* yang baru dan handal yang memungkinkan pengembang meningkatkan kinerja aplikasi mereka dengan mengurangi beban pemrosesan server web dan server basis data.

- **Perbaikan konfigurasi**

ASP.NET menggunakan metode baru penyimpanan informasi konfigurasi untuk aplikasi web. Sebagai pengganti memerintah IIS menyimpan informasi ini dalam basis data yang sulit diakses, informasi ini disimpan dalam file konfigurasi berbasis XML yang bisa dibaca oleh mesin maupun pengembang.

- **Perbaikan Pengelolaan Status**

Pada ASP klasik terdapat banyak keterbatasan dalam mengelola status. ASP.NET mengatasi keterbatasan ini dengan menyediakan dukungan pendistribusian status sesi dalam server web, menaruh informasi status dalam basis data SQL Server, serta menyediakan pengelolaan status tanpa menggunakan *cookies*.

- **Pengamanan**

Fitur ini merupakan fitur yang terpenting dalam aplikasi web saat ini. Model pengamanan ASP.NET telah mengalami perbaikan besar, termasuk metode otentikasi yang baru dan

lebih baik, pengamanan akses kode, serta otorisasi berdasarkan peran.

- **Perbaikan kehandalan**

ASP.NET berisi fitur baru ini untuk meningkatkan kehandalan aplikasi web, termasuk restart aplikasi yang proaktif serta proses otomatis daur ulang untuk mengatasi kondisi *deadlock* dan kebocoran memori.

2.4.2 ADO.NET

Kemampuan menyimpan dan mengakses data adalah inti dari kebanyakan aplikasi web, dan ini bisa dibuat dengan mudah memakai ASP klasik, para pengembang bisa melakukannya melalui *ActiveX Data Objects* (ADO) pada komponen COM. Platform .NET menyediakan serangkaian kelas bernama ADO.NET yang merupakan kelanjutan logis dari ADO, meskipun obyek yang mendasarinya telah mengalami perubahan besar. Sedangkan ASP.NET mengekspos serangkaian kontrol yang terikat data yang berintegrasi dengan ADO.NET untuk menyediakan layanan pengikatan data.

Dalam ASP klasik, agar dapat mengakses data pengembang menggunakan obyek *Connection* pada ADO untuk melakukan koneksi ke sebuah basis data, dan kemudian memakai obyek *Command* dan *Recordset* untuk memanggil, mengubah dan meng-update data.

Arsitektur kelas ADO.NET dibuat agak berbeda dari ADO klasik. Kelas-kelas pada ADO.NET dibagi ke dalam dua kategori : spesifik-sumberdata dan non-spesifik-sumberdata.

Kelas-kelas yang spesifik untuk sumberdata tertentu dikatakan bekerja spesifik dengan *NET Data Provider*. *NET Data Provider* adalah serangkaian kelas yang memungkinkan kode yang teratur untuk berinteraksi dengan sumberdata spesifik untuk memanggil, meng-update, dan mengubah data. ADO.NET dilengkapi dengan dua *NET Data Provider* yaitu *SQL Server NET Data Provider*, yang menyediakan akses optimal ke basis data *SQL Server*, dan *OLE DB .NET Data Provider*, yang memungkinkan untuk melakukan koneksi ke semua sumberdata.

Kelas-kelas utama dalam *SQL Server NET Data Provider* dan *OLE DB .NET Data Provider* yang merupakan kelas spesifik-sumberdata adalah :

- Pada *SQL Server NET Data Provider* yaitu kelas ***SqlConnection***, pada *OLE DB .NET Data Provider* yaitu kelas ***OleDbConnection***, yang digunakan untuk membangun koneksi ke suatu basis data *SQL Server*.
- Pada *SQL Server NET Data Provider* yaitu kelas ***SqlCommand***, pada *OLE DB .NET Data Provider* yaitu kelas ***OleDbCommand***, digunakan untuk mengeksekusi pernyataan *SQL* atau menyimpan prosedur ke basis data *SQL Server*.
- Pada *SQL Server NET Data Provider* yaitu kelas ***SqlDataReader***, pada *OLE DB .NET Data Provider* yaitu kelas ***OleDbDataReader***, yang menyediakan akses *forward-only, read-only* ke serangkaian baris yang dihasilkan dari basis data *SQL Server*.

- Pada SQL Server NET Data Provider yaitu kelas ***SqlDataAdapter***, pada OLE DB .NET Data Provider yaitu kelas ***OleDbDataAdapter***, yang dipakai sebagai jembatan antara kelas *DataSet* dengan SQL Server.

Kelas utama yang tidak spesifik ke sumberdata adalah kelas *DataSet*. Kelas ini merupakan representasi satu atau lebih tabel data dalam memori. Data ini bisa dibaca dari aliran atau file XML, atau bisa juga hasil dari query ke sumberdata.

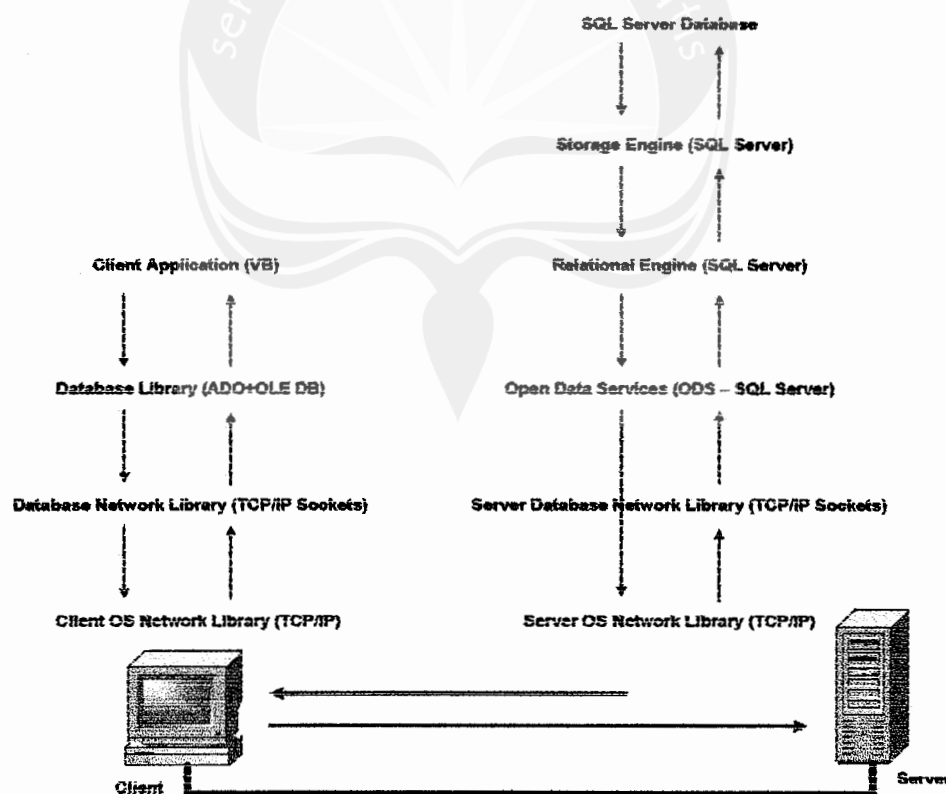
DataSet berisi sekumpulan tabel, yang masing-masing menyimpan koleksi baris, koleksi kolom dan koleksi batasan. Koleksi-koleksi ini dapat digunakan untuk mendapatkan informasi tentang obyek yang ada di dalamnya, serta dapat mengakses dan meng-update setiap nilai data. *DataSet* juga dapat menyimpan koleksi relasi antar tabel di dalamnya, sehingga memungkinkan data hierarkis disajikan.

Satu hal penting tentang kelas *DataSet* adalah bahwa kelas ini sama sekali tidak tahu tentang sumberdata yang datanya diterima, kelas ini hanya tahu apa datanya dan tipe kolom data. Artinya dataset dipisahkan dari sumberdata *back-end*. Hal ini mempermudah pengelolaan aplikasi web karena basis data *back-end* dapat diubah tanpa melakukan perubahan dalam halaman *web forms*.

2.5 SQL Server 2000

Data mempunyai nilai jika data tersebut akurat dan cukup komprehensif untuk memenuhi kebutuhan bisnis. Cara penyimpanan data dan mekanisme yang tersedia untuk menerima data merupakan faktor penting untuk dipertimbangkan. Sistem manajemen basis data menyediakan sistem penyimpanan data yang terpercaya dan alat penerimaan data yang fleksibel. Microsoft SQL Server 2000 merupakan salah satu sistem manajemen basis data yang baru dan berkemampuan lebih.

SQL Server adalah suatu sistem manajemen basis data relasional berbasis *client server*. Berikut arsitektur SQL Server :



Gambar 2.1 Akses client ke basis data SQL Server
(Sumber : Guerrero, Fernando, G., dan Rojas, Carlos, E., 2001)

Layanan Microsoft SQL Server Service terdiri dari komponen-komponen berbeda yang berkolaborasi untuk menyediakan layanan data. Terdapat tiga komponen utama yaitu :

- **Open Data Services**
Komponen ini menerima permintaan dari klien dari jaringan dan melewatkannya ke SQL Server. Setelah SQL Server menyelesaikan proses *query*, maka dikirimkan hasilnya ke ODS untuk ditransfer melalui jaringan ke aplikasi klien.
- **Relational Engine**
Komponen ini bertanggung jawab dalam hal *parsing*, optimasi, eksekusi *query*, dan keamanan.
- **Storage Engine**
Komponen ini mengatur operasi penyimpanan fisik, seperti penyimpanan data, alokasi dan dealokasi *data pages*, menciptakan log transaksi dan pemulihan, *backup* dan *restore* basis data, penguncian, dsb.

SQL Server 2000 pada dasarnya merupakan kumpulan dari beberapa layanan Windows berikut :

- **Layanan Microsoft SQL Server (MSSQLServer)**
Merupakan layanan utama, bertanggungjawab mengenai penyimpanan, integritas, konsistensi, konkurensi, keamanan data, pemrosesan *query*, optimisasi, dan eksekusi.
- **Microsoft SQL Server Agent (SQLServerAgent)**

Merupakan layanan yang penting dalam SQL Server Administration karena terdapat banyak operasi administratif yang bergantung pada SQLServerAgent untuk dieksekusi secara otomatis pada interval tertentu, seperti *backup*, pemeriksaan konsistensi data, dsb. Menangani penjadwalan pekerjaan, pengaturan sinyal, dan operator pemberitahuan.

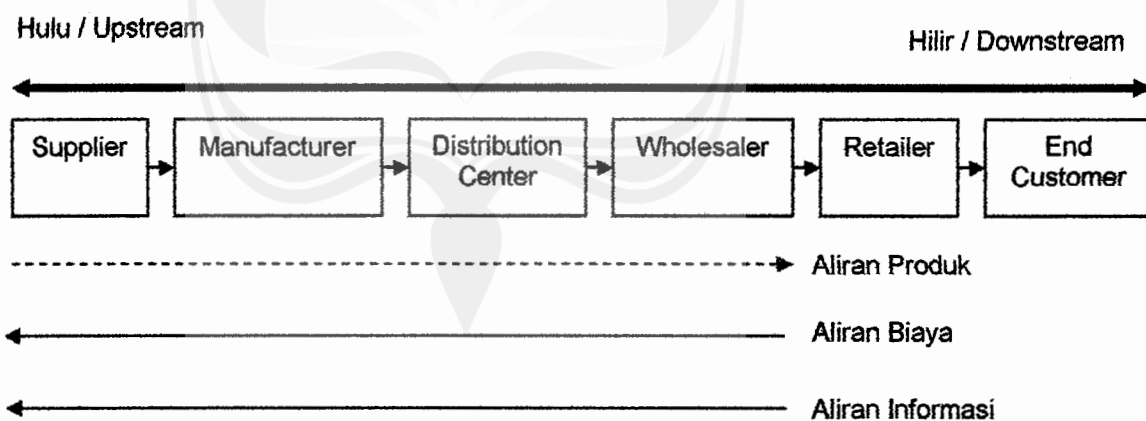
- *Microsoft Search*
Menyediakan kemampuan pencarian berbasis text bagi SQL Server, sama halnya bagi Microsoft Exchange dan Index Server.
- *Microsoft SQL Server OLAP Service*
Menyediakan *back-end supports* untuk Analysis Service.
- *Microsoft Distributed Transaction Coordinator (MS-DTC)*
Mendukung transaksi yang terjadi pada multiserver dan situasi yang beragam.
- *Server Network Libraries*
SQL Server dapat menyimpan beberapa *libraries* jaringan dalam saat bersamaan, menunggu *queries* menjawab, dan menggunakan salah satu dari *libraries* tersebut untuk dikirimkan kepada pihak klien.

(Sumber : Guerrero, Fernando, G., dan Rojas, Carlos, E., 2001)

2.6 Manajemen Rantai Pasok

Manajemen Rantai Pasok atau *Supply Chain Management* atau yang biasa disingkat SCM adalah konsep atau mekanisme yang mengatur aliran material atau produk serta informasi dan layanan dari titik awal (pemasok) sampai ke titik konsumsi (konsumen), yang mana aturan ini digunakan untuk meningkatkan produktivitas total perusahaan dalam rantai suplai melalui optimalisasi waktu, lokasi dan aliran kuantitas bahan.

Proses produksi suatu produk perusahaan melibatkan banyak pihak baik di dalam maupun di luar perusahaan. Semua pihak yang terkait ini membentuk suatu rantai suplai yang bertujuan akhir mensukseskan keseluruhan proses produksi.

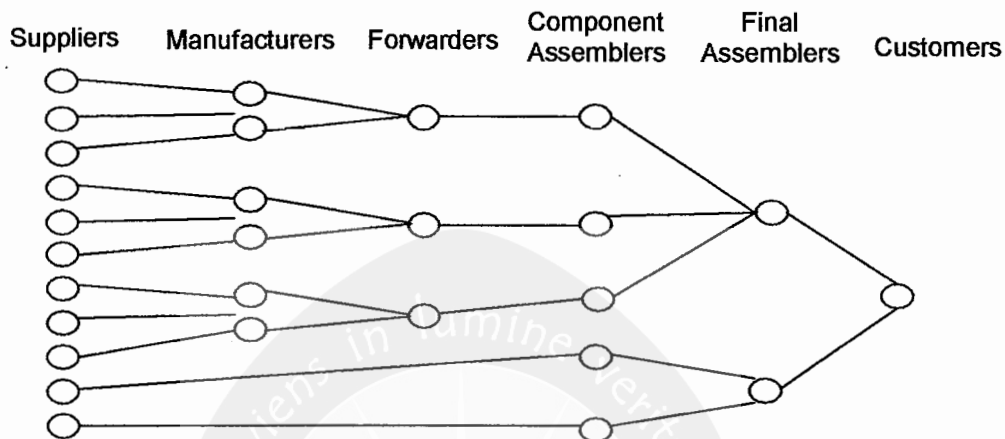


Gambar 2 Struktur Supply Chain yang disederhanakan

(Sumber : Yasrin Zabidi, 2001)

Struktur di atas merupakan struktur yang cukup sederhana. Peranan kunci dipegang oleh perusahaan manufaktur yang bertugas untuk memahami data penjualan yang ada dan menerjemahkannya dalam usaha mendapatkan order yang diinginkan.

Adapun sebuah struktur *Supply Chain* dapat berbentuk jauh lebih kompleks, dengan melibatkan banyak perusahaan manufaktur, banyak pemasok, distributor, atau pihak pemasok juga merupakan perusahaan manufaktur, dan lain sebagainya.



Gambar 3 Supply Web

(Sumber : SGV Bulletin, March-April 2001)

Struktur di atas jelas jauh lebih kompleks dimana jika perusahaan ingin menerapkannya maka perusahaan tersebut harus memantapkan terlebih dahulu penerapan rantai pasok secara sederhana, setelah meyakini efisiensi *supply chain* sederhana tersebut maka perusahaan harus mulai mencari rekan dan serikat kerja yang dapat mendukung, yang berarti akan berbagi keuntungan dan kerugian yang diperoleh nantinya.

SCM muncul dengan dilatarbelakangi oleh praktek tradisional dan perubahan lingkungan bisnis. Praktek tradisional tidak mementingkan adanya kerjasama antar pihak-pihak yang terkait dalam proses produksi untuk menghasilkan suatu produk. Masing-masing pihak bekerja dengan ukuran

keberhasilan sendiri. Pola-pola transaksi dan negosiasi yang terjadi mementingkan keberhasilan individual dan bukan mengacu ke kinerja keseluruhan pihak yang membentuk sebuah rantai suplai secara keseluruhan.

Dalam suatu *training module* mengenai *Supply Chain Management* yang disusun oleh *Multimedia Victoria*, SCM terdiri dari lima komponen dasar yaitu :

1. *Plan* : strategi-strategi pengendalian sumber daya untuk menghadapi permintaan konsumen akan barang dan jasa.
2. *Source* : seleksi pemasok yang mengirimkan bahan baku atau jasa untuk produksi barang dan jasa.
3. *Produce* : pembuatan barang dan jasa, termasuk proses produksi, pemeriksaan (*testing*), dan pengepakan (*packaging*).
4. *Deliver* : koordinasi order klien, pengiriman barang, serta penerimaan pembayaran.
5. *Return* : bagian untuk penerimaan produk cacat atau berlebih dari klien.

Konsep *Supply Chain Management* (SCM) dimungkinkan untuk dikembangkan dalam lingkup yang memanfaatkan perkembangan teknologi informasi yaitu teknologi internet. Teknologi internet memungkinkan aplikasi berbasis SCM dapat diakses oleh berbagai pihak dari berbagai tempat terpisah dan menawarkan integrasi aplikasi dan data yang

baik. SCM yang memanfaatkan keunggulan teknologi internet ini dikenal dengan istilah e-SCM.

Pada dasarnya e-SCM merupakan suatu konsep manajemen dimana perusahaan berusaha memanfaatkan internet dan teknologinya untuk mengintegrasikan seluruh mitra kerja perusahaan, terutama yang berhubungan dengan sistem pemasokan bahan-bahan atau sumber daya-sumber daya yang dibutuhkan dalam proses produksi (*sisi supply*).

Menurut Richardus Eko Indrajit dalam tulisannya *Extraprise Value Network*, konsep e-Supply Chain Management yang baik, biasanya memiliki 5 (lima) buah komponen yang saling mendukung, yaitu masing-masing :

1. **Supply Chain Replenishment** - adalah proses yang berkaitan dengan bagaimana para pemasok saling bekerja sama untuk menyediakan produk-produk atau bahan-bahan yang dibutuhkan oleh perusahaan sedemikian rupa sehingga memenuhi target permintaan dan *service level* yang dicanangkan.
2. **Collaborative Planning** - adalah proses yang memfokuskan diri pada aktivitas perencanaan yang berkaitan dengan operasi, produksi, inventori, dan distribusi sehingga keseluruhan perusahaan yang bekerja-sama mengetahui obyektivitasnya masing-masing sehingga untuk mencegah adanya konflik yang dapat bermuara pada tidak tercapainya kebutuhan pelanggan.

3. ***Collaborative Product Development*** - adalah proses yang berkaitan dengan aktivitas penciptaan produk atau jasa yang membutuhkan kerja-sama antara berbagai mitra bisnis tersebut dengan perusahaan, sehingga kualitas produk dan/atau jasa dapat terpenuhi sesuai dengan spesifikasi yang telah disepakati.
4. ***E-Procurement*** - pada dasarnya adalah manifestasi baru dari proses pengadaan konvensional, dimana dalam aktivitas ini teknologi internet dan prinsip-prinsip *e-business* benar-benar diterapkan secara sungguh-sungguh.
5. ***E-Logistics*** - sama seperti *e-Procurement*, hanya saja proses ini berkaitan dengan aktivitas manajemen pergudangan dan transportasi.

Adapun keuntungan dari *Supply Chain Management* adalah :

- ***Visibility and Management Collaboration***
(Transparansi dan Kolaborasi Manajemen)
Salah satu fitur kunci dari SCM adalah transparansi, dimana dimungkinkan penyebaran informasi antara pihak-pihak berbeda yang terkait dalam rantai pasok, yang merujuk kepada kerja sama atau kolaborasi dalam hal perencanaan pada proses primer dan sekunder.
- ***Cost Reduction and Quality Assurance***
(Pengurangan Biaya dan Asuransi Kualitas)

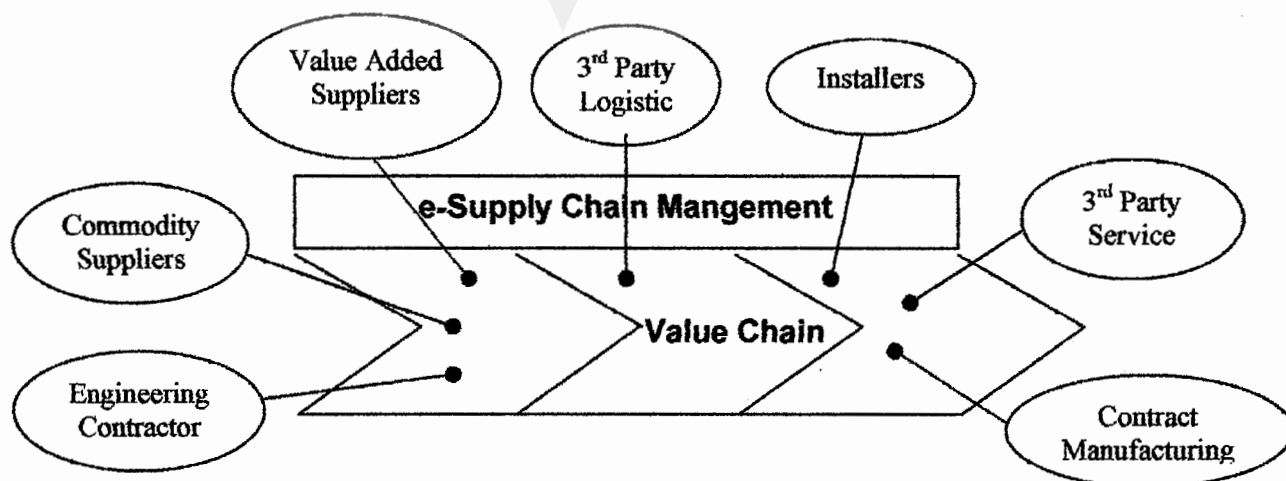
Eksekusi rantai pasok bertujuan untuk otomasi bisnis, yang berarti mengganti dan mengecilkan porsi kerja manual yang terlibat dalam proses bisnis. Pertukaran data bisnis yang terkomputerisasi di dalam rantai pasok dapat berarti otomasi *invoice*, otomasi proses order dan pembayaran. Dengan demikian, bisnis dapat menghemat waktu, meningkatkan pendapatan, mengurangi tingkat inventarisasi dan kesalahan manusia dalam pemrosesan data.

Seiring dengan menyebarnya konsep-konsep SCM di dunia industri, baik jasa maupun manufaktur, konsep-konsep yang lebih canggih yang merupakan pengembangan dari SCM banyak bermunculan. SCM merupakan suatu konsep yang sangat baik dan tampaknya menjadi keharusan bagi setiap perusahaan yang ingin bertahan, bukan bagi perusahaan yang ingin memimpin kompetisi di pasaran.

2.7 Manajemen Rantai Pasok dalam Sistem Informasi

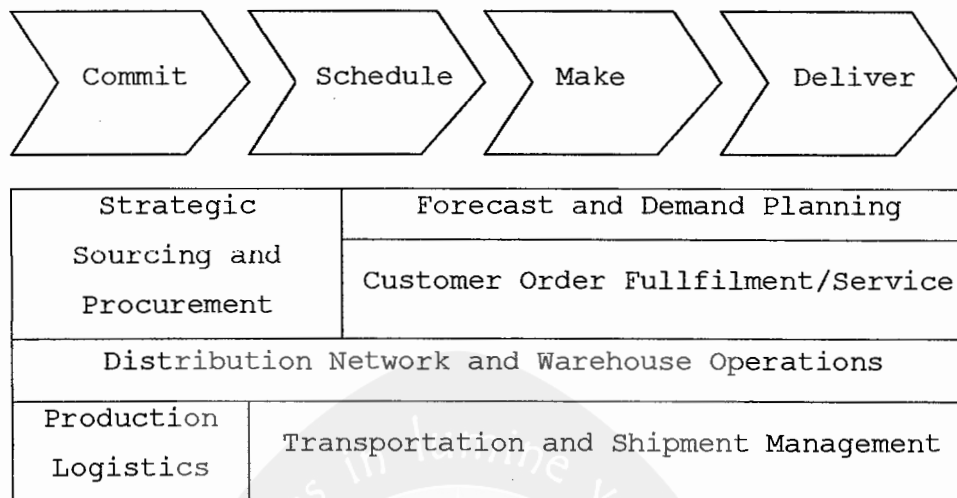
Konsep Manajemen Rantai Pasok yang ditarik ke dalam bentuk sistem informasi akan membentuk suatu sistem informasi yang menyediakan sarana-sarana pendukung keefisienan produksi dan kinerja perusahaan. Ditunjang dengan teknologi internet untuk membangun sistem informasi berbasis web bagi konsep Manajemen Rantai Pasok atau SCM ini, maka sistem informasi dapat digunakan untuk membangun ikatan secara elektronik dengan pemasok. Beberapa hal yang umumnya dapat difasilitasi oleh sistem seperti ini antara lain penempatan pesanan, mengikuti aliran pengantaran produk, memonitor kontrol kualitas, memberitahukan perubahan-perubahan rencana yang terjadi kepada rekan-rekan kerja, serta melakukan pembayaran.

Sesuai dengan konsep Manajemen Rantai Pasok dan konsep jaringan internet pada umumnya, beberapa pihak yang dapat terlibat dalam Manajemen Rantai Pasok berbasis internet dapat digambarkan sebagai berikut :



Gambar 2.4 e-Supply Chain Management
(Sumber : Martin V. Deise et al., 2000)

Proses-proses yang terjadi dalam rantai pasok sesuai dengan siklus hidup suatu Manajemen Rantai Pasok dapat digambarkan sebagai berikut :



Gambar 5 Supply Chain Life Cycle & SCM Functional Processes
(Sumber : James A. O'Brien, 2003)

Dari gambar di atas dapat dilihat bahwa suatu sistem informasi Manajemen Rantai Pasok seharusnya mampu untuk menangani perencanaan, pemesanan, pendistribusian, logistik produksi, serta pengendalian pengantaran.

Gambaran umum proses-proses atau fungsi-fungsi yang terdapat dalam suatu sistem informasi SCM berbasis web adalah sebagai berikut : pihak perusahaan melakukan perencanaan permintaan bahan baku berdasarkan data penjualan dan kebutuhan perusahaan dan mengalirkan informasinya ke sistem informasi SCM berbasis web. Dengan demikian setiap pemasok bahan baku tersebut dapat melihat kebutuhan permintaan perusahaan melalui internet. Jika perusahaan bekerja sama dengan *manufacturers* lainnya (*contract manufacturers*), maka perusahaan

manufaktur ini akan membandingkan permintaan perusahaan terhadap inventori dan kapasitas yang ada, lalu memasukkan komponen yang diperlukan ke dalam sistem. *Second-tier suppliers* (pemasok tingkat kedua) akan mengetahui dengan pasti kebutuhan tersebut dan akan mengirimkan apa yang harus mereka pasok ke perusahaan manufaktur ini.

Pada saat perusahaan memberitahukan informasi kebutuhan perusahaan secara online, dan sudah menjadi komitmen para pemasok untuk mengirimkan material, maka para pemasok ini akan menyebarkan informasi material tersebut kembali ke perusahaan melalui sistem rantai pasok yang ada. Dengan demikian perusahaan akan mendapatkan gambaran tentang aliran material/produk yang akan datang.

