

BAB II

LANDASAN TEORI

2.1 Definisi Sistem

Sistem adalah keseluruhan yang terdiri dari sejumlah variabel yang berinteraksi. Suatu sistem pada dasarnya adalah suatu susunan yang teratur dari kegiatan yang berhubungan satu sama lain dan prosedur-prosedur yang berkaitan yang melaksanakan dan memudahkan pelaksanaan kegiatan utama dari suatu organisasi (Prof. Komarudin, t.t.). Adapun syarat-syarat sistem sebagai berikut:

- Sistem harus dibentuk untuk menyelesaikan masalah.
- Elemen sistem harus mempunyai rencana yang ditetapkan.
- Adanya hubungan diantara elemen sistem.
- Unsur dasar dari proses (arus informasi, energi, dan material) lebih penting daripada elemen sistem.
- Tujuan organisasi lebih penting dari pada tujuan elemen.

2.2 Definisi Informasi

Informasi sebagai obyek materi ilmu komunikasi mempunyai makna: hal atau energiyang terpolakan yang mempengaruhi dan memungkinkan seseorang membuat keputusan dari beberapa kemungkinan yang ada. Informasi bermanfaat untuk mencapai tujuan ideal maupun material (Shannon dan Weaver, t.t.).

Bahwa semakin banyak informasi dapat mempengaruhi atau menambah pengetahuan seseorang dan dengan pengetahuan menimbulkan kesadaran yang akhirnya seseorang akan berperilaku sesuai dengan pengetahuan yang dimilikinya.

2.3 Definisi Sistem Informasi

Sistem informasi yaitu suatu sistem yang menyediakan informasi untuk manajemen dalam mengambil keputusan dan juga untuk menjalankan operasional perusahaan, di mana sistem tersebut merupakan kombinasi dari orang-orang, teknologi informasi dan prosedur-prosedur yang terorganisasi. Biasanya suatu perusahaan atau badan usaha menyediakan semacam informasi yang berguna bagi manajemen. Adapun komponen sistem informasi sebagai berikut:

- **Komponen input**
Input merupakan data yang masuk ke dalam sistem informasi.
- **Komponen model**
Kombinasi prosedur, logika, dan model matematika yang memproses data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.
- **Komponen output**
Output informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem
- **Komponen teknologi**
Teknologi merupakan alat dalam sistem informasi, teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan output dan memantu pengendalian sistem.
- **Komponen basis data**
Merupakan kumpulan data yang saling berhubungan yang tersimpan di dalam komputer dengan menggunakan software database.
- **Komponen kontrol**
Pengendalian yang dirancang untuk menanggulangi gangguan terhadap sistem informasi.

2.4 Sistem Informasi Eksekutif (EIS)

Menurut Kadir (2003,120) Sistem informasi eksekutif disebut sebagai sistem pendukung eksekutif. Sistem ini merupakan sistem informasi yang menyediakan fasilitas yang fleksibel bagi eksekutif dalam mengakses informasi eksternal dan internal yang berguna untuk mengidentifikasi masalah atau mengenali peluang. Pemakai yang awam dengan komputer pun tidak sulit mengoperasikannya karena sistem dilengkapi antarmuka yang sangat memudahkan pemakai untuk menggunakannya. EIS dapat membantu pihak eksekutif dalam pengambilan keputusan, dapat membuat *scheduling*, dan dalam membuat catatan-catatan penting.

Menurut Mcleod (2001) Sistem Informasi Eksekutif merupakan suatu sistem yang khusus dirancang bagi manajer tingkat perencanaan strategis yang menyediakan informasi bagi eksekutif mengenai kinerja keseluruhan perusahaan. Informasi dapat diambil dengan mudah dan dalam berbagai tingkat rincian. Informasi dapat ditampilkan dengan bentuk grafik, tabel, atau narasi.

Menurut Jogyanto (2003,339) Sistem Informasi Eksekutif adalah sistem informasi yang digunakan oleh manajer tingkat atas untuk membantu pemecahan masalah tidak terstruktur (*unstructured*), karena EIS mempunyai karakteristik yang khusus.

Adapun komponen dari Sistem Informasi Eksekutif (EIS) adalah sebagai berikut :

1. Perangkat keras (hardware)

Ketika membicarakan tentang perangkat keras untuk satu lingkungan EIS kita harus memfokuskan pada perangkat keras yang dibutuhkan pertemuan eksekutif. Eksekutif harus diletakan yang pertama dan kebutuhan eksekutif harus didefinisikan sebelum perangkat keras dapat terpilih. Perangkat keras komputer dasar diperlukan untuk EIS meliputi empat komponen:

- **Alat Masukan / input device**
Alat ini mengizinkan eksekutif untuk memasuki, verifikasi, dan pembaharuan data dengan seketika
- **Central Processing Unit**
adalah pusat komponen karena ini dapat mengontrol komponen mesin komputer yang lain.
- **File Penyimpanan Data**
Eksekutif dapat mempergunakan ini secara terpisah untuk menyimpan keterangan bisnis, dan bagian ini juga dapat membantu eksekutif mencari keterangan informasi bisnis historis dengan mudah.
- **Output Device / Alat keluaran**
Eksekutif dapat mempergunakan alat ini untuk membaca rekaman visual dan sistem ini memerlukan dukungan hardware komputer yang tidak begitu mahal. Alat ini juga dapat meningkatkan akses dari keterangan EIS untuk banyak pengguna yang lain dengan suatu perusahaan.

2. Software (perangkat lunak)

Memilih perangkat lunak penting untuk mendesain suatu EIS yang efektif. Oleh sebab itu, komponen perangkat lunak dan bagaimana cara mengintegrasikan data ke dalam suatu sistem sangatlah penting. Perangkat lunak dasar yang diperlukan suatu EIS meliputi empat komponen :

- **Teks**
Teks yang mendasari perangkat lunak. Bentuk paling umum dari teks dapat didokumentasikan.

- Database

Database heterogen bercokol pada suatu jangkauan spesifik vendor dan platform komputer membuka akses eksekutif bagi eksekutif.

- Dasar grafis

Grafis dapat mengarahkan volume dari teks dan statistik ke dalam keterangan visual untuk eksekutif. Jenis grafis yang khas adalah : bagan gugus berkala, diagram, peta, grafis gerak, bagan urutan, dan perbandingan mengorientasikan graf (bagan balok).

- Dasar model

EIS memodelkan data yang mengandung data statistik rutin dan khusus, keuangan, dan analisa kuantitatif lain.

3. Penggunaan interface

EIS membutuhkan efisiensi untuk mendapatkan kembali data relevan untuk pembuat keputusan, sehingga interface pemakai adalah sangat penting. Beberapa jenis pertemuan dapat tersedia di struktur `EIS`, seperti laporan terjadwal, soal atau jawab, pandu menu, bahasa perintah, bahasa alami, dan input atau output. Kalau eksekutif tidak nyaman dengan keterangan bertanya atau menjawab corak mode, EIS sepenuhnya dimanfaatkan. Alat penghubung ideal untuk satu EIS akan sederhana untuk mempergunakan dan sangat tinggi lentur, menyediakan kinerja konsisten, mencerminkan eksekutif dunia, dan mengandung keterangan pertolongan.

4. Telekomunikasi

Sebagai desentralisas sedang menjadi kecenderungan saat ini di perusahaan, telekomunikasi akan bermain satu peran sangat penting di dalam terhubung

jaringan sistem informasi. Mengirimkan data dari satu tempat ke tempat lain. Sebagai tambahan, telekomunikasi di dalam suatu EIS dapat mempercepat kebutuhan akan akses ke distribusi data.

5. Aplikasi

EIS perbolehkan eksekutif untuk menemukan suatu data sesuai dengan kriteria didefinisikan pengguna dan meningkatkan keterangan mendasari pengertian yang mendalam dan pemahaman. Tidak sama dengan satu presentasi management information sistem tradisional, EIS dapat mencirikan di antara penting dan jarang data terpakai, dan jejak aktivitas berbeda kritis kunci untuk eksekutif, keduanya yang sangat menolong di dalam mengevaluasi kalau perusahaan sedang menjumpai obyektif perusahaannya. Setelah menyadari keuntungannya, orang-orang telah menerapkan EIS pada beberapa area, terutama, di pabrikan, pemasaran, dan biaya area.

6. Pabrikasi

Pada dasarnya, memproduksi menjadi perubahan bentuk bahan baku ke dalam barang jadi yang akan dijual, atau proses intermediate menyertakan produksi atau penyelesaian semi-manufactures. Ini adalah satu Branch besar dari industri dan dengan penghasilan sekunder. Membuat kontrol operasional fokuskan pada operasi sehari-hari, dan ide pusat dari proses ini adalah efektivitas dan efisiensi. Untuk menghasilkan managerial yang penuh arti dan keterangan operasional untuk mengontrol operasi pabrikasi, eksekutif harus membuat perubahan pada keputusan berjalan. EIS menyediakan evaluasi dari Vendor dan pembeli, evaluasi dari membeli materi dan bagian, dan analisa dengan area pembelian kritis. Oleh sebab itu,

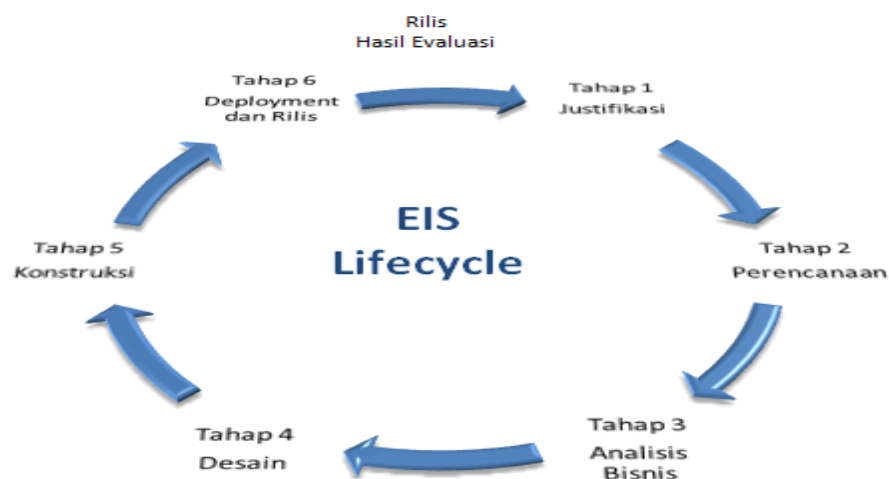
eksekutif dapat mengatur dan menelaah operasi pembelian secara efektif dengan EIS. Sebagai tambahan, karena perencanaan produksi dan kontrol menyesuaikan dengan berat pada datanya pabrik berlandaskan dan komunikasinya dengan sepenuhnya membuat pekerjaan pusat, EIS juga menyediakan satu pendekatan untuk meningkatkan perencanaan produksi dan kontrol.

7. Pemasaran

Pada satu organisasi, eksekutif pemasaran peran adalah untuk menciptakan masa depan. Bea utama mereka sedang mengatur sumber daya pemasaran tersedia untuk menciptakan satu lebih perdagangan berjangka efektif. Untuk ini, mereka memerlukan pertimbangan perbuatan sekitar risiko dan ketidakpastian dari satu proyek dan dampaknya pada perusahaan pada jangka pendek dan jangka panjang. Untuk membantu eksekutif pemasaran di keputusan pemasaran efektif pembuatan, satu EIS dapat teraplikasi. EIS menyediakan satu pendekatan ke peramalan penjualan, yang yang dapat mengizinkan eksekutif pasar untuk membandingkan penjualan ramalan dengan penjualan masa lalu. EIS juga menawarkan satu pendekatan ke harga produk, yaitu ditemukan di analisa proyek. Eksekutif pasar dapat mengevaluasi harga sebagai terkait ke kompetisi seiring dengan hubungan dari mutu produk dengan harga tertagih. Secara ringkas, Paket software EIS memperbolehkan eksekutif pemasaran untuk memanipulasi data dengan mencari kecenderungan, melaksanakan audit dari data penjualan, dan penjumlahan penghitung, rata-rata, perubahan, perbedaan, atau rasio. Semua fungsi analisa penjualan ini eksekutif pemasaran pertolongan untuk membuat keputusan terakhir.

8. Keuangan

Satu analisa keuangan adalah salah satu paling penting tahapan ke perusahaan hari ini. Eksekutif perlu mempergunakan rasio keuangan dan analisa arus kas untuk menaksir kecenderungan dan mengambil keuntungan keputusan investasi. Satu EIS adalah satu tanggungjawab mengorientasi pendekatan yang perencanaan terintegrasi atau penganggaran dengan kontrol dari laporan kinerja, dan ini dapat sangat sangat menolong ke eksekutif pendanaan. Pada dasarnya, EIS memfokuskan pada tanggungjawab dengan kinerja keuangan dan ini mengenali kepentingan dari standar biaya. EIS memperbolehkan eksekutif untuk memfokuskan lebih pada basis jangka panjang dari tahun arus dan berada di luar, yang berarti eksekutif bukan saja dapat mengatur satu aliran cukup untuk memelihara operasi arus kecuali juga dapat membayangkan bagaimana caranya memperluas operasi di tahun yang akan datang.



Gambar 2.1 siklus hidup Sistem Informasi Eksekutif

Dalam membangun EIS digunakan metode rekayasa siklus hidup sistem EIS yang terdiri dari beberapa tahapan yang digambarkan dalam gambar 2.1, hal ini dibahas oleh Lungu dkk(2005).

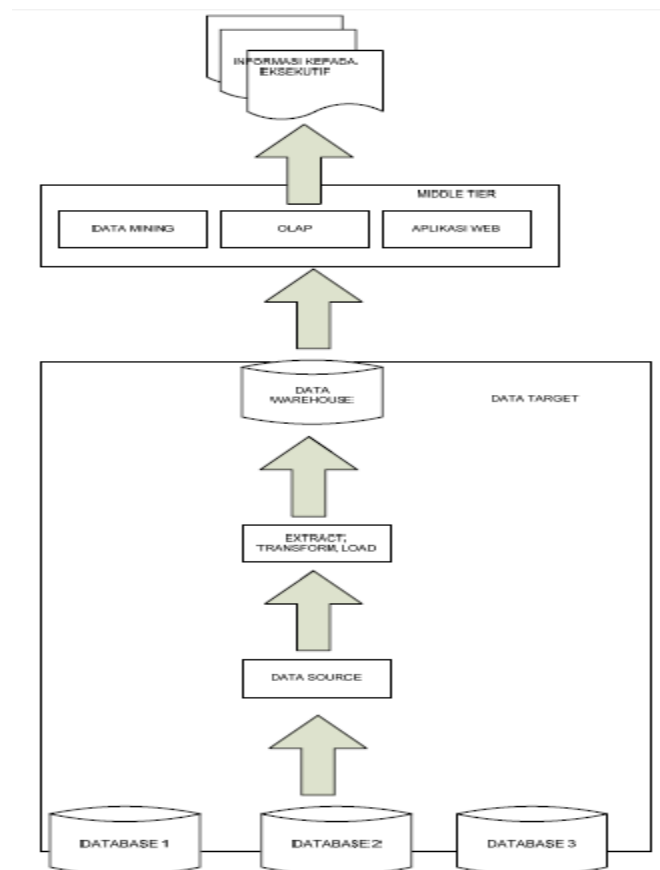
Banyaknya informasi yang dapat diperoleh akan sangat berguna dalam persaingan. Besarnya kemampuan untuk mendapatkan informasi yang beragam, maka perlu untuk melakukan peninjauan kembali terhadap informasi tersebut apakah masih relevan atau tidak. Peninjauan itu dapat dilakukan oleh pihak eksekutif sendiri, staff, maupun dengan bantuan mesin, hasil dari informasi tersebut kemudian dievaluasi kembali dan kemudian diteruskan ke analisis kualitas dan ke analisis kuantitas. Kemudian interpretasi diambil baik oleh pihak eksekutif maupun oleh tim berdasarkan informasi yang telah diperoleh, apabila timbul masalah baru pada hasil interpretasi tersebut, maka interpretasi tersebut akan dilanjutkan sebagai input pada phase selanjutnya untuk pengambilan keputusan terhadap masalah tersebut.

Salah satu keuntungan EIS adalah menyediakan kemampuan untuk melihat lebih detail dari informasi yang diberikan. Misalnya eksekutif menghadapi masalah kemunduran penjualan, berdasarkan laporan tersebut eksekutif kemudian mencari tahu daerah mana yang bermasalah pada penjualan, setelah mengetahui daerah yang penjualannya bermasalah kemudian eksekutif menganalisa produk maupun *sales person* yang ada pada daerah tersebut. Melalui *drill-down* ini pihak eksekutif dapat melihat permasalahan dan sumber informasinya lebih detail. Status akses pada program, pada dasarnya untuk menghindari adanya manipulasi pada informasi dan laporan. Status akses ini disesuaikan dengan *user*-nya. Untuk kemampuan *drill-down* status akses akan dibatasi, jadinya hanya *user* tertentu yang dapat melakukan *drill-down*.

Aristektur EIS terdiri dari 3 level, dijelaskan dalam gambar 2.2, hal ini dibahas oleh Lungu dkk(2005), yaitu:

1. Data Management, yaitu data yang disimpan dalam database, data-warehouse, dan lain-lain.

2. Model Management, level yang memuat proses pengambilan data dari sumber luar lainnya di luar database yang sudah ada dan mentransformasikannya untuk dimasukkan ke dalam database yang digunakan dalam EIS. Proses ini dikenal dengan nama *Extract, Transform, Load* (ETL).
3. Visualisasi Data, yaitu penampilan informasi baik berbentuk grafik maupun narasi yang digunakan oleh eksekutif untuk mengambil keputusan.



Gambar 2.2 Arsitektur Sistem Informasi Eksekutif

Penggunaan warna pada EIS sangat penting untuk menekankan bagian-bagian yang perlu mendapatkan perhatian lebih dari eksekutif, misalnya bila ada *warning* maka akan terlihat warna merah. Perlu juga *navigation of information*, agar eksekutif dapat dengan mudah mencari data

dan informasi yang dibutuhkan dengan cepat. Kemampuan ini sangat membantu pihak eksekutif terutama apabila terdapat data dalam jumlah yang besar.

Adapun karakteristik dan sifat sistem informasi eksekutif (EIS), sebagai berikut:

- Sesuaikan dengan pemakai eksekutif perorangan, mengekstrasi, menyaring, meringkas, dan menangkap data yang penting.
- Memberikan akses status online, analisa trend, pelaporan kekecualian dan drill down (yaitu memungkinkan pemakai untuk mengakses kerincian atau data yang mendukung yang berada di bawah yang teringkas).
- Mengakses dan mengadakan jangkauan data internal dan eksternal yang bersifat luas.
- Bersifat user-friendly dan menggunakannya hanya dibutuhkan ketrampilan yang sedikit tanpa pelatihan.
- Digunakan secara langsung oleh eksekutif tanpa intermediary (perantara).
- Menampilkan informasi grafik, tabular dan tekstual.

Dari beberapa hal tersebut di atas, maka sistem informasi eksekutif sebaiknya:

- Dapat memenuhi kebutuhan informasi bagi eksekutif senior.
- Harus dibuat atau dikembangkan oleh personil yang mempunyai ketrampilan bisnis maupun teknis.
- Harus mudah digunakan, sehingga bisa dianggap bersifat intuitif (mudah dimengerti).

Dalam membangun EIS para eksekutif menggunakan beberapa konsep dasar yang bertujuan memungkinkan para eksekutif dapat memantau seberapa baiknya kinerja perusahaan atau organisasi dalam mencapai tujuan. Konsep dasar tersebut terbagi atas 3 hal yaitu:

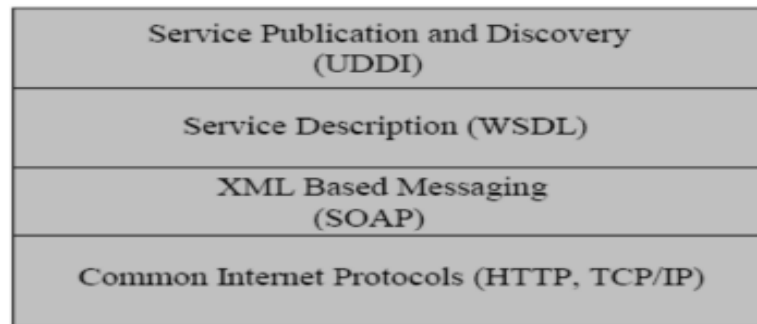
- faktor penentu keberhasilan (critical success factor)

adalah hal-hal menentukan keberhasilan atau kegagalan segala jenis kegiatan organisasi. Faktor-faktor ini dalam setiap organisasi berbeda-beda tergantung dari kegiatan yang dilakukan.

- **Management By Exception (MBE)**
Perbandingan antara kinerja yang direncanakan dengan kinerja actual. Sehingga informasi dapat langsung didapat dan digunakan untuk menyelesaikan setiap permasalahan.
- **Model Mental**
Peran utama EIS adalah membuat sari dari data informasi yang volumenya besar untuk meningkatkan kegunaannya. Pengambilan sari ini disebut penempatan informasi. Dimana menghasilkan sesuatu gambaran model mental dari operasi organisasi atau perusahaan.

2.5 Web Service

Web service adalah aplikasi sekumpulan data (database), perangkat lunak (software) atau bagian dari perangkat lunak yang dapat diakses secara remote oleh berbagai piranti dengan sebuah perantara tertentu. Secara umum, web service dapat diidentifikasi dengan menggunakan URL seperti halnya web pada umumnya. Namun yang membedakan web service dengan web pada umumnya adalah interaksi yang diberikan oleh web service. Berbeda dengan URL web pada umumnya, URL web service hanya mengandung kumpulan informasi, perintah, konfigurasi atau sintaks yang berguna membangun sebuah fungsi-fungsi tertentu dari aplikasi.



Gambar 2.3 Blok bangunan web service

Gambar 2.3 merupakan blok bangunan *web service* yang mana menyediakan fasilitas komunikasi jarak jauh antara dua aplikasi yang merupakan layer arsitektur *webservice*.

1. Layer 1 : protokol internet standar yang digunakan sebagai sarana transportasi adalah HTTP dan TCP/IP.
2. Layer 2 : *Simple Object Access Protocol* (SOAP) berbasiskan XML dan digunakan untuk pertukaran informasi antar sekelompok layanan.
3. Layer 3 : *Web service Definition Language* (WSDL) digunakan untuk mendiskripsikan attribute layanan.
4. Layer 4 : *Universal Description, Discovery and Integration*, yang mana merupakan direktori pusat untuk deskripsi layanan.

Web service dapat diartikan juga sebuah metode pertukaran data, tanpa memperhatikan dimana sebuah database ditanamkan, dibuat dalam bahasa apa sebuah aplikasi yang mengkonsumsi data, dan di platform apa sebuah data itu dikonsumsi. Web Service memungkinkan kita dapat mengakses informasi atau mendapatkan potongan informasi dari suatu aplikasi melalui aplikasi yang lain. Web service mampu menunjang interoperabilitas, Sehingga web service mampu menjadi sebuah jembatan penghubung antara berbagai sistem yang ada. Web service sendiri dibentuk dari :

- Service provider, merupakan pemilik Web Service yang berfungsi menyediakan kumpulan operasi dari Web Service.

- Service requestor, merupakan aplikasi yang bertindak sebagai klien dari Web Service yang mencari dan memulai interaksi terhadap layanan yang disediakan.
- Service registry, merupakan tempat dimana Service provider mempublikasikan layanannya. Pada arsitektur Web Service, Service registry bersifat optional. Teknologi web service memungkinkan kita dapat menghubungkan berbagai jenis software yang memiliki platform dan sistem operasi yang berbeda.

Ada beberapa komponen yang harus diperhatikan dalam pemanfaatan web services sebagai integrasi data koperasi, komponen tersebut berfungsi dapat menghasilkan *software* yang berkualitas. Adapun komponen yang perlu diperhatikan tersebut diantaranya: *Web Services*, *XML (eXtensible Markup Language)*, *WSDL (Web service Description Language)*, *UDDI (UniversalDescription Discovery, and Integration)*

- *XML (eXtensible Markup Language)*

XML merupakan suatu bahasa yang digunakan untuk mendeskripsikan dan memanipulasi dokumen secara terstruktur. Secara teknis XML didefinisikan sebagai suatu bahasa meta-markup yang menyediakan format tertentu untuk dokumen-dokumen yang mempunyai data terstruktur. Bahasa markup adalah mekanisme untuk mengenal suatu struktur didokumen. Tujuan dari XML secara umum untuk memungkinkan SGML untuk membantu, menerima dan memproses pada program web, dimana cara ini dapat dilakukan dengan HTML. XML mempunyai tujuan memungkinkan XML berfungsi sebagai suatu format pertukaran data (*dataexchange*) antar dua entitas, XML juga dapat berperan dalam penyimpanan dokumen agar dapat digunakan oleh program lain, karena formatnya standar dan mampu mendeskripsikan data.

- *WSDL (Web service Description Language)*

WSDL merupakan bahasa *standard* yang menyediakan mekanisme untuk mendeskripsikan *service* yang disediakan oleh sistem (*web service*), lokasi keberadaan *service* tersebut dan bagaimana cara memperolehnya, secara terstruktur dalam format XML. WSDL dapat dianalogikan sebagai IDL (*Interface definition language*) dalam CORBA dan COM. *Service* dideskripsikan sebagai koleksi dari *entry-point* atau *port* komunikasi. WSDL mendeskripsikan *service* dengan menggunakan elemen sebagai berikut:

1. *Type* – tipe data yang digunakan sebagai *argument* dan *return type*
2. *Message* – merepresentasikan definisi data yang ditransmisikan
3. *Port type* – sekumpulan operasi yang didukung oleh satu atau lebih *end-point*
4. *Binding* – mendefinisikan protokol dan format pertukaran data untuk operasi yang didefinisikan oleh *port type*
5. *Port* – menspesifikasikan *end-point* yang digunakan untuk *binding*

WSDL menyediakan sebuah kamus XML untuk menjabarkan detail-detail ini. WSDL digunakan di mana skema XML tidak digunakan lagi dengan menyediakan jalur pesan-pesan grup menjadi operasi-operasi dan operasi-operasi menjadi antarmuka. Ini juga menyediakan jalur untuk mendefinisikan *binding-binding* untuk setiap antarmuka dan kombinasi protokol sepanjang alamat titik akhir untuk setiap kalinya. Definisi WSDL yang lengkap terdiri dari seluruh informasi yang dibutuhkan untuk meminta *web service*. Pengembang yang mau mempermudah yang lain untuk mengakses *service-service*nya harus menyediakan definisi-definisi WSDL.

WSDL memainkan peranan penting pada seluruh arsitektur *web service* semenjak menjabarkan kontrak lengkap pada komunikasi aplikasi. Walaupun teknik-teknik lain untuk menjabarkan Web service ada, WS-I Basic Profile Versi 1.0 memadati penggunaan WSDL dan skema XML untuk menjabarkan *web service*. Ini membantu untuk memastikan interoperabilitas pada layer deskripsi servis. Karena WSDL adalah mesin yang dapat dibaca (misalnya hanya file XML), *tool-tool* dan infrastruktur dan dengan mudah dibuat seputar ini.

Saat ini pengembang dapat definisi-definisi WSDL untuk membangun kode yang tahu dengan tepat bagaimana berinteraksi dengan *web service* yang menjabarkan. Pembangunan code tipe ini menyembunyikan detail-detail membosankan yang terlibat pada pengiriman dan penerimaan pesan-pesan SOAP pada protokol-protokol yang berbeda-beda dan menyebabkan *web service* dapat dicapai oleh massa. Kelas-kelas digenerasi dari definisi WSDL sama harus mampu berkomunikasi dengan yang lain sepanjang antarmuka WSDL yang tersedia, tanpa memperhatikan bahasa pemrograman yang digunakan

- UDDI (UniversalDescription Discovery, and Integration)
UDDI menyediakan layanan direktori pusat untuk mempublikasikan informasi teknikal layanan web.

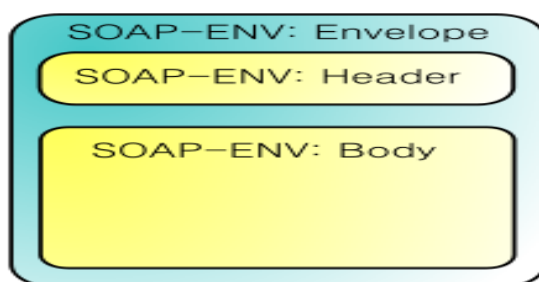
2.6 Metode SOAP (Simple Object Access Protocol)

SOAP Merupakan *protocol* pertama dari jenisnya yang akan diterima secara praktis oleh setiap perusahaan perangkat lunak besar dunia. SOAP menyediakan cara standar pemaketan sebuah pesan. SOAP sangat efektif untuk berbagai macam aplikasi multi tier untuk berkomunikasi lintas sistem operasi/platfrom yang berbeda dengan teknologi dan bahasa pemrograman yang berbeda-beda pula.

SOAP menjadi sangat mudah diterima oleh berbagai pihak , terutama oleh berbagai vendor TI , dikarenakan protokol ini memanfaatkan berbagai teknologi yang sudah ada sebelumnya dan sudah banyak digunakan. Misalnya untuk protokol transport, yang paling banyak digunakan adalah HTTP, walaupun dimungkinkan untuk menggunakan protokol transport lainnya. Sedangkan untuk format data atau message digunakan XML yang tidak diragukan lagi manfaat dan perannya di dalam pertukaran data.

Peran SOAP di dalam teknologi *web service* adalah sebagai protokol pemaketan untuk pesan-pesan (*messages*) yang digunakan secara bersama oleh aplikasi-aplikasi penggunanya. Spesifikasi yang digunakan tidak lebih seperti sebuah amplop biasa berbasis XML untuk informasi yang ditransfer, serta sekumpulan aturan bagi translasi aplikasi dan tipe-tipe data platform yang spesifik menjadi bentuk XML. Desain bentuk dari SOAP membuatnya cocok untuk berbagai pertukaran pesan pada aplikasi.

Adapun struktur XML dalam SOAP selama proses transaksi antar sistem baik dalam proses call/request maupun response dalam web service, digambarkan pada gambar berikut ini:



Gambar 2.4 Struktur XML dalam SOAP

Sebuah pesan SOAP adalah sebuah dokumen XML yang berisi elemen-elemen berikut:

1. *Envelope element* yang mengidentifikasi dokumen XML sebagai sebuah pesan SOAP.
2. Elemen *header* yang berisi informasi *header*. Elemen ini bersifat opsional.

3. Elemen *body* yang berisi panggilan dan merespon informasi.
4. *Fault element* yang berisi pesan kesalahan yang terjadi pada waktu proses. Elemen ini opsional.

Dengan demikian, tidaklah terlalu mengherankan bila kemudian SOAP dianggap sebagai solusi penyelamat untuk mengatasi berbagai masalah yang dihadapi oleh teknologi – teknologi pendahulunya.

2.7 NuSOAP

NuSOAP adalah *library* yang digunakan untuk membangun *web service* berbasis SOAP yang ditulis dengan menggunakan bahasa PHP. NuSOAP merupakan sebuah kumpulan class-class PHP yang memungkinkan user untuk mengirim dan menerima pesan SOAP melalui protokol HTTP. NuSOAP ditulis oleh Dietrich Ayala dengan proyek awal bernama SOAPx4. Selanjutnya didistribusikan oleh NuSphere Corporation (<http://www.nusphere.com>) sebagai open source toolkit dibawah lisensi GNU LGPL. NuSOAP merupakan toolkit *web service* berbasis komponen. NuSOAP memiliki sebuah class dasar yang menyediakan method seperti serialisasi variabel dan pemaketan.

SOAP-Envelope. Interaksi *web service* dilakukan dengan class client yang disebut dengan class “soapclient” dan class server yang disebut dengan class “soap_server”. Class-class ini mengizinkan user untuk melakukan proses pengiriman dan penerimaan pesan-pesan SOAP dengan bantuan beberapa class-class pendukung lainnya untuk melengkapi proses tersebut

Dalam membangun sebuah sistem aplikasi berbasis *web service*, dibutuhkan dua komponen utama yaitu, *server* sebagai *provider entity* dan *client* sebagai *requester entity*.

2.8 Konsep Dasar Basis Data

Basis data merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dandigunakan perangkat lunak untuk memanipulasinya. Database merupakan

salah satu komponen yang penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi bagi para pemakai. Penerapan database dalam sistem informasi disebut dengan database system. Sistem dasar data (*database system*) adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan satu dengan yang lainnya dan membuat tersedia untuk beberapa aplikasi yang bermacam-macam dalam suatu organisasi.

Database bisa diartikan sebagai suatu file database yang memiliki tabel, record, field, index, query, filter dan view. Berikut adalah definisi umum isi sebuah file database.

1. Tabel adalah sekelompok record data, masing-masing berisi informasi yang sejenis.
2. Record adalah entri tunggal dalam tabel. Bisa saja disebut sebagai baris mengingat sebuah tabel terdiri dari baris (record) dan kolom (field).
3. Field adalah item tertentu dalam tabel. Bisa disebut sebagai kolom.
4. Index adalah field kunci yang ditujukan ke suatu record yang spesifik serta diurutkan dalam urutan tertentu.
5. Query adalah perintah SQL yang dirancang untuk memanggil kelompok record tertentu dari satu tabel/lebih.
6. View merupakan tabel virtual yang berisi record dari berbagai tabel. Fungsi utamanya untuk memudahkan kita mendapatkan data yang spesifik dari berbagai tabel.

2.8.1 DBMS (*Database Management System*)

Pada basis data terdapat istilah independensi data. Independensi data adalah sifat yang memungkinkan perubahan struktur berkas yang tidak mempengaruhi program, dan juga sebaliknya. Independensi data juga bisa berarti bahan data yang bersifat tidak bergantung pada data lain. Bila suatu data dihapus, data lain harus tidak berpengaruh oleh tindakan tersebut.

Sistem pemrosesan basis data dimaksudkan untuk mengatasi kelemahan-kelemahan yang ada pada sistem pemrosesan berkas.

Kebanyakan DBMS menyediakan mekanisme pengaturan sekuritas terhadap basis data berdasarkan wewenang pengguna. Sebagai contoh, si A hanya boleh membaca satu data, tetapi si B boleh mengubahnya. Sekuritas data sangat penting pada basis data yaitu untuk menghindari pengaksesan data sensitif oleh orang yang tidak berhak.


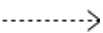
2.9 UML (Unified Modelling Language)









UML (*Unified Modelling Language*) adalah bahasa untuk menspesifikasikan, memvisualisasikan, serta mengkonstruksi bangunan dasar dari sistem perangkat lunak termasuk didalamnya dengan melibatkan pemodelan aturan-aturan bisnis. Beberapa Diagram UML, diantaranya adalah sebagai berikut.

2.9.1 Use Case Diagram

Use Case diagram merupakan suatu diagram yang menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah use case dapat memrepresentasikan interaksi antara aktor dengan sistem. Adapun simbol-simbol dari use case diagram dapat dilihat pada tabel 2.1 sebagai berikut.

Tabel 2.1 Simbol use case Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
2		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri(<i>independent</i>) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (<i>independent</i>).





NO	GAMBAR	NAMA	KETERANGAN
3		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
4		<i>Include</i>	Menspesifikasikan bahwa <i>use case</i> sumber secara <i>eksplisit</i> .
5		<i>Extend</i>	Menspesifikasikan bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.
6		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya.
7		<i>System</i>	Menspesifikasikan paket yang menampilkan sistem secara terbatas.
8		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
9		<i>Collaboration</i>	Interaksi aturan-aturan dan elemen lain yang bekerja sama untuk menyediakan perilaku yang lebih besar dari jumlah dan elemen-elemennya (sinergi).
10		<i>Note</i>	Elemen fisik yang eksis saat aplikasi dijalankan dan mencerminkan suatu sumber daya komputasi

2.9.2 Activity Diagram

Activity diagram menggambarkan berbagai alur aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alur berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

Activity diagram merupakan state diagram khusus, di mana sebagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (internal processing). Oleh karena itu activity diagram tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Adapun simbol pada activity diagram dapat dilihat pada tabel 2.2 sebagai berikut.

Tabel 2.2 Simbol Activity Diagram

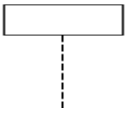

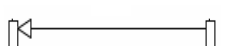
NO	GAMBAR	NAMA	KETERANGAN
1		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain
2		<i>Action</i>	State dari sistem yang mencerminkan eksekusi dari suatu aksi
3		<i>Initial Node</i>	Bagaimana objek dibentuk atau diawali.
4		<i>Activity Final Node</i>	Bagaimana objek dibentuk dan dihancurkan
5		<i>Fork Node</i>	Satu aliran yang pada tahap tertentu berubah menjadi beberapa aliran

1.9.3 Sequential Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa message yang digambarkan terhadap waktu. Sequence diagram terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). Sequence diagram dapat digunakan untuk menggambarkan skenario atau rangkaian

langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan output tertentu. Diawali dari apa yang men-trigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki lifeline vertikal. Message digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, message akan dipetakan menjadi operasi/metoda dari class. Adapun simbol sequence diagram dapat dilihat pada tabel 2.3 sebagai berikut.

Tabel 2.3 Simbol Sequence Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi
3		<i>Message</i>	Spesifikasi dari komunikasi antar objek yang memuat informasi-informasi tentang aktifitas yang terjadi



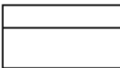




1.9.4 Class Diagram

Class Diagram adalah diagram yang digunakan untuk menampilkan beberapa kelas serta paket-paket yang ada dalam sistem atau perangkat lunak yang sedang kita gunakan. Class Diagram memberikan gambaran tentang sistem atau perangkat lunak dan relasi-relasi yang ada di dalamnya.

Elemen –elemen class diagram dalam pemodelan UML terdiri dari: Class-class, struktur class, sifat class, perkumpulan atau gabungan (association), pengumpulan atau kesatuan (agregation),

ketergantungan (dependency), relasi-relasi turunannya, keberagaman dan indikator navigasi, role name (peranan tugas nama). Adapun simbol-simbol class diagram dapat dilihat pada tabel 2.4 sebagai berikut.

Tabel 2.4 Simbol Class Diagram

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Generalization</i>	Hubungan dimana objek anak (<i>descendent</i>) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk (<i>ancestor</i>).
2		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
4		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor
5		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek.
6		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (<i>independent</i>) akan memengaruhi elemen yang bergantung padanya elemen yang tidak mandiri
7		<i>Association</i>	Apa yang menghubungkan antara objek satu dengan objek lainnya