

BAB II

LANDASAN TEORI

2.1 SISTEM INFORMASI

Pengertian sistem informasi tidak bisa dilepaskan dari pengertian sistem dan informasi. Definisi sistem adalah kumpulan elemen-elemen yang berinteraksi untuk mencapai tujuan tertentu. Sedangkan definisi dari informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

Data diolah melalui suatu model menjadi informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus. Siklus itu disebut dengan siklus informasi (*information cycle*).

Untuk dapat berguna, maka informasi harus didukung oleh tiga hal sebagai berikut: tepat nilainya atau akurat (*accurate*), tepat waktu (*timeliness*) dan tepat kepada orangnya atau relevan (*relevance*). Keluaran yang tidak didukung ketiga pilar ini tidak dapat dikatakan sebagai informasi yang berguna, tetapi merupakan sampah (*garbage*) (Jogiyanto, 1999: 10).

Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan tidak biasa atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya.

Tepat pada waktunya, berarti informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi karena informasi merupakan landasan didalam pengambilan keputusan.

Relevan, berarti informasi tersebut harus mempunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda. Sistem informasi memiliki tiga fungsi dasar :

1. Menerima data (*input*)
2. Mengubah data menjadi informasi (proses)
3. Untuk memproduksi dan mengkomunikasikan informasi ke dalam *timely fashion* bagi *user* untuk membuat keputusan (*output*).

2.2 KOMPONEN SISTEM INFORMASI

Sistem informasi mempunyai enam buah komponen, yaitu *input*, model, *output*, teknologi, basis data, dan kontrol. Keenam komponen ini harus ada bersama-sama dan membentuk satu kesatuan. Jika satu atau lebih komponen tersebut tidak ada, maka sistem informasi tidak akan dapat melakukan fungsinya, yaitu pengolahan data dan tidak dapat mencapai tujuannya, yaitu menghasilkan informasi yang relevan, tepat waktu, dan akurat. Komponen-komponen dari sistem ini dapat digambarkan sebagai berikut ini :

1. *Input*

Input merupakan data yang masuk ke dalam sistem informasi. Sistem sistem informasi tidak akan

dapat menghasilkan *output* jika tidak mempunyai komponen *input*.

2. *Output*

Produk dari sistem informasi adalah *output* berupa informasi yang berguna bagi para pemakainya. *Output* dari sistem informasi dibuat dengan menggunakan data yang ada di basis data dan diproses menggunakan model tertentu.

3. Basis data

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

4. Model

Model yang digunakan di sistem informasi dapat berupa model logika yang menunjukkan suatu proses perbandingan logika atau model matematik yang menunjukkan perhitungan matematika.

5. Teknologi

Teknologi merupakan komponen yang penting di sistem informasi. Teknologi dapat dikelompokkan ke dalam dua macam kategori, yaitu teknologi sistem komputer (perangkat keras dan perangkat lunak) dan teknologi sistem telekomunikasi.

6. Kontrol

Kontrol ini digunakan untuk menjamin bahwa informasi yang dihasilkan oleh sistem informasi sifatnya akurat.

2.3 WEB BASED INFORMATION SYSTEM

Saat ini komputer dan piranti pendukungnya telah masuk dalam setiap aspek kehidupan dan pekerjaan. Komputer yang ada sekarang memiliki kemampuan yang lebih dari sekedar perhitungan matematika biasa.

Internet adalah jaringan computer internasional yang menghubungkan orang dan organisasi diseluruh dunia. Dengan ditemukannya internet yang merupakan hasil kemajuan yang sangat pesat di bidang teknologi informasi dan komunikasi, maka kegiatan operasional antar perusahaan tidak terbatas pada satu tempat saja.

Dengan menggunakan internet maka pemikiran bisnis tentang pengiriman nilai ke konsumennya, interaksi dengan supplier dan juga mengelola tenaga kerjanya dapat dilakukan dengan cepat dan efektif.

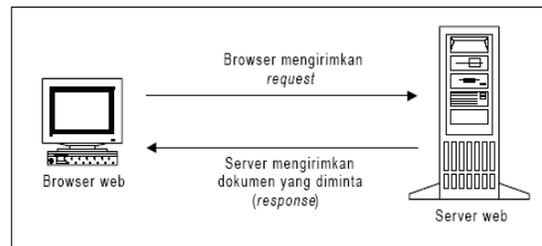
Oleh karena itu pada era sekarang ini banyak perusahaan yang telah memanfaatkan jasa pelayanan internet untuk menunjang kegiatan bisnisnya, demikian pula dalam manajemen operasional di perusahaan.

Web adalah fasilitas *hypertext* untuk menampilkan data berupa teks, gambar, bunyi, animasi, dan data multimedia lainnya, yang mana data tersebut saling berhubungan satu sama lainnya.

2.4 WEB SERVER

Web server adalah suatu perangkat lunak yang mengatur halaman *web* dan membuat halaman-halaman *web* tersebut dapat diakses di klien, yaitu melalui jaringan lokal atau melalui jaringan *Internet*. Ada banyak *web*

server yang tersedia diantaranya *Apache*, *IIS (Internet Information Service)*, dan *IPlanet's Enterprise server*.



Gambar 2.1 Konsep dasar browser dan server web

2.5 WEB SERVICE

Web Service merupakan salah satu teknologi yang memfasilitasi komunikasi pada platform yang berbeda. *Web Service* memiliki konsep interoperabilitas yaitu kemampuan suatu sistem atau teknologi untuk dapat berkomunikasi dengan sistem lain yang berbeda melalui suatu protokol atau sistem tertentu yang dikenal dengan *Middleware*.

Teknologi *Middleware* didefinisikan sebagai suatu sistem yang berada di antara dua sistem yang berbeda yang memfasilitasi tata cara komunikasi keduanya

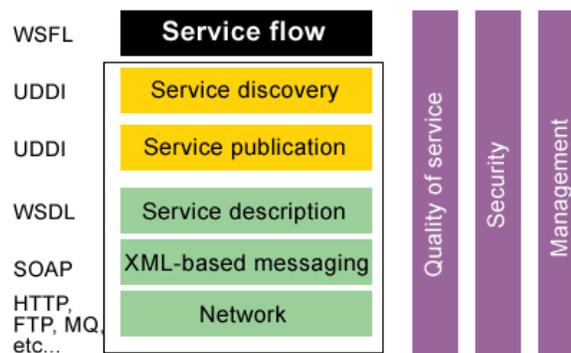
Web Service memiliki beberapa karakteristik, di antaranya :

- Pada umumnya *Web Service* adalah bagian dari solusi suatu bisnis proses atau sebuah sistem.
- *Platform-agnostics*, *Web service* selayaknya dapat dikembangkan dan dikonsumsi oleh semua platform.
- Interoperabilitas, *Web service* selayaknya dapat mendukung model komunikasi yang bersifat sederhana dan dapat memfasilitasi

komunikasi antara dua sistem secara lebih mudah.

- Berjalan di protokol standar Web, misalnya protokol Web, seperti SMTP, HTTP.
- Tanpa GUI, Web service berada di aras logika bisnis, sehingga secara konseptual Web service tidak memiliki antarmuka.

How does it stack up?



Gambar 2.2 Arsitektur Web Service

2.6 WEB SERVICE PADA SMART PROJECT

Teknik sinkronisasi pada dunia aplikasi berbasis .NET dapat dilakukan melalui pendekatan sinkronisasi basisdata ataupun sinkronisasi menggunakan komponen bisnis. Sinkronisasi pada basis data pada umumnya membutuhkan suatu teknologi tambahan yang memfokuskan bagaimana data dapat senantiasa sinkron antara satu basis data dengan basis data lain. Pada pengembang aplikasi Windows Mobile, teknik sinkronisasi dibantu dengan salah satu komponen yang sudah tertanam di SQL CE, yakni *Synchronization Services*. *Synchronization*

Services membantu pola komunikasi dan sinkronisasi dengan basis data host, seperti SQL Server. Teknik lain adalah dengan menggunakan komponen bisnis, baik yang dibuat sendiri ataupun dikembangkan oleh pihak lain, seperti Microsoft dengan *Microsoft Sync Framework*.

2.7 TEKNOLOGI ASP.NET

Active Server Pages .NET (sering disingkat sebagai ASP.NET) adalah sebuah teknologi layanan web dinamis, aplikasi web, dan XML web service yang dikembangkan oleh Microsoft sebagai pengganti *Active Server Pages* (ASP) yang lama. Teknologi ini berbasis *.NET Framework* dan dibangun di atas *Common Language Runtime* (CLR), sehingga para programmer dapat menulis kode ASP.NET dengan menggunakan semua bahasa pemrograman .NET, meski yang populer digunakan adalah bahasa C# dan *Visual Basic .NET*.

ASP.NET memiliki beberapa kelebihan dibandingkan teknologi terdahulu, antara lain:

1. Kemudahan mengakses berbagai *library .NET Framework* secara konsisten dan *powerful*, yang mempercepat pengembangan aplikasi.
2. Penggunaan berbagai bahasa pemrograman secara penuh, misalnya VB.NET, C#, J#, dan C++. Dalam ASP.NET bahasa-bahasa ini dapat digunakan secara penuh sebagaimana layaknya bekerja di *windows application*. Hal ini jauh berbeda dibandingkan ASP klasik yang menggunakan *scripting language*. Misalnya VBScript agak berbeda syntaxnya dengan VB, dalam ASP.NET kita dapat menggunakan VB.NET dengan syntax yang 100% sama dengan

ketika membangun *windows application*. Demikian juga dengan C#, tidak ada perbedaan syntax antara C# untuk membuat kode ASP.NET dengan *windows form*. Ini sangat memudahkan memprogram aplikasi *windows* dan *web* secara konsisten.

3. Tersedia berbagai *web control* yang dapat digunakan membangun aplikasi secara cepat. *Programmer* dapat dengan mudah mengaitkan data ke *web control* sebagaimana layaknya memprogram *windows application*. Hal ini sangat mempercepat pembuatan aplikasi dibandingkan harus menyusun kode-kode HTML secara manual.

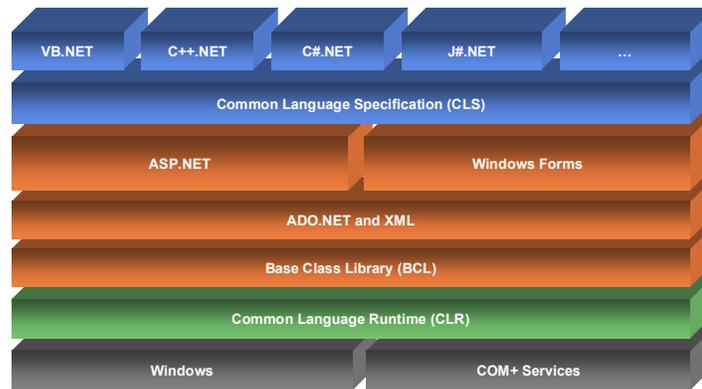
4. *Code Behind*, artinya kode-kode pemrograman yang menjadi *logic* aplikasi ditempatkan terpisah dengan kode *user interface* yang berbentuk HTML. Ini sangat memudahkan dalam *debugging*, karena kode untuk *presentation layer* tidak tercampur dengan kode *application logic*.

2.8 .NET FRAMEWORK

Microsoft .NET ialah model pemrograman dari *platform* .NET untuk membangun, menyebarkan dan menjalankan XML *web service* dan aplikasi. .NET *Framework* menyediakan lingkungan berbasis standar produksi yang tinggi untuk memadukan investasi yang ada dengan aplikasi dan *service* generasi yang akan datang. Arsitektur .NET dapat dilihat pada gambar 2.2, .NET *Framework* terdiri atas dua bagian utama, yaitu *Common Language Runtime* (CLR) dan gabungan kelas *library* termasuk ASP.NET untuk aplikasi *web* dan XML *web*

services, windows forms untuk aplikasi *client* dan ADO.NET.

Visual Studio .NET dibangun menggunakan fondasi *.NET Framework*. *.NET Framework* menyediakan lingkungan yang cerdas, mudah dikembangkan untuk membangun, menyebarkan dan menjalankan XML *web services* yang terdistribusi. Dalam istilah yang mudah, *.NET* memisahkan *platform* sistem operasi menjadi dua *layer*, yaitu *layer* pemrograman dan *layer* eksekusi.



Gambar 2.3 Arsitektur .NET Framework

2.9 USER CONTROL VISUAL STUDIO .NET

Pada suatu saat, kita mungkin memerlukan fungsionalitas dalam sebuah *control* yang tidak disediakan oleh *built-in Web server control* ASP.NET. Dalam kasus ini, kita dapat menciptakan *control* kita sendiri. Kita dapat menciptakan *user control*.

User control adalah kontainer yang dapat menampung *markup* dan *Web server control*. Kita dapat memperlakukan *user control* sebagai sebuah unit dan mendefinisikan properti-properti dan *method-method* untuk *user control* tersebut. *User control* lebih mudah untuk diciptakan

daripada *custom control*, karena kita dapat menggunakan *control-control* yang sudah ada. Hal tersebut membuat lebih mudah untuk menciptakan *control* dengan elemen-elemen antarmuka yang kompleks.

Sebuah ASP.NET *Web server control* mirip dengan sebuah halaman komplit ASP.NET *Web page* (.aspx), dengan sebuah halaman antarmuka dan *code*-nya. Kita dapat menciptakan *user control* dengan cara yang sama ketika kita menciptakan sebuah halaman ASP.NET dan menambahkan *markup* dan *child control* yang kita butuhkan. Sebuah *user control* dapat menyertakan *code* untuk memanipulasi isinya seperti yang bisa dilakukan pada halaman ASP.NET, termasuk melakukan tugas-tugas seperti *data binding*.

Sebuah *user control* memiliki perbedaan dari sebuah halaman *Web* ASP.NET dalam beberapa hal, yaitu :

1. Nama ekstensi file untuk *user control* adalah .ascx.
2. *User control* mengandung sebuah petunjuk *Control* yang mendefinisikan konfigurasi dan properti-properti yang lain.
3. *User control* tidak dapat berjalan sebagai sebuah file yang *stand-alone*. Kita harus menambahkan *user control* ke sebuah halaman *Web* ASP.NET.
4. *User control* tidak memiliki elemen *html*, *body*, atau *form* di dalamnya.

Kita dapat menggunakan elemen-elemen HTML yang sama (kecuali elemen *html*, *body*, atau *form*) dan *Web control* pada sebuah *user control* seperti yang kita lakukan pada sebuah halaman *Web* ASP.NET. Sebagai contoh, jika kita menciptakan sebuah *user control* untuk

digunakan sebagai *toolbar*, kita dapat meletakkan beberapa tombol *Web server controls* dalam *user control* tersebut dan menciptakan *event handler* untuk tombol-tombol tersebut.

2.10 ASPEK USABILITY PENGGUNA

Usability adalah kemampuan sebuah sistem agar mudah digunakan dan sederhana dalam pengoperasiannya. Secara umum *usability* merujuk pada sebaik apa pengguna dapat belajar dalam menggunakan sistem untuk mencapai tujuannya, dan seberapa puas pengguna akan proses tersebut.

Usability berarti bahwa pengguna yang menggunakan sistem dapat melakukan dengan cepat dan mudah untuk mencapai tujuannya. *Usability* mengukur kualitas dari pengalaman pengguna ketika berinteraksi dengan sebuah sistem baik itu sebuah situs *web*, sebuah aplikasi perangkat lunak, teknologi *mobile*, dan semua peralatan yang dioperasikan oleh manusia.

Usability terdiri dari kombinasi beberapa aspek yang sangat penting, yaitu :

1. *Learnability* adalah aspek yang berbicara tentang seberapa cepat seorang pengguna yang tidak pernah melihat antarmuka sebelumnya untuk belajar secukupnya untuk menyelesaikan tugas dasarnya.
2. *Efficiency of use* adalah aspek yang berbicara tentang seberapa cepat pengguna menyelesaikan tugasnya setelah seorang pengguna yang berpengalaman telah belajar menggunakan sistem.

3. *Memorability* adalah aspek yang berbicara tentang jika seorang pengguna telah terbiasa dengan sistem sebelumnya, dapatkah pengguna tersebut ingat untuk menggunakan sistem tersebut secara efektif pada saat akan memakainya lagi, ataukah pengguna harus belajar lagi semuanya dari awal.

4. *Error frequency and severity* adalah aspek yang berbicara tentang seberapa sering pengguna membuat kesalahan selama menggunakan sistem, seberapa serius kesalahan ini, dan bagaimana pengguna mengatasi masalah ini.

5. *Subjective satisfication* adalah aspek yang berbicara tentang seberapa besar kepuasan subjektif pengguna dalam menggunakan sistem.

2.11 SQL SERVER 2005

Sql Server memakai sebuah tipe database yang dinamakan *relational*. *Database relational* adalah database yang mengorganisasikan data dalam bentuk tabel. Tabel yang dibentuk dengan mengelompokkan data yang mempunyai subjek yang sama. Tabel berisi baris - baris dan kolom -kolom informasi. Tabel - tabel dapat saling berhubungan jika diinginkan.

Database merupakan kumpulan dari elemen-elemen data. Elemen data tersebut terbagi ke dalam tabel-tabel, dimana tabel-tabel itu mempunyai sejumlah kolom dan baris. Suatu tabel akan mempunyai kolom kunci yang digunakan untuk membangun hubungan dengan tabel lain.

Dalam *SQL Server*, *database* bukanlah sebuah file tetapi merupakan sebuah konsep logis yang berisi sekumpulan objek yang berhubungan.

SQL Server mendukung pembuatan aplikasi web yang memakai database. Beberapa fasilitas yang ada :

1. *SQL Server* mendukung ADO yang merupakan interface yang disarankan dipakai pada saat mengembangkan aplikasi *Web* dengan memakai *Internet Information Service(IIS)* dan *Active Server Pages(ASP)*.
2. *Retrive* dan *Publish* data menggunakan *SQL Sever Web Assistant*.
3. Distribusi data menggunakan *SQL Server*.