

BAB 2

LANDASAN TEORI

2.1 Sistem Informasi

Sistem bisa dikatakan sebagai kumpulan dari obyek atau elemen yang berinteraksi satu sama lain untuk mencapai tujuan tertentu.

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

Informasi juga harus mempunyai kualitas, kualitas suatu informasi tergantung dari tiga hal, yaitu (Jogiyanto, 1999):

1. **Akurat;** Informasi yang dihasilkan harus bebas dari kesalahan-kesalahan dan tidak menyesatkan.
2. **Tepat waktu;** Informasi yang diterima harus tepat waktu dan tidak boleh terlambat. Informasi yang sudah usang tidak mempunyai nilai lagi karena akan berakibat fatal untuk organisasi.
3. **Relevan;** Informasi tersebut mempunyai nilai manfaat bagi pemakai.

Sehingga informasi yang berkualitas adalah informasi yang terbebas dari kesalahan dan dapat mencerminkan

maksud dari informasi dengan jelas, tidak usang dan mempunyai manfaat bagi penerima informasi tersebut. Informasi dapat diperoleh dari sistem informasi yang disebut juga dengan *processing systems* atau *information processing systems*. Sistem informasi dapat didefinisikan sebagai berikut : Suatu sistem informasi adalah suatu kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar untuk mengambil keputusan yang cerdas.

Sistem informasi terdiri dari beberapa komponen yang saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai suatu tujuan tertentu. Komponen-komponen dari sistem informasi dapat disebut dengan istilah blok bangunan (*building block*), yaitu : blok masukan (*input block*), blok model (*model block*), blok keluaran (*output block*), blok teknologi (*teknology block*) dan blok kendali (*control block*). Dalam hal ini juga dibutuhkan blok basis data (*database block*) yang mendukung Sistem Informasi.

2.1.1 Basis Data

Basis Data adalah suatu kumpulan data terhubung (*Interrelated data*) yang disimpan secara bersama-sama pada suatu media, tanpa adanya suatu kerangkapan data (*Controlled redundancy*) dengan cara-cara tertentu sehingga mudah untuk digunakan atau ditampilkan kembali

(Sutanta, 1996). Basis data dapat didefinisikan dalam berbagai sudut pandang, seperti:

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat digunakan kembali.
2. Kumpulan data yang saling berhubungan yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redudansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan.
3. Kumpulan *file*/tabel/arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

Dalam pemanfaatan basis data yang mendukung suatu sistem informasi harus memenuhi sejumlah tujuan, yaitu :

1. Kecepatan dan kemudahan (*Speed*).
Pada saat dilakukan kegiatan menyimpan data, manipulasi dan menampilkan data kembali harus dapat dilakukan dengan cepat.
2. Efisiensi ruang penyimpanan (*Space*).
Untuk melakukan penyimpanan data diusahakan untuk memanfaatkan ruang penyimpanan semaksimal mungkin, dengan cara mengurangi redudansi.
3. Keakuratan (*Accuracy*).
Data pada saat digunakan harus sama seperti pada saat disimpan sehingga tidak terjadi kesalahan dan ketidakakuratan informasi.
4. Ketersediaan (*Availability*).
Data yang disimpan dalam basis data harus siap digunakan kapanpun dan ruang dalam basis data harus selalu siap untuk pertumbuhan basis data.
5. Kelengkapan (*Completeness*).

Kelengkapan suatu data sangat relatif, sehingga basis data harus dapat mengakomodasi penambahan data dan struktur data.

6. Keamanan (*Security*).

Basis data harus dapat memberikan jaminan kepada pengguna bahwa data yang disimpan aman, tidak akan hilang, berubah maupun dapat diakses sembarang pengguna.

7. Kebersamaan Pemakai (*Shareability*).

Basis data tidak hanya bisa dimanfaatkan oleh satu orang pengguna saja, tetapi sebaiknya dapat dibagi dengan pengguna lainnya.

Database Management System (DBMS) berisi suatu koleksi data yang saling berelasi dari suatu set program untuk mengakses data tersebut. Dengan kata lain DBMS adalah kumpulan *file* yang saling berkaitan dengan program untuk mengelolanya. Jadi DBMS terdiri dari basis data dan set program pengelola untuk menambah, menghapus, mengambil dan membaca data (*Kristanto, 1994*). Sistem basis data terdiri dari empat komponen, yaitu data, perangkat keras, perangkat lunak, dan *brainware*.

1. Data merupakan sekumpulan fakta yang akan diolah menjadi informasi. Data ini akan dikumpulkan pada suatu media penyimpan dalam bentuk basis data.
2. Perangkat Keras merupakan piranti komputer yang digunakan untuk menyimpan dan memproses data yang telah disimpan.
3. Perangkat Lunak merupakan program-program yang digunakan untuk memberi perintah pada komputer mengerjakan suatu pekerjaan tertentu.

4. *Brainware* merupakan orang yang berkepentingan dengan data serta hasil pengolahan data.

2.2 Sistem Informasi Berbasis *Mobile*

Saat ini, lingkungan pekerjaan sudah mengalami revolusi dari kantor tradisional menjadi kantor *mobile*. Perkembangan teknologi piranti *mobile* yang didukung oleh internet, menciptakan akses di setiap waktu dan tempat. Piranti *mobile* menjadi semakin pintar, efektif dan mempunyai kecepatan tinggi.

Banyaknya pemakai piranti *mobile* telah merevolusi kantor tradisional menjadi kantor *mobile*. Mobilitas yang tinggi tidak menjadi penghalang lagi, karena sejak lima tahun belakangan, piranti *mobile* sudah mampu mengakses server di pusat data. Piranti *mobile* sekarang ini tidak hanya berfungsi sebagai pencatat jadwal dan buku alamat. Fungsi piranti *mobile* sudah berkembang pesat, dan idealnya siap menggantikan dokumen bisnis berbasis kertas.

Secara dasar aplikasi piranti *mobile* terbagi atas :

1. *Personal Information Management (PIM)*: Menyediakan fungsi seperti kalender, buku alamat, jadwal memo dan tugas yang harus dilakukan.
2. Dokumen : Menyediakan fungsi *word processing* dan *spread sheet*. Fungsi-fungsi yang lebih maju antara lain akses web, e-book, multimedia, dan presentasi.
3. Aplikasi *Mobile Business (M-Business)* : Menyediakan fungsi penangkapan, pemrosesan, dan pengiriman data.

4. *Games* dan Hiburan : Menyediakan fungsi permainan, musik video dan sebagainya.

2.2.1 Aplikasi *Mobile Business*

Aplikasi *mobile business* dikategorikan secara horisontal dan vertikal. Kategori horisontal dikategorikan berdasarkan jenis aplikasinya, sedangkan aplikasi vertikal berdasarkan jenis bisnisnya.

Secara Horisontal antara lain :

1. *Customer Relationship Management* (CRM) : CRM memiliki 5 modul dasar yaitu : *Sales Force Automation, Field Service, Customer Suport, Call Center, dan Marketing Analysis*. Secara operasional biasanya para pengguna garis depan CRM ini bekerja di lapangan terutama *salesman*. Penggunaan aplikasi *M-Business* ini, tentunya akan sangat mempercepat eksekusi proses CRM. Contohnya jika ada komplain pelanggan terhadap *customer field service* maka dengan cepat komplain tersebut bisa diproses langsung di lapangan tanpa harus menunggu di kantor.
2. *Enterprise Resource Planing* (ERP) : ERP bisa dikatakan sebagai inti dari sistem informasi sebuah organisasi. Mengingat ERP adalah suatu *backbone system*, tentunya akan sangat menguntungkan sekali jika koneksinya ditingkatkan ke level *mobile*. Para pekerja bermobilitas tinggi tidak akan kerepotan lagi untuk menyelesaikan laporan dan proses-proses lainnya dimana dan kapan saja melalui piranti *mobile*.

3. *Transportation Management System (TMS)* : Fungsi utama TMS adalah untuk mengatur *utility loading* (Pemuatan), *routing* (Perjalanan) dan estimasi pengiriman. Penggunaan aplikasi *mobile* akan sangat meningkatkan kecepatan proses TMS. Contohnya, sebuah *transport* yang diatur untuk mengantarkan barang, akan dengan mudah mengirimkan status posisi dan kondisinya melalui perangkat *mobile*.

Selain kategori horisontal, aplikasi *mobile business* juga dikategorikan secara vertikal. Beberapa kategori vertikal antara lain :

1. Jasa Keuangan
2. Kesehatan
3. *Customer Care*
4. Manufaktur
5. Distribusi

Sekarang ini pemakaian *mobile device* atau *smart device* semakin marak. Dengan menggunakan VB.NET 2003, kita dapat membuat aplikasi untuk *smart device*.

2.3 Visual Basic.Net (*Mobile Application*)

PDA (*Personal Digital Asssistants*) merupakan komputer genggam yang awalnya dibuat untuk *organizer* pribadi yang biasanya memiliki fitur dasar : buku tanggal, buku alamat, daftar tugas dan memo. Terdapat juga fitur tambahan untuk melakukan sinkronisasi informasi dari PDA ke PC. Ada tiga sistem operasi yang saling bersaing dalam pasar PDA yaitu Palm, Microsoft Pocket PC dan Symbian.

Microsoft Visual Studio.NET 2003 menyediakan Smart Device Extention (SDE), yang dapat digunakan untuk

membuat aplikasi bagi Pocket PC dan Microsoft Windows CE. Membuat aplikasi menggunakan SDE dari VB.NET 2003, sama seperti membuat aplikasi Windows umumnya yaitu membuat proyek baru SDE, menambahkan kontrol ke form dan menulis kode. Jika tidak memiliki perangkat keras PDA dengan sistem operasi pocket PC, maka tidak perlu khawatir untuk mencoba aplikasi ini karena Microsoft sudah menyediakan emulator Pocket PC.

Untuk membuat aplikasi *mobile*, programmer dapat menerapkan hampir semua fitur yang ada pada Visual Basic .NET seperti fitur bahasa berorientasi objek yang memudahkan programmer Visual Basic .NET untuk membuat aplikasi *mobile* yang berjalan pada sistem operasi *mobile* seperti Windows CE atau Pocket PC.

2.4 Windows CE

Windows CE adalah perangkat lunak keluaran Microsoft yang dirancang untuk perangkat *organizer*, *handheld computer*, dan bahkan untuk mesin game.

Untuk menjangkau pasar yang berbeda Microsoft mengambil inti dari sistem operasi Windows CE dan memproduksi kategori yang berbeda seperti *Handled PC*, *Pocket PC*, dan *Auto PC*. Saat ini *Pocket PC* menjadi lebih populer dengan *form* dari Windows CE dan tampilan layar 320 x 240.

Beberapa fitur yang tersedia pada Windows CE pada kegiatan sehari-hari seperti :

1. Akses *email* lewat GPRS
2. Akses *internet*, LAN, modem
3. Koneksi ke alat lainnya melalui *Bluetooth*, *Infrared*, dan *WiFi*

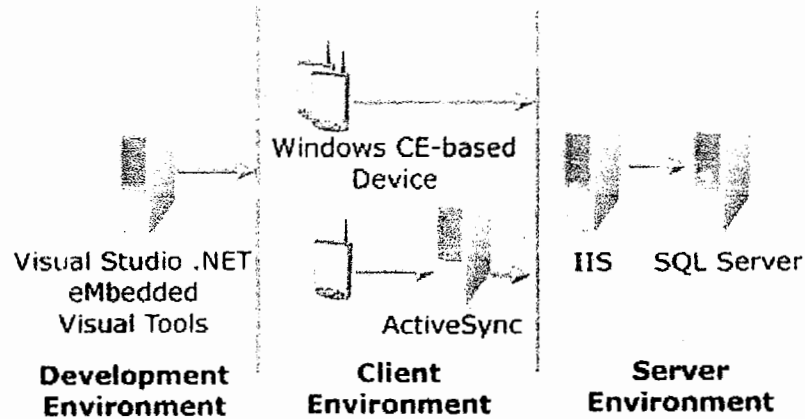
Keunggulan utama dari Windows CE adalah kompatibilitas dengan Microsoft Windows dalam komunikasi data, penanganan transfer file, sinkronisasi, dan akses pada basis data sehingga *Pocket PC* dengan berbasis Windows CE sudah mulai dipakai pada beberapa *PDA* (*Personal Digital Assistant*) terbaru.

Untuk membuat komunikasi antara Windows CE dan komputer desktop dalam pembuatan software, diperlukan suatu software dari Microsoft yang bernama ActiveSync. Dengan aplikasi tersebut pengguna *Pocket PC* dapat menginstall/menghapus aplikasi, membackup data, restore data, sinkronisasi email/file/kontak, remote data access, dsb

Seperti halnya dengan sistem operasi windows yang lain, mesin dengan Windows CE dapat diprogram dengan menggunakan banyak bahasa. Misalnya memprogram dengan bahasa C dengan melakukan pemanggilan standar Win32 API, atau dengan menggunakan bahasa C++ dan visual basic, bahkan dimungkinkan untuk membuat program di mesin *Pocket PC* dengan menggunakan bahasa Java.

2.5 SQL CE 2.0

SQL CE 2.0 merupakan sebuah solusi untuk *mobile database* yang dikeluarkan oleh Microsoft. Dengan menggunakan SQL server CE edition maka dapat dilakukan akses basis data pada *mobile device*, yang memungkinkan perubahan data pada perangkat *mobile* dengan sistem operasi Windows CE atau *Pocket PC*. Berikut adalah gambar *environment* dimana SQL CE digunakan



Gambar 2.5.1 SQL CE Environment

1. Development Environment

Ini adalah komputer dimana aplikasi *mobile* dibuat dengan menggunakan tools Visual Studio.Net.

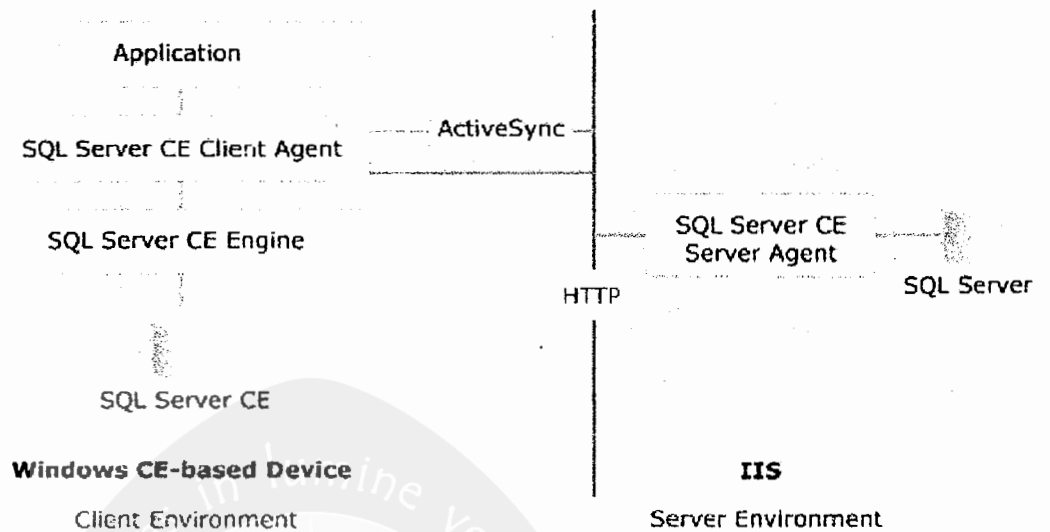
2. Client Environment

Ini adalah perangkat dimana aplikasi *mobile* diimplementasikan. Pada perangkat dimana tidak terdapat koneksi seperti GPRS dan Bluetooth, maka digunakan Microsoft ActiveSync untuk melakukan koneksi dengan kabel data yang memungkinkan terjadi sinkronisasi data antara *server* dan *client*.

3. Server Environment

Ini adalah bagian dari sistem dimana pada perangkat ini terpasang *Microsoft Internet Information Services (IIS)* dan *SQL server 2000* yang merupakan basis data induk dari sistem. IIS digunakan untuk berkomunikasi antara basis data pada *mobile device* dan basis data pada *server*. Untuk berkomunikasi diperlukan metode *Remote Data Access (RDA)* atau *merge replication* yang keduanya berkomunikasi melalui IIS.

Berikut ini ilustrasi hubungan antara SQL Server CE (*Client*) dengan SQL server (*Server*)



Gambar 2.5.2 Arsitektur Hubungan antara SQL Server CE dengan SQL Server

Pada sisi *client mobile device*, SQL server CE engine bertugas mengelola penyimpanan data pada basis data lokal SQL Server CE termasuk *track record* yaitu penambahan, pengubahan dan penghapusan data ketika data disinkronisasi dengan data yang ada pada server. SQL Server Client Agent adalah komponen yang digunakan untuk berkoneksi dengan server. Pada SQL CE ini, di implementasikan dengan menggunakan SQL CE objek seperti SQL CE engine dan *Remote Data Access (RDA)* objek. Pada sisi server, SQL Server CE Server Agent berperan sebagai perantara untuk SQL Server dengan SQL CE untuk berkomunikasi. Komunikasi dilakukan dengan HTTP melalui web server.

2.6 Remote Data Access (RDA)

Merupakan salah satu solusi koneksi basis data pada SQL CE yang menggunakan *authentication, authorization,*

dan *encryption services* dari *Microsoft Internet Information Services (IIS)*. Koneksi dengan RDA dapat dilakukan menggunakan *Microsoft ActiveSync* dengan kabel data yang memungkinkan terjadinya sinkronisasi data antara *SQL CE* dengan *SQL Server*. Hal ini dapat mengatasi kekurangan dari *mobile device* yang tidak memiliki koneksi seperti *GPRS* dan *bluetooth*.

RDA dapat digunakan dengan dua cara yaitu *client* dapat mengambil data pada server untuk disimpan pada basis data lokal, kemudian *client* dapat melakukan manipulasi secara *offline* pada basis data lokal di *mobile device*. Setelah itu *client* dapat mengembalikan data pada yang telah dimanipulasi ke server untuk disinkronisasi.

Implementasi untuk RDA dapat dilakukan melalui 2 cara yaitu:

1. *Pull method*

Pull method digunakan untuk menarik data dari server dan menyimpannya pada basis data lokal di *mobile device*.

2. *Push Method*

Push method digunakan untuk mengembalikan data ke server dari basis data lokal pada *mobile device* untuk digabungkan dengan basis data yang ada pada server.

2.7 Model Arsitektur Smart Client

2.7.1 Konsep Smart Client

Menurut definisi linguistik, smart client terdiri dari dua kata yaitu smart yang berarti cerdas, dan client yang dalam bahasa komputer dimaknai dengan suatu

program komputer yang memiliki kemampuan untuk memanipulasi, mengelola, menjalankan sekumpulan fungsi, dan dapat meminta sekumpulan layanan aplikasi yang dimiliki server.

Smart Client mengkombinasikan dua buah teknologi yang telah terlebih dahulu dikenal oleh para pengguna komputer, yakni Rich Client dan Thin Client. Rich Client adalah aplikasi *stand-alone* yang semenjak dekade 90-an dikenal oleh masyarakat seperti aplikasi pengolah kata Notepad. Sementara Thin Client dikenal saat internet tengah berkembang. Aplikasi Thin Client pada umumnya disimpan di web server dan diakses melalui web browser. Aplikasi Smart Client adalah suatu alternatif yang tepat untuk aplikasi Thin Client. Aplikasi ini menyediakan pengguna dengan suatu *user interface* yang *responsive*, mampu bekerja secara *offline*, dan dapat mengoptimalkan *local hardware* dan sumber daya lokal. Di sisi lain, Smart Client dirancang untuk dijalankan pada suatu *client device*, termasuk desktop PCs, dan dapat diterapkan pada aplikasi *mobile* seperti Pocket PCs dan Smartphone. Smart Client memungkinkan user untuk mengakses informasi dan layanan secara *remote*, dan merupakan sebuah solusi yang efektif dan fleksibel untuk aplikasi yang berorientasi pada user demi meningkatkan produktivitas dan kepuasan user.

Sebuah aplikasi dikatakan mengikuti arsitektur smart client bila memiliki beberapa karakteristik kemampuan berikut :

- Data terkoneksi dengan sumber daya luar seperti server database, jaringan internet, hingga layanan web.

- Memiliki kemampuan bekerja *offline* dikala koneksi internet dalam keadaan terputus, termasuk di dalamnya kemampuan untuk melakukan sinkronisasi data pada saat kembali *online*.
- Memudahkan dalam instalasi, pengelolaan, dan pembaruan sistem.
- Tampilan antarmuka pengguna yang lebih fleksibel, performa yang responsif, dan mendukung operasi-operasi umum yang dimiliki sistem operasi.
- Fleksibilitas piranti, smart client menekankan tingkat fleksibilitas yang khusus digunakan untuk piranti tertentu seperti notebook, pocket PC, mobile phone. Dengan demikian, aplikasi smart client dapat memanfaatkan teknologi yang khusus diimplementasikan pada piranti tersebut seperti penggunaan touch pen, dll.

2.7.2 Jenis Smart Client

Secara umum terdapat tiga jenis Smart Client, yakni:

1. *Windows Smart Client Applications*. Aplikasi smart client yang berjalan di atas sistem operasi Windows. Tipe aplikasi seperti ini sangat cocok untuk aplikasi yang berjalan di desktop, laptop, atau tablet PC.
2. *Office Smart Client*. Aplikasi smart client yang menekankan pada produktivitas perkantoran di skala enterprise. Dengan solusi ini, sumber data dapat diintegrasikan dan diakses melalui *web services* dengan fitur dari Word 2003, Excel 2003, InfoPath 2003 atau *office application* lain untuk pengembangan solusi smart client.

3. *Mobile Smart Client*. Aplikasi smart client yang dikembangkan untuk Pocket PC dan *Smartphones*.

2.7.3 **Arsitektur *Disconnected***

Arsitektur *disconnected* adalah arsitektur tanpa mode koneksi internet (*offline*) yang memiliki karakteristik penggunaan *local resources*, menyediakan *client device flexibility*, menyediakan mekanisme sinkronisasi data dan *intelligent install and update*. Dalam arsitektur *disconnected* ini *client* mengambil data pada server untuk disimpan pada basis data lokal, kemudian *client* melakukan manipulasi secara *offline* pada basis data lokal di *mobile device*. Setelah itu *client* dapat mengembalikan data pada yang telah dimanipulasi ke server untuk disinkronisasi.

Arsitektur *disconnected* ini mendukung konsep arsitektur smart client yang dikembangkan pada piranti *mobile* seperti pada Pocket PC atau *Smartphones*.