

BAB III

LANDASAN TEORI

3.1 Konsep Dasar Sistem Informasi

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu, (Scott, 1996). Menurut Robert dalam (Jogiyanto, 1999), Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lanjut. (Jogiyanto,1999). Informasi dapat dihasilkan dari sistem informasi (Information System) atau disebut juga *processing system* atau *information processing system* atau *information generation system*.

Menurut Leitch dan Davis R. dalam Jogiyanto (1990) sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan laporan-laporan tertentu kepada pihak luar. Laporan tersebut dapat digunakan sebagai informasi untuk mengambil sebuah keputusan.

Sistem informasi adalah sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar

tertentu dengan laporan-laporan yang diperlukan. Sistem informasi manajemen merupakan penerapan sistem informasi di dalam organisasi untuk mendukung informasi-informasi yang dibutuhkan oleh semua tingkatan manajemen. (Jogiyanto,1999)

Sistem informasi ini dapat juga di artikan sebagai suatu sistem yang terdiri dari komponen-komponen atau blok-blok yang berinteraksi satu sama lain membentuk satu kesatuan mencapai sasaran. Komponen atau blok-blok yang dimaksut adalah:

a) Blok Masukan

Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

b) Blok Model

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

c) Blok Keluaran

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

d) Blok Teknologi

Teknologi merupakan “kotak alat” (*toolbox*) dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan.

e) Blok Basis Data

Basis data (*database*) adalah merupakan kumpulan dari data yang saling berhubungan satu dengan lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan di dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa, supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpannya. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak paket yang disebut dengan DBMS (*Database Management System*).

f) Blok Kendali

Banyak hal yang dapat merusak sistem informasi, seperti misalnya bencana alam, api, temperatur, air, debu, kekurangan-kekurangan, kegagalan-kegagalan sistem itu sendiri, kesalahan-kesalahan, ketidak efisienan, sabotase, dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung diatasi.

3.2 Analisa Sistem Informasi

Analisa Sistem Informasi adalah penggunaan dari Sistem Informasi ke dalam bagian sub sistem untuk mengidentifikasi dan mengevaluasi permasalahan, hambatan, kesempatan yang terjadi serta kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan.

Tahap analis sistem dilakukan setelah tahap perencanaan sistem dan sebelum tahap desain sistem. Tahap analisis merupakan tahap yang kritis dan sangat penting, karena kesalahan dalam tahap ini akan sangat berpengaruh pada tahap selanjutnya.

Langkah-langkah untuk menganalisis sistem tersebut adalah :

- 1) Mengidentifikasi masalah
- 2) Memahami kerja sistem yang ada
- 3) Menganalisis sistem
- 4) Membuat laporan hasil analisis

3.3 Perancangan Sistem Informasi

Perancangan Sistem Informasi adalah proses penyusunan atau mengembangkan sistem informasi yang telah ada atau baru. Dalam tahap ini harus dapat dipastikan bahwa semua persyaratan untuk menghasilkan sistem informasi dapat dipenuhi. Hasil sistem yang dirancang harus sesuai kebutuhan pemakai untuk mendapatkan informasi.

Perancangan sistem harus mampu memberikan gambaran-gambaran yang jelas dan yang berguna serta lengkap kepada programer serta ahli-ahli teknik yang terlibat. Hal ini perlu diperhatikan adalah bahwa sistem yang disusun harus dapat berkembang lagi.

Dalam perencanaan dan pembuatan suatu sistem, langkah-langkah yang harus dilakukan adalah :

- 1) Pembuatan bagan alur sistem

- 2) Bagan berjenjang

- 3) Data Flow Diagram

- 4) *Entity Relation Diagram*

3.4 Desain Sistem

Setelah tahap analisis dan perancangan sistem selesai dilakukan, maka analis sistem telah mendapat gambaran dengan jelas apa yang harus dikerjakan. Lalu tahap selanjutnya yaitu desain sistem.

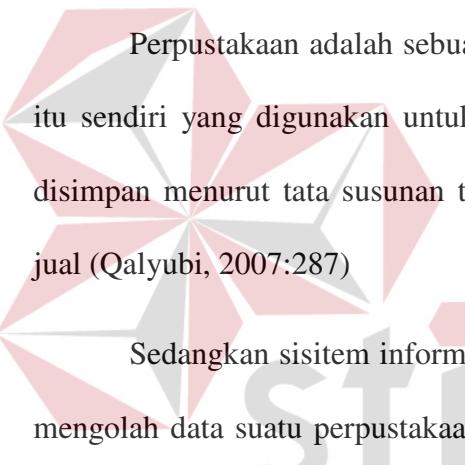
Desain sistem adalah tahap setelah analisis dari siklus pengembangan sistem pendefisian dari kebutuhan-kebutuhan fungsional dan persiapan untuk rancang bangun implementasi, menggambarkan bagaimana suatu sistem dibentuk.

Analisis sistem dapat mendesain model dari sistem informasi yang diusulkan dalam bentuk *physical system* dan *logical model*. Bagan alir sistem (*system flowchart*) merupakan alat yang tepat digunakan untuk menggambarkan *physical system*. Simbol

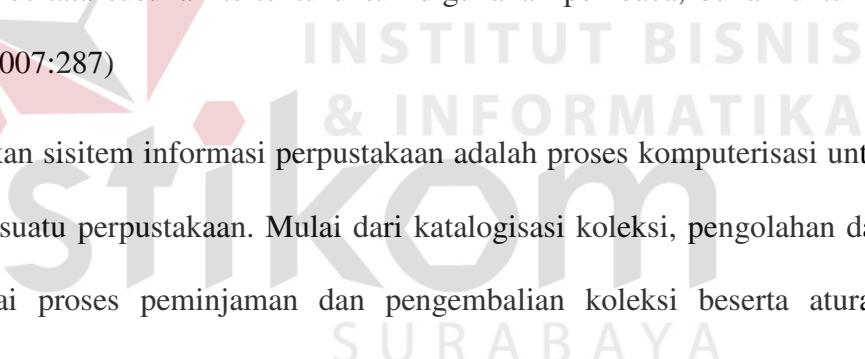
simbol bagan alir sistem ini menunjukkan secara tepat arti fisiknya, seperti simbol terminal, hard disk, laporan-laporan.

Logical model dari sistem informasi lebih menjelaskan kepada user bagaimana nantinya fungsi-fungsi di sistem informasi secara logika akan bekerja. Logical model dapat digambarkan dengan menggunakan arus data (*data flow diagram*).

3.5 Perpustakaan



Perpustakaan adalah sebuah ruangan, bagian sebuah gedung, maupun gedung itu sendiri yang digunakan untuk menyimpan buku dan terbitan lainnya yang bisa disimpan menurut tata susunan tertentu untuk digunakan pembaca, bukan untuk dijual (Qalyubi, 2007:287)



Sedangkan sistem informasi perpustakaan adalah proses komputerisasi untuk mengolah data suatu perpustakaan. Mulai dari katalogisasi koleksi, pengolahan data anggota, sampai proses peminjaman dan pengembalian koleksi beserta aturan-aturannya dan lamanya peminjaman buku.

Katalogisasi

Proses pembuatan katalog tidaklah sama pada setiap perpustakaan. Begitu juga standart yang dipakai tidak selalu seragam. Hal ini disebabkan oleh keadaan dan sejarah dari perpustakaan itu sendiri. Akan tetapi, menurut pengalaman akan lebih baik meneruskan cara-cara katalogisasi perpustakaan yang sedang berjalan daripada mengubahnya kembali menurut sistem yang dianggap lebih baik.

Katalog perpustakaan merupakan gambaran singkat semua koleksi yang ada pada perpustakaan (Suherlan, 2008:41). Didalamnya dicantumkan nama pengarang, judul, subyek dan lain-lain.

Fungsi katalog adalah sebagai berikut :

- a. sebagai inventaris koleksi yang ada di perpustakaan
- b. memberikan kemudahan kepada pemakai perpustakaan yang hanya mengetahui pengarang, judul, atau subjeknya saja untuk menemukan bahan pustaka.
- c. Memberikan deskripsi singkat kandungan bahan pustaka terutama buku.
- d. Sebagai alat bantu untuk mencari lokasi bahan pustaka yang disusun dalam rak.

3.6 Bagan Alir Dokumen

Bagan alir dokumen (*document flowchart*) atau disebut juga bagan alir formulir (*form flowchart*) atau *paperwork flowchart* merupakan bagan alir yang menunjukkan dokumen gambaran arus data dengan menggunakan simbol seperti pada tabel 3.1 berikut :

NO	SIMBOL	NAMA SIMBOL FLOWCHART	FUNGSI
1.		Dokumen	Untuk menunjukkan dokumen input dan output baik untuk proses manual, mekanik, atau komputer.
2.		Proses Komputerisasi	Menunjukkan kegiatan dari operasi program komputer.
3.		Database	Untuk menyimpan data.
4.		Penghubung	Menunjukkan hubungan di halaman yang sama.
5.		Penghubung Halaman Lain	Menunjukkan hubungan di halaman lain.
6.		Terminator	Menandakan awal /akhir dari suatu sistem.
7.		Decision	Menggambarkan logika keputusan dengan nilai <i>true</i> atau <i>false</i> .
8.		Kegiatan Manual	Untuk menunjukkan pekerjaan yang dilakukan secara manual.
9.		Simpanan Offline	Untuk menunjukkan file non komputer yang diarsip urut angka .
10.		Catatan	Menunjukkan data catatan

Gambar 3.1 Tabel Dokument Flowchart

3.7 Entity Relationship Diagram (ERD)

Entity relationship diagram adalah suatu bentuk perencanaan database secara konsep fisik yang nantinya akan dipakai sebagai kerangka kerja dan pedoman dari struktur penyimpanan data. ERD digunakan untuk menggambarkan model hubungan data dalam sistem, dimana di dalamnya terdapat hubungan entitas beserta atribut relasinya dan mendokumentasikan kebutuhan-kebutuhan untuk sistem pemrosesan data. ERD memiliki beberapa jenis model yaitu :

NO	JENIS ERD	KETERANGAN
1.	<i>Conceptual Data Model (CDM)</i>	Merupakan model <i>universal</i> dan dapat menggambarkan semua struktur <i>logic database</i> (DBMS), dan tidak bergantung dari <i>software</i> atau pertimbangan struktur <i>data storage</i> . Sebuah CDM dapat diubah langsung menjadi PDM.
2.	<i>Physical Data Model (PDM)</i>	Merupakan model ERD yang mengacu pada pemilihan <i>software</i> DBMS yang spesifik. Hal ini seringkali berbeda secara signifikan dikarenakan oleh struktur tipe <i>database</i> yang bervariasi, dari model schema, tipe data penyimpanan dsb.

Gambar 3.2 Tabel Jenis ERD

ERD memiliki 4 jenis obyek, yaitu :

1. *Entity*

Sesuatu yang ada dan terdefinisikan bisa berupa nyata maupun abstrak yang dapat dibedakan satu dengan yang lainnya dan adanya hubungan saling ketergantungan.

Ada 2 macam tipe entity, yaitu :

a. *Strong Entity*

Strong Entity merupakan tipe entity yang mempunyai *key attribute* untuk setiap individu yang ada didalamnya.

b. *Weak Entity*

Strong Entity merupakan entity yang tidak memiliki *key attribute*, oleh karena itu *weak entity* harus dihubungkan dengan *strong entity* untuk menggunkan atribut kunci secara bersama-sama.

2. *Attribute*

Setiap *entity* memiliki beberapa *attribute*, yang merupakan ciri atau karakteristik dari *entity* tersebut. *Attribute* sering disebut juga data elemen atau data field.

3. *Key*

Beberapa elemen data memiliki sifat, dengan mengetahui nilai yang telah diberikan oleh sebagian elemen data dari *entity* tertentu, dapat diidentifikasi nilai-nilai yang terkandung dalam elemen-elemen data lain ada entity yang sama. Elemen penentu tersebut adalah sebagai elemen data kunci (*key*).

4. *Relationship*

Relationship menggambarkan hubungan yang terjadi antar *entity* yang mewujudkan pemetaan antar *entity*. Bentuk *relationship* yaitu :

a. *One to One Relationship*

Hubungan satu entity dengan satu entity yang lain.

b. *One to Many Relationship*

Hubungan antar entity satu dengan entity yang lainnya adalah satu berbanding banyak.

3.8 Data Flow Diagram (DFD)

Data Flow Diagram atau yang sering disebut *Bubble Chart* atau diagram, model proses, digram alur kerja atau model fungsi adalah alat pembuatan model yang memungkinkan profesional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alir data baik secara manual maupun komputerisasi. DFD merupakan alat pembuat model yang sering digunakan untuk menjelaskan aliran informasi dan transformasi data yang bergerak dari pemasukan data hingga keluaran. Untuk memudahkan proses pembacaan DFD, maka penggambaran DFD disusun berdasarkan tingkatan atau level dari atas ke bawah, yaitu :

1. *Context Diagram*

Merupakan diagram paling atas yang terdiri dari suatu proses dan menggambarkan ruang lingkup proses. Hal yang digambarkan dalam diagram konteks adalah hubungan terminator dengan sistem dan juga sistem dalam suatu proses. Sedangkan hal yang tidak digambarkan dalam *Context Diagram* adalah hubungan antar terminator dan data *source*.

2. *Diagram Zero* (Level 0)

Merupakan diagram yang berbeda diantara diagram konteks dan diagram detail serta menggambarkan proses utama dari DFD. Hal yang digambarkan dalam *Diagram Zero* adalah proses utama dari sistem serta hubungan entity, proses, alur data dan data *source*.

3. Diagram Detail (Primitif)

Merupakan penguraian dalam proses yang ada dalam Diagram Zero. Diagram yang paling rendah dan tidak dapat diuraikan lagi.

Data Flow Diagram (DFD) memiliki empat komponen, yaitu :

1. Terminator

Terminator atau *External Entity* atau kesatuan luar mewakili entitas external yang berkomunikasi dengan sistem yang dikembangkan. Terminator merupakan kesatuan di lingkungan sistem yang dapat berupa orang atau sistem yang berada di lingkungan luar sistem yang memberikan inputan maupun yang menerima output dari sistem serta berupa bagian atau divisi diluar sistem yang berkomunikasi dengan sistem. Terminator ini sering juga disebut entitas (*external*), sumber atau tujuan (*source and sink*).

2. Proses

Proses sering dikenal dengan *Bubble*, fungsi atau informasi. Komponen proses menggambarkan bagian dari sistem yang mentransformasikan input ke output, atau dapat dikatakan bahwa komponen proses menggambarkan transformasi satu

inputan atau lebih menjadi output dari sistem. Dilambangkan dengan lingkaran atau empat persegi panjang tegak dengan sudut tumpul. Proses diberi nama untuk menerangkan proses atau kegiatan apa yang sedang atau kegiatan yang akan dilaksanakan. Dan setiap proses harus diberi penjelasan lengkap sebagai berikut :

a. Identifikasi Proses

Umumnya berupa angka yang menunjukkan nomor dari proses dan ditulis pada bagian atas simbol.

b. Nama Proses

Menunjukkan apa yang sedang dikerjakan oleh proses tersebut. Nama proses harus jelas dan lengkap menggambarkan bagian prosesnya. Nama proses diletakkan dibawah identifikasi proses.

3. Data Store

Data Store digunakan sebagai sarana untuk pengumpulan data. Data Store disimbolkan dengan dua garis horizontal yang paralel dimana tertutup pada satu ujungnya atau dua garis horizontal. Suatu nama perlu diberikan pada data store menunjukkan nama dari filenya. Data Source biasanya berkaitan dengan penyimpanan file atau database yang dilakukan secara terkomputerisasi. Data Store dihubungkan dengan alur data hanya pada komponen proses pengertiannya sebagai berikut :

- a. Alur data dari store yang berarti sebagai pengaksesan data untuk suatu proses.
- b. Alur data ke proses berarti meng-update data seperti menambah data, mengurangi data maupun mengubah data.

4. Alur Data

Alur data dapat digambarkan dengan anak panah yang menuju ke dalam proses maupun ke luar proses. Alur data digunakan untuk menerangkan perpindahan data atau informasi dari suatu bagian ke bagian lainnya.

5. Syarat-syarat sebuah DFD :

- a. Pemberian nama untuk setiap komponen DFD.
- b. Pemberian nomor pada proses DFD.
- c. Penggambaran DFD serapi mungkin.
- d. Menghindari pembuatan DFD yang rumit.
- e. Memastikan DFD dibangun secara konsisten.

3.9 Database

Database adalah suatu sistem menyusun dan mengelola *record-record* menggunakan komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap dengan sebuah organisasi/perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses pengambil keputusan (Linda, 2004:1). Database dapat dinyatakan sebagai suatu sistem yang memiliki karakteristik seperti berikut:

- a. Merupakan suatu kumpulan interaksi data yang disimpan bersama dan tanpa mengganggu satu sama lain atau membentuk duplikat data.
- b. Kumpulan data di dalam database dapat digunakan oleh sebuah program secara optimal.
- c. Penambahan data baru, modifikasi dan pengambilan kembali dari data dapat dilakukan dengan mudah dan terorganisasi.

Dalam arsitektur database terdapat tiga tingkatan yang saling mendukung. Di bawah ini adalah penjelasannya yaitu:

- a. *Internal level* yaitu tingkat yang basis datanya secara fisik ditulis atau disimpan di media storage dan level yang berkaitan.
- b. *External level* disebut juga individual user views, yaitu tingkat yang basis datanya dapat berdasarkan kebutuhan masing-masing aplikasi di user atau level yang berkaitan dengan para pemakai.
- c. *Conceptual level* disebut juga *community user view*, yaitu tingkat *user view* dari aplikasi yang berbeda digabungkan sehingga menggunakan basis data secara keseluruhan dengan menyembunyikan penyimpanan data secara fisik yang merupakan penghubung dari *internal level* dan *external level*.

Seluruh operasi yang dilakukan pada database didasarkan atas tabel-tabel dan hubungannya. Dalam model relasional dikenal antara lain *table*, *record*, *field*, *indeks*, *query* penjelasannya seperti dibawah ini:

- a. Tabel atau entity dalam model relasional digunakan untuk mendukung antar muka komunikasi antara pemakai dengan profesional komputer.
- b. *Record* atau baris atau dalam istilah model relasional yang formal disebut tuple adalah kumpulan data yang terdiri dari satu atau lebih.
- c. *Field* atau kolom atau dalam istilah model relasional yang formal disebut dengan attribute adalah sekumpulan data yang mempunyai atau menyimpan fakta yang sama atau sejenis untuk setiap baris pada *table*.
- d. *Indeks* merupakan tipe dari suatu table tertentu yang berisi nilai-nilai field kunci atau field.
- e. Query merupakan sekumpulan perintah *Structure Query Language* (SQL) yang dirancang untuk memanggil kelompok *record* tertentu dari satu tabel atau lebih untuk melakukan operasi pada tabel.

3.10 Microsoft Visual Basic.Net 2005

Visual Basic (VB) 2005 merupakan bahasa pemrograman yang terdapat dalam satu paket aplikasi Visual Studio 2005. Visual Studio 2005 merupakan suatu produk Microsoft yang merupakan penerus dari Visual Studio 2003. (Budiharto, 2006:1) menyebutkan, “Visual Basic 2005 ialah bahasa pemrograman terbaru yang memudahkan programmer VB 6/VB.Net beralih ke VB 2005”. (Budiharto, 2006:3-4) juga menyebutkan alasan penting lainnya untuk melakukan migrasi VB 2005, yaitu :

- a. Visual Basic 2005 mengatasi semua masalah yang sulit di sekitar pengembangan aplikasi berbasis Windows dan mengurangi penggunaan aplikasi lainnya serta versi komponen, bahkan mewarisi sifat C++ dan berbau Java.
- b. Visual Basic 2005 memiliki fasilitas penanganan bug yang hebat dan *real time background compiler* yang mengakibatkan developer visual C# dapat mengetahui kesalahan kode yang terjadi secara *up-to-date*.
- c. *Windows Form designer* memungkinkan developer memperoleh aplikasi desktop dalam waktu yang singkat.
- d. Bagi developer, Visual Basic 2005 menyediakan model pemrograman data akses ActiveX Data Object (ADO) yang sudah dikenal dan diminati, ditambah XML baru yang berbasis Microsoft ADO.Net dengan ADO.Net, developer akan memperoleh akses ke komponen yang lebih powerfull, seperti control DataSet.
- e. Visual Basic 2005 menghasilkan web. Menggunakan form web yang baru, anda dapat dengan mudah membangun *thin – client* aplikasi berbasis web yang secara cerdas berjalan di browser dan platform manapun.
- f. Mendukung pembangunan Aplikasi *client-server*, terdistribusi serta berupa aplikasi yang berbasis Windows serta web.
- g. *.NET Framework* secara mendasar dibuat untuk dipasangkan dengan Windows 2003 dengan keunggulan memonitor kelalaian dari aplikasi yang sedang berjalan, dan mengisolasi setiap aplikasi yang sedang berjalan dan mengisolasi setiap aplikasi.

- h. *Developer* dengan berbagai latar belakang dapat dengan segera menguasai .NET karena kemudahan dan kemiripan kode yang ditawarkannya.
- i. *Deployment/Penyebaran* yang mudah, baik untuk aplikasi windows maupun aplikasi web karena sudah tersedia wizard atau tool secara khusus dengan fasilitas tambahan yang menarik. Tool canggih ini tidak tersedia pada aplikasi sebelumnya bahkan pada bahasa pemrograman lain.
- j. Integrasi dengan sistem yang sudah ada sangat mudah, Net Framework com memungkinkan Anda berinteraksi dan dengan sistem yang sudah ada menggunakan XML web Service. Terakhir, Visual Studio *Upgrate tool* yang tersedia pada Visual Studio.NET dan *Java Language Convention Assistant* membantu Anda mengkonversi Visual Basic 6 dan Visual J++ agar berjalan pada .Net Framework.
- k. Mendukung lebih dari 20 bahasa pemrograman, *NET Framework* mendukung integrasi lebih dari 20 bahasa pemrograman yang tidak terbayang sebelumnya. Memungkinkan pengembang memilih bahasa pemrograman yang tepat sesuai latar belakang pemrogramannya.

3.11 *SQL Server*

SQL Server adalah salah satu jenis *database server* yang sangat terkenal Kepopulerannya disebabkan *SQL Server* menggunakan SQL sebagai bahasa dasar untuk mengakses databasenya. *SQL Server* bersifat *Open Source* (tidak perlu

membayar untuk menggunakannya) pada berbagai *platform* (kecuali untuk jenis *Entrepise*, yang bersifat komersial).

SQL Server termasuk jenis RDBMS (*Relational Database Management System*). Itulah sebabnya, istilah seperti table, baris, dan kolom digunakan pada SQL Server. Pada SQL Server, sebuah *database* mengandung satu atau sejumlah tabel. Tabel terdiri atas sejumlah baris dan setiap baris mengandung satu atau sejumlah tabel.

