BAB II

LANDASAN TEORI

2.1 Sistem Informasi

Sistem adalah sekumpulan elemen yang dipadukan untuk mencapai satu atau lebih tujuan bersama, sedangkan informasi adalah data yang telah diproses dan dikelola hingga memiliki arti bagi penerimanya. Jadi, sistem informasi adalah sekumpulan elemen (orang, data, prosedur, mesin, dan pemroses data) yang bekerja sama untuk menghasilkan informasi yang berguna dan berarti (relevan, tepat waktu, akurat, lengkap, dan memenuhi aturan-aturan tertentu).

2.1.1 Konsep Dasar Sistem

Konsep dasar sistem ada dua pendekatan yaitu penekanan pada prosedurnya dan penekanan pada komponennya.

1. Definisi sistem yang lebih menekankan pada prosedur adalah : suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu.
    a. Suatu prosedur adalah : suatu urut-urutan operasi klerikal (tulis-menulis), biasanya melibatkan beberapa orang di dalam satu atau lebih departemen, yang diterapkan untuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis yang terjadi.
    b. Definisi lain dari prosedur adalah : urut-urutan yang tepat dari tahapan-tahapan instruksi yang menerangkan apa yang harus dikerjakan, siapa

yang mengerjakannya, kapan dikerjakan dan bagaimana mengerjakannya.

2. Definisi sistem yg lebih menekankan pada konponen/elemen adalah : kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu.
3. Karakteristik sistem yaitu : mempunyai komponen, batas sistem, lingkungan luar sistem, penghubung, masukan, keluaran, pengolah/proses, dan sasaran atau tujuan.
4. Klasifikasi sistem dari beberapa sudut pandang sebagai berikut :
    1) Sistem sebagai sitem alamiah dan sistem buatan manusia.
    2) Sistem sebagai sistem abstrak dan sistem fisik.
    3) Sistem sebagai sistem tertentu (deterministic) dan sistem tak tentu (probabilistic)
    4) Sistem sebagai sitem tertutup dan sistem terbuka.

2.1.2 Konsep Dasar Informasi

Didalam organisasi sangat penting dalam mengelola sumberdaya-sumberdaya utama seperti buruh, dan bahan mentah, tapi saat ini informasi juga merupakan sumberdaya yang tidak kalah pentingnya harus dikelola. Para pembuat keputusan memahami bahwa informasi tidak hanya sekedar produk sampingan bisnis yang sedang berjalan, namun juga sebagai bahan pengisi bisnis dan menjadi faktor kritis dalam menentukan kesuksesan atau kegagalan suatu usaha. Informasi ibarat darah yang mengalir di dalam tubuh suatu organisasi. Sehingga informasi

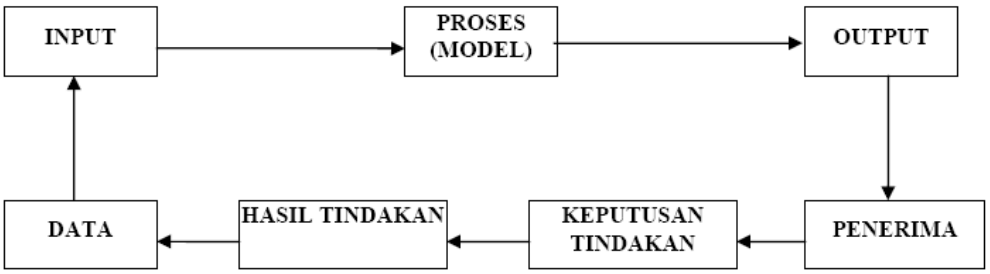
merupakan salah satu bentuk sumber daya utama dalam suatu organisasi yang digunakan oleh manager untuk mengendalikan perusahaan dalam mencapai tujuan.

2.1.2.1 Definisi informasi

Data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya. Data adalah : kenyataan yang menggambarkan suatu kejadian–kejadian dan kesatuan yang nyata. Atau data adalah : representasi dunia nyata yang mewakili suatu objek seperti manusia (pegawai, mahasiswa, pelanggan), hewan, peristiwa, konsep, keadaan dll, yang direkam dalam bentuk angka, huruf, simbol, teks, gambar, bunyi atau kombinasinya.

2.1.2.2 Siklus informasi

Data yang masih merupakan bahan mentah yang harus diolah untuk menghasilkan informasi melalui suatu model. Model yang digunakan untuk mengolah data tersebut disebut model pengolahan data atau dikenal dengan siklus pengolahan data (siklus informasi).



Gambar 2.1 Siklus Informasi

2.1.3 Konsep Dasar Sistem Informasi

Informasi merupakan hal yang sangat penting bagi manajemen di dalam pengambilan keputusan.

2.1.3.1 Definisi sistem informasi

Suatu 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.

2.1.3.2 Definisi lain sistem informasi

Kerangka kerja yang mengkoordinasikan sumber daya (manusia dan komputer) untuk mengubah masukan (input) menjadi keluaran (informasi) guna mencapai sasaran-sasaran perusahaan.

2.2 Klasifikasi Sistem

2.2.1 Sistem Abstrak dan Sistem Fisik

Sistem abstrak (*abstract system*) adalah system yang berisi gagasan atau konsep. Misalnya, sistem teologi yang berisi gagasan tentang hubungan manusia dan Tuhan. Sedangkan sistem fisik (*physical system*) adalah sistem yang secara fisik dapat dilihat. Misalnya, sistem komputer.

2.2.2 Sistem Deterministik dan Probabilistik

Sistem deterministik (*deterministic system*) adalah suatu sistem yang operasinya dapat diprediksi secara tepat. Misalnya, sistem komputer. Sedangkan

sistem probabilistik (*probabilistic system*) adalah sistem yang tak dapat diramal dengan pasti karena mengandung unsur probabilitas. Misalnya, sistem arisan.

2.2.3 Sistem Tertutup dan Terbuka

Sistem tertutup (*closed system*) adalah sistem yang tidak dapat bertukar materi, informasi, atau energi dengan lingkungan. Dengan kata lain, sistem ini tidak berinteraksi dan tidak dipengaruhi oleh lingkungan. Misalnya, sistem penerimaan siswa baru (SPMB). Sistem terbuka (*open system*) adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan. Cirri-cirinya, sistem menerima masukan yang diketahui, yang bersifat acak, maupun gangguan. Selain itu, umumnya sistem melakukan adaptasi terhadap lingkungan. Misalnya, sistem perusahaan dagang.

2.2.4 Sistem Alamiah dan Sistem Buatan Manusia

Sistem alamiah (*natural system*) adalah sistem yang terjadi karena alam (tidak dibuat oleh manusia). Misalnya, sistem tata surya. Sedangkan sistem buatan manusia (*human made system*) adalah sistem yang dibuat oleh manusia. Misalnya, sistem komputer.

2.2.5 Sistem Sederhana dan Sistem Kompleks

Berdasarkan tingkat kerumitannya, sistem dibedakan menjadi sistem sederhana (misalnya sepeda) dan sistem yang kompleks (misalnya otak manusia).

2.2.6 Kedudukan Sistem Informasi Sebagai Sistem

Sistem informasi tergolong sebagai :

1. Sistem buatan manusia,
2. Sistem terbuka,
3. Bersifat fisik, serta
4. Dapat tergolong sebagai sistem probabilistik atau deterministik (tergantung pada titik pandang untuk meninjaunya).

Sistem ini adalah sistem buatan manusia karena manusialah yang merancang dan mengoperasikannya. Tergolong sistem terbuka karena menerima masukan dan keluaran bagi lingkungan, serta beradaptasi dengan perubahan atau permintaan lingkungan. Termasuk sistem fisik karena secara fisik dapat dilihat. Sistem informasi dapat tergolong sebagai sistem kompleks atau sederhana tergantung pada implementasinya. Sistem informasi digolongkan sebagai sistem probabilistik karena kondisi masa depan sistem ini tidak adapat diramalkan dengan pasti (Wilkinson, 1992). Namun, bagian tertentu seperti sistem pemrosesan transaksi dapat dianggap sebagai sistem yang deterministik.

2.3 Basis Data

2.3.1 Definisi Basis Data

Basis data terdiri dari dua kata, yaitu basis dan data. Basis kurang lebih dapat diartikan sebagai markas atau gudang, tempat bersarang/berkumpul. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek

seperti manusia (pegawai, siswa, pembeli, pelanggan), barang, hewan, peristiwa, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, symbol, teks, gambar, bunyi, atau kombinasinya.

Basis data sendiri dapat didefinisikan dalam sejumlahsudut pandang seperti :

1. Himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.
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 elektronis.

2.3.2 Database Management System (DBMS)

Pengelolaan basis data secara fisik tidak dilakukan oleh pemakai secara langsung, tetapi ditangani oleh sebuah perangkat lunak (sistem) yang khusus. Perangkat lunak inilah yang akan menentukan bagaimana data diorganisasi, disimpan, diubah dan diambil kembali. DBMS ini juga memberikan mekanisme pengamanan data, pemakaian data secara bersamaan, pemaksaan keakuratan/konsistensi data, dan lain sebagainya.

Perangkat lunak yang termasuk DBMS diantaranya yaitu dBase III+, dBase IV, FoxBase, Rbase, MS-Access, dan Borlan Paradox (untuk kelas

sederhana) sedangkan Borland-Interbase, MS-SQLServer, CA-Open Ingres, Oracle, Informix dan Sybase (untuk kelas kompleks).

2.3.3 Bahasa Basis Data (*Database Language*)

DBMS merupakan perantara bagi pemakai dengan basis data dalam disk. Cara berinteraksi antara pemakai dengan basis data tersebut diatur dalam suatu bahasa khusus yang ditetapkan oleh perusahaan pembuat DBMS. Bahasa itu dapat kita sebut Bahasa Basis Data yangterdiri atas sejumlah perintah (*statement*) yang diformulasikan dan dapat diberikan *user* dan dikenali/diproses oleh DBMS untuk melakukan suatu aksi/pekerjaan tertentu. Contoh-contoh bahasa basis data adalah SQL, dBase, QUEL dan sebagainya.

Sebuah bahasa basis data biasanya dapat dipilah ke dalam dua bentuk, yaitu :

1. *Data Definition Language* (DDL) dan
2. *Data Manipulation Language* (DML).

2.3.4 Data Definition Language (DDL)

Struktur/skema basis data yang menggambarkan/mewakili desain basis data secara keseluruhan dispesifikasikan dengan bahasa khusus yang disebut *Data Definition Language* (DDL). Dengan bahasa inilah kita dapat membuat tabel baru, membuat indeks, mengubah tabel, menentukan struktur penyimpanan tabel, dan sebagainya. Hasil dari kompilasi perintah DDL adalah kumpulan tabel yang disimpan dalam file khusus yang disebut Kamus Data (*Data Dictionary*).

Kamus data merupakan suatu metadata (superdata) yaitu data yang mendeskripsikan data yang sesungguhnya. Kamus data ini akan selalu diakses dalam suatu operasi basis data sebelum suatu file data yang sesungguhnya diakses.

2.3.5 Data Manipulation Language (DML)

Merupakan bentuk bahasa basis data yang berguna untuk melakukan manipulasi dan pengambilan data pada suatu basis data. Manipulasi data dapat beupa :

1. Penyisipan/Penambahan data baru ke suatu basis data.
2. Penghapusan data dari suatu basis data.
3. Pengubahan data di suatu basis data.

Data Manipulation Language (DML) merupakan bahasa yang bertujuan memudahkan pemakai untuk mengakses data sebagaimana direpresentasikan oleh model data. Ada dua jenis DML yaitu :

1. Prosedural, yang mensyaratkan agar pemakai menentukan data apa yang diinginkan serta bagaimana cara mendapatkannya.
2. Nonprosedural, yang membuat pemakai dapat menentukan data apa yang diinginkan tanpa menyebutkan bagaimana cara mendapatkannya.

2.3.6 Metode Perancangan Sistem

2.3.6.1 Diagram Aliran Data

Pada saat informasi mengalir melalui perangkat lunak, dia dimodifikasi oleh suatu deretan transformasi. Diagram aliran data/*data flow diagram* (DFD) adalah sebuah teknik grafis yang menggambarkan aliran informasi dan transformasi yang diaplikasikan pada saat data bergerak dari input menjadi output. DFD tingkat o, yang disebut juga dengan model sistem fundamental atau diagram konteks, merepresentasikan seluruh elemen sistem sebagai sebuah bubble tunggal dengan data *input* (masuk) dan *output* (keluar) yang ditunjukan dengan anak panah yang masuk dan keluar secara berurutan.

2.3.6.2 Model Entity-Relationship (Model Keterhubungan-Entitas)

Pada Model Entity-Relationship, semesta data yang ada didunia nyata diterjemahkan dengan memanpfaatkan sejumlah perangkat konseptual menjadi sebuah diagram data, yang umum disebut sebagai Diagram Entity-Relationship (Diagram E-R). ada dua komponen utama pembentuk Model Entity-Relationship, yaitu Entitas (*Entity*) dan Relasi (*Relation*). Kedua komponen ini dideskripsikan lebih jauh melalui sejumlah Atribut/Properti.

Komponen Model Keterhubungan-Entitas lebih jelas adalah sebagai berikut :

1. Entitas (*Entity*) dan Himpunan Entitas (*Entity Sets*)

   Entitas merupakan individu yang mewakili sesuatu yang nyata (eksistensinya) dan dapat dibedakan dari sesuatu yang lain, sekelompok

entitas yang sejenis dan berada dalam lingkup yang sama membentuk sebuah Himpunan Entitas (*Entity Set*).

2. Atribut (*Atributes/Properties*)

   Setiap entitas pasti memiliki atribut yang mendeskripsikan karakteristik (*Properti*) dari entitas tersebut.

3. Relasi (*Relationship*) dan Himpunan Relasi (*Relationship Sets*)

   Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda, kumpulan dari semua relasi di antara entitas-entitas yang terdapat pada himpunan entitas tersebut membentuk himpunan relasi (*Relationship Sets*).

4. Kardinalitas/Derajat Relasi

   Kardinalitas relasi menunjukkan jumlah maksimum entitas yang sapat berelasi dengan entitas pada himpunan entitas yang lain. Kardinalitas relasi yang terjadi diantara dua himpunan entitas (misalnya A dan B) dapat berupa :

   a. Satu ke Satu (*One to One*)

      Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu juga sebaliknya setiap entitas pada himpunan entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B.

b. Satu ke Banyak (*One to Many*)

Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, di mana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

c. Banyak ke Satu (*Many to One*)

Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya, di mana setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B.

d. Banyak ke Banyak (*Many to Many*)

Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian juga sebaliknya, di mana setiap entitas pada himpunan entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

2.3.6.3 Diagram Entity-Relationship (Diagram E-R)

Model Entity-Relationship yang berisi komponen-komponen himpunan entitas dan himpunan relasi yang masing-masing dilengkapi dengan atribut-atribut yang merepresentasikan seluruh fakta dari dunia nyata yang kita tinjau, dapat

digambarkan dengan lebih sistematis dengan menggunakan diagram entity-relationship (Diagram E-R). Notasi-notasi simbolik di dalam diagram E-R yang dapat kita gunakan adalah :

1. Persegi panjang, menyatakan himpunan entitas.
2. Lingkaran/elips, meyatakan atribut (Atribut yang berfungsi sebagai *key* digarisbawahi).
3. Belah ketupat, menyatakan himpuanan relasi.
4. Garis, sebagai penghubung antara himpunan relasi dengan himpunan entitas dan himpunan entitas dengan atributnya.
5. Kardinalitas relasi dapat dinyatakan dengan banyaknya garis cabang atau dengan pemakaian angka (1 dan 1 untuk relasi satu-ke-satu, dan N untuk relasi satu-ke-banyak atau N dan N untuk relasi banyak-ke-banyak).

2.4 Perangkat Lunak Pendukung

2.4.1 Borland Delphi

Delphi adalah sebuah bahasa pemrograman dan lingkungan pengembangan perangkat lunak. Produk ini dikembangkan oleh Borland (sebelumnya dikenal sebagai Inprise). Bahasa Delphi, yang sebelumnya dikenal sebagai *object pascal* (pascal dengan ekstensi pemrograman berorientasi objek (PBO/OOP)) pada mulanya ditujukan hanya untuk Microsoft Windows, namun saat ini telah mampu digunakan untuk mengembangkan aplikasi untuk Linux dan Microsoft .NET framework (lihat di bawah). Dengan menggunakan Free Pascal

yang merupakan proyek opensource, bahasa ini dapat pula digunakan untuk membuat program yang berjalan di sistem operasi Mac OS X dan Windows CE.

Pada tanggal 8 Februari 2006, Borland mengumumkan akan melepas seluruh jajaran produk pengembangan aplikasi komputernya termasuk di antaranya Delphi.

2.4.1.1 Perkembangan Delphi

Umumnya delphi lebih banyak digunakan untuk pengembangan aplikasi desktop dan enterprise berbasis database, tapi sebagai perangkat pengembangan yang bersifat general-purpose ia juga mampu dan digunakan dalam berbagai jenis proyek pengembangan software. Ia juga yang dikenal sebagai salah satu yang membawa istilah *RAD* tool, kepanjangan dari Rapid Application Development, saat dirilis tahun 1995 untuk windows 16-bit. Delphi 2, dirilis setahun kemudian, mendukung lingkungan windows 32-bit, dan versi c++, C++Builder, dirilis beberapa tahun kemudian. Pada tahun 2001 sebuah versi linux yang dikenal sebagai Kylix tersedia. Dengan satu rilis baru setiap tahunnya, pada tahun 2002 dukungan untuk Linux (melalui Kylix dan CLX component library) ditambahkan dan tahun 2003 .NET mulai didukung dengan munculnya Delphi.Net (Delphi 8).

Chief Architect yang membidani Delphi, dan pendahulunya Turbo Pascal, adalah Anders Hejlsberg sampai kemudian ia pindah ke Microsoft tahun 1996 di mana ia sebagai chief designer C# dan termasuk orang kunci dalam perancangan Microsoft .Net Framework. Dukungan penuh untuk .Net ditambahkan pada

Delphi 8 (dirilis pada bulan Desember 2003) dengan penampilan user interface (look and feel) mirip dengan Microsoft Visual Studio .NET.

Delphi 2005 (nama lain dari Delphi 9) mendukung *code generation* baik untuk win32 maupun .NET, dan seperti yang telah dikenal, fitur-fitur manipulasi data secara *live* dari database secara *design-time*. Ia juga membawa banyak pembaruan pada IDE secara signifikan.

Para penganjur delphi mengklaim dengan bahasa pemrograman Delphi, IDE dan component library (VCL/CLX) yang disediakan oleh vendor tunggal memungkinkan satu paket yang lebih konsisten dan mudah dikenali.

Produk delphi ini didistribusikan dalam beberapa rancangan: Personal, Professional, Enterprise (sebelumnya Client/Server) dan Architect.

2.4.1.2 Keuntungan dan Kerugian Delphi

Delphi membawa keuntungan-keuntungan berikut:

1. Komunitas pengguna yang besar pada Usenet maupun web.
2. Dapat mengkompilasi menjadi single executable, memudahkan distribusi dan meminimalisir masalah yang terkait dengan *versioning*
3. Banyaknya dukungan dari pihak ketiga terhadap VCL (biasanya tersedia berikut source codenya) ataupun tools pendukung lainnya (dokumentasi, *tool debugging*)
4. Optimasi kompiler yang cukup cepat
5. Mendukung multiple platform dari source code yang sama

Berikut ini kerugiannya:

1. Partial single vendor lock-in (Borland dapat menetapkan standar bahasa, kompatibilitas yang harus mengikutinya)
2. Terbatasnya kamampuan portabilitas antar-platform OS (sebelum ada kylix)
3. Akses pada platform dan library pihak ketiga membutuhkan file-file header yang diterjemahkan ke dalam bahasa pascal
4. Dokumentasi atas platform dan teknik-teknik yang menyertainya sulit ditemukan dalam bahasa pascal (contoh akses COM dan Win32)

2.4.2 MySQL

Database sebuah subjek yang teramat penting. Setuju? Database ada di mana-mana—dari kita lahir, sekolah, bekerja, sampai pensiun. Seluruh aktivitas komputasi adalah seputar mengolah data—data keuangan, data percobaan ilmiah, data statistik pengunjung. Tanpa database banyak aspek kehidupan yang akan pincang dan akhirnya mati—perorangan, perusahaan, dan negara. Aplikasi apa pun yang Anda buat, entah itu shopping cart, CRM, portal komunitas, webledger, di jantungnya adalah database.

Pasar untuk database pun amat besar. Untuk mendapatkan gambaran, lihatlah Larry Ellison. Dari satu [keluarga] produk saja, yaitu Oracle, pria ini berhasil mengeruk uang hingga menjadi orang terdua terkaya di dunia setelah Bill Gates—bahkan sempat sehari menduduki tampuk pertama pada posisi $53 milyar saat harga saham Microsoft turun. Dan meski merajai lebih dari separuh pangsa pasar

untuk enterprise besar, tentu saja Oracle bukan satu-satunya produk yang tersedia bagi pengguna database, terlebih di kelas kecil dan menengah.

Ukuran pasar database tahun 2002 diperkirakan mencapai $8 milyar, dan telah bertumbuh terus lebih dari 10% dari tahun demi tahun secara konsisten. Saat ini pasar tersebut memang masih didominasi oleh hanya beberapa pemain besar saja. Selain Oracle yang menjadi brand paling top—dan produk paling menguras kocek—terdapat pula Microsoft dengan SQL Servernya, IBM dengan DB2-nya, Sybase, NCR dengan produk data mart/data warehousingnya, dan Borland dengan Interbasenya. Tapi lambat laun dan pasti, gerakan open source mulai merasuki dunia perusahaan dan menggerogoti penjualan produk-produk komersial ini. Beberapa database open source telah ada sejak bertahun-tahun lalu dan kini cukup matang untuk siap dipakai. Memang betul, belum ada database open source yang sematang atau selengkap Oracle dalam hal fitur, tapi:

1) tidak semua orang butuh semua fitur itu;
2) tidak semua orang punya uang untuk membeli Oracle. Didorong oleh kebutuhan untuk menurunkan biaya dan menaikkan interoperabilitas, beberapa organisasi besar—komersial maupun nonkomersial, sebutlah seperti NASA dan Yahoo!—mulai beralih dari modus mahal ke modus gratis. Dalam fokus kali ini kita akan berjalan-jalan dan melihat-lihat produk database open source yang ada.

2.4.2.1 Sejarah MySQL

SQL (Structured Query Language) adalah bahasa standart yang digunakan untuk mengakses server database . Semenjak tahun 70-an bahasa ini

telah dikembangkan oleh IBM, yang kemudian diikuti dengan adanya Oracle, Informix dan Sybase. Dengan menggunakan SQL, proses akses database menjadi lebih user-friendly dibandingkan dengan misalnya dBase ataupun Clipper yang masih menggunakan perintah – perintah pemrograman murni.

Selain MySQL, ada beberapa jenis pemograman yang berorientasi database yang dapat digunakan untuk aplikasi di web seperti ORACLE. Oracle merupakan sebuah perusahaan besar di dunia yang cakupan bisnis salah satunya adalah penjualan software dan pembuatan software database yang diperuntukkan bagi perusahaan-perusahaan besar di dunia. Karena softwarenya tidak bebas di "pasaran" atau tidak free software maka sebagian besar perusahaan kecil atau menengah atau programmer web masih menggunakan database MySQL sebagai software database perusahaan atau webnya.

MySQL adalah sebuah server database SQL multiuser dan multi-threaded. SQL sendiri adalah salah satu bahasa database yang paling populer di dunia. Implementasi program server database ini adalah program daemon 'mysqld' dan beberapa program lain serta beberapa pustaka.

MySQL dibuat oleh TcX dan telah dipercaya mengelola sistem dengan 40 buah database berisi 10,000 tabel dan 500 di antaranya memiliki 7 juta baris (kira-kira 100 gigabyte data). Database ini dibuat untuk keperluan sistem database yang cepat, handal dan mudah digunakan. Walaupun memiliki kemampuan yang cukup baik, MySQL untuk sistem operasi Unix bersifat freeware, dan terdapat versi shareware untuk sistem operasi windows. Menurut pembuatnya, MySQL disebut seperti "my-ess-que-ell" dan bukan my-sequel !

Sebagaimana database sistem yang lain, dalam SQL juga dikenal hirarki server dengan database-database. Tiap-tiap database memiliki tabel-tabel. Tiap-tiap tabel memiliki field-field.

Umumnya informasi tersimpan dalam tabel – tabel yang secara logik merupakan struktur 2 dimensi terdiri atas baris dan kolom. Field-field tersebut dapat berupa data seperti integer , realm char, date, time dan lainnya.

SQL tidak memiliki fasilitas pemrograman yang lengkap, tidak ada looping ataupun percabangan. Sehingga untuk menutupi kelemahan ini perlu digabung dengan bahasa pemrograman semisal C.

Berbagai kelebihannya. Antara lain:

1. Source MySQL dapat diperoleh dengan mudah dan gratis
2. Sintaksnya lebih mudah dipahami dan tidak rumit
3. Pengaksesan database dapat dilakukan dengan mudah

2.4.2.2 Perkembangan MySQL

1. Seri 3.22 MySQL mulai diadopsi banyak orang dan meningkat populasi penggunanya, maka di seri 3.23 dan 4.0-lah terjadi banyak peningkatan dari sisi teknologi. Ini tidak terlepas dari tuntutan pemakai yang semakin mengandalkan MySQL, namun membutuhkan fitur-fitur yang lebih banyak lagi. Sejak dari sebelum 3.22 dan awal 3.23, MySQL terus dikritik praktisi database maupun penggunanya mengenai tidak adanya fasilitas transaksi (**COMMIT dan ROLLBACK**) . Pengembang MySQL menyarankan penggunaan **LOOK TABLES** untuk mengatasi masalah

update atomik, tapi ini tentu saja bukan pengganti transaksi, karena tidak adanya kemampuan membatalkan efek perubahan di tengah jalan. Di seri 3.23-lah MySQL mulai memiliki kemampuan transaksi, row-level locking, dan foreign key constraint. Meski beberapa fasilitas popular lainnya masih belum ada—seperti subselek, view, dan trigger—ini sudah direncanakan dan akan mulai ditambahkan di seri 4.x.

2. Seri 3.23 MySQL menambahkan tiga jenis tabel baru: pertama MyISAM, yang sampai sekarang menjadi tipe tabel default; kedua BerkeleyDB, yang pertama kali menambahkan kemampuan transaksi pada MySQL; dan ketiga InnoDB, primadona baru yang potensial. MySQL memang memiliki arsitektur yang memungkinkan tiap tabel ditangani oleh *handler* yang berbeda. Handler ini menerima perintah akses dan modifikasi dari lapisan MySQL yang lebih atas dan mewujudkannya secara fisik dari dan ke disk. Tabel tipe MyISAM merupakan tabel yang lebih cepat dari tabel ISAM, karena pola aksesnya telah disesuaikan dan dioptimasi untuk pola akses SQL. Selain itu MyISAM mendukung indeks pada kolom bertipe **TEXT** dan **BLOB**, serta mendukung tipe indeks **FULL TEXT**. Tabel tipe BerkeleyDB, atau BDB, menggunakan database embedded BerkeleyDB yang sudah terkenal itu untuk memanfaatkan kemampuan transaksinya. Sejak memiliki handler BDB, MySQL naik statusnya menjadi database yang ACID compliant, sesuatu yang amat penting bagi keamanan data. Namun BerkeleyDB tidaklah terlalu optimal untuk sebuah database SQL, sehingga akhirnya sebuah perusahaan Finlandia Innobase Oy membuat

handler table baru bagi MySQL yang menggunakan database embedded InnoDB-nya—kala itu bernama Innobase, namun berganti nama karena masalah trademark produk Innobase yang sudah ada sebelumnya. InnoDB membuat MySQL menarik karena peningkatan kecepatan dan kemampuan tambahan yang dimungkinkannya. Pertama-tama, InnoDB memiliki fitur transaksi dengan sistem multiversi. Artinya, jika sebuah klien memulai transaksi, maka perubahan yang dilakukan klien tersebut tidak akan terlihat oleh klien lain. Klien lain akan melihat kondisi tabel sebelum transaksi. Barulah jika transaksi oleh klien pertama dicommit, perubahan ini menjadi terlihat di semua klien lain. Bandingkan ini dengan tabel BDB misalnya, di mana klien yang ingin mengakses sebuah tabel yang sedang dipakai dalam sebuah transaksi akan diblok. Dengan kata lain, multiversi menghindari bloking yang tidak perlu. Ini ditambah lagi dengan row-level locking yang disediakan InnoDB, sehingga mempertinggi konkurensi dan mencegah terlalu banyak bloking. Handler tabel MySQL yang lain hanya mendukung table-level locking, sehingga sebuah klien hanya bisa memblok dengan tingkat granulasi tabel dan berpotensi memblok banyak klien lain yang ingin mengakses tabel yang sama. PostgreSQL dan beberapa database komersial sudah memiliki kemampuan-kemampuan ini. Berkat InnoDB, MySQL pun kini memiliki keduanya. Selain itu, InnoDB menambahkan foreign key constraint, sesuatu yang sering didambakan pengguna MySQL karena bisa menyederhanakan logika pemrograman dan menjaga kekonsistenan database mereka.

3. Seri 4.x. Di seri yang baru berjalan hingga 4.0 tahap alfa ini, pengembang MySQL berjanji akan menjadikan MySQL satu derajat lebih tinggi lagi. Fitur-fitur yang sejak dulu diminta akan dikabulkan, seperti subselek (di 4.1), union (4.0), foreign key constraint (4.0 atau 4.1—meski InnoDB sudah menyediakan ini di 3.23.x), stored procedure (4.1), view (4.2), cursor (4.1 atau 4.2), trigger (4.1). MySQL AB tetap berdedikasi mengembangkan dan memperbaiki MySQL, serta mempertahankan MySQL sebagai database open source terpopuler. Di versi 3.23 dan selanjutnya MySQL semakin menarik untuk dilirik oleh pengguna non-Web. Situs mysql.com di halaman depannya dengan bangga memajang profil para pengguna MySQL yang terkenal, seperti Yahoo! Finance (yang mulai mengaku menggunakan MySQL beberapa waktu lalu di milis mysql), NASA (yang membuang Oracle demi memotong budget), atau perusahaan sekuritas Jepang Aizawa. MySQL tidak lagi sekedar dipercaya menyimpan data posting komentar pengunjung situs Web (seperti di **slashdot.org**, pengguna veteran MySQL) atau hit atau log Web, tapi juga kini sebagai tempat menyimpan data keuangan dan transaksi bisnis. Meskipun demikian, karakteristik pengguna database—yang biasanya terdiri dari perusahaan—adalah cenderung konvensional dalam mengadopsi produk baru. Jadi jangan terkejut kalau misalnya sampai sekarang Anda akan masih menjumpai orang yang menganggap MySQL belum ACID-compliant atau belum bisa melakukan transaksi. Padahal sudah dua tahun fitur ini ada di MySQL.

2.4.2.3 Keunggulan MySQL

1. MySQL merupakan program yang multi-threaded, sehingga dapat dipasang pada server yang memiliki multi-CPU.
2. Didukung program-program umum seperti C, C++, Java, Perl, PHP, Python, TCL APIs dls.
3. Bekerja pada berbagai platform. (tersedia berbagai versi untuk berbagai sistem operasi).
4. Memiliki jenis kolom yang cukup banyak sehingga memudahkan konfigurasi sistem database.
5. Memiliki sistem sekuriti yang cukup baik dengan verifikasi host.
6. Mendukung ODBC untuk sistem operasi Microsoft Windows.
7. Mendukung record yang memiliki kolom dengan panjang tetap atau panjang bervariasi.

2.4.2.4 Sistem Server Database MySQL

Sistem database MySQL memiliki sistem sekuritas dengan tiga verifikasi yaitu user(name), password dan host. Verifikasi host memungkinkan untuk membuka sekuriti di ‘localhost’, tetapi tertutup bagi host lain (bekerja di lokal komputer). Sistem sekuriti ini ada di dalam database mysql dan pada tabel user. Proteksi juga dapat dilakukan terhadap database, tabel, hingga kolom secara terpisah.

2.4.3 DSS (DECISION SUPPORT SYSTEM)

DSS merupakan implementasi teori-teori pengambilan keputusan yang telah diperkenalkan oleh ilmu-ilmu seperti operation research dan management science. Hanya bedanya adalah bahwa jika dahulu untuk mencari penyelesaian masalah yang dihadapi harus dilakukan perhitungan iterasi secara manual (biasanya untuk mencari nilai minimum, maksimum, atau optimum), saat ini komputer PC telah menawarkan kemampuannya untuk menyelesaikan persoalan yang sama dalam waktu relatif singkat. Misalnya dengan Jenis DSS menggunakan sebuah Grafik Keputusan.

Dalam merepresentasikan DSS agar mudah dipergunakan dan dimengerti oleh user (dalam hal ini adalah manajer perusahaan), format grafik mutlak dipergunakan untuk melengkapi teks yang ada. Contoh-contoh model grafik yang populer dipergunakan adalah sebagai berikut:

1. Time Series Charts – untuk melihat dampak sebuah variable terhadap waktu
2. Bar Charts – untuk memperbandingkan kinerja beberapa entiti
3. Pie Charts – untuk melihat komposisi atau persentasi suatu hal
4. Scattered Diagrams – untuk menganalisa hubungan antara beberapa variabel
5. Maps – untuk merepresentasikan data secara geografis
6. Layouts – untuk menggambarkan lokasi barang secara fisik, seperti pada bangunan dan kantor
7. Hierarchy Charts – untuk menggambarkan struktur organisasi

8. Sequence Charts – untuk merepresentasikan sesuatu dengan logika yang tersetruktur (contohnya adalah diagram flowchart)
9. Motion Graphics – untuk memperlihat-kan perilaku dari variabel yang diamati denvan cara animasi.

Jenis-jenis grafik di atas biasanya dapat ditampilkan dalam dua macam format dua dimensi dan tiga dimensi.