

BAB II DASAR TEORI

2.1 Food Court

Food Court merupakan suatu tempat yang terdapat pada pusat perbelanjaan atau pusat-pusat keramaian kota, yang menyediakan dan menjual aneka jenis makanan dan minuman. Salah satu food court yang digunakan sebagai acuan untuk pengembangan Sistem Transaksi Pembayaran Food Court adalah Food Court yang terdapat pada pusat perbelanjaan Plaza Araya dan Malang Town Square, dimana masing-masing pusat perbelanjaan memiliki sistem pembayaran yang berbeda. Food Court Plaza Araya menggunakan sistem pembayaran secara terpusat pada satu loket kasir, sedangkan Food Court Malang Town Square menggunakan sistem pembayaran langsung pada masing-masing counter makanan.

2.2 Sistem Transaksi

Sistem Transaksi dapat didefinisikan sebagai kumpulan elemen yang saling berhubungan satu sama lain yang membentuk satu kesatuan untuk memproses atau mengolah suatu transaksi.

Transaksi merupakan suatu peristiwa yang dapat dinilai dengan uang. Adapun bukti transaksi dapat berupa : kuitansi, nota, memo, faktur, dan lain-lain.

2.2.1 Transaksi Keuangan

Transaksi keuangan merupakan kegiatan ekonomi pada suatu unit organisasi atau kejadian yang menyangkut unit organisasi yang harus diukur jumlah rupiahnya dan dicatat dalam sistem akuntansi sehingga jumlah rupiahnya akan mempengaruhi laporan keuangan yang dihasilkan [SUW – 91]. Untuk dapat dikatakan sebagai transaksi keuangan, transaksi tersebut harus dapat dinyatakan dalam satuan uang. Untuk dapat disebut sebagai transaksi, kegiatan tersebut tidak harus berupa kegiatan fisik. Hasil suatu keputusan manajemen yang berpengaruh terhadap laporan keuangan juga termasuk dalam pengertian transaksi.

Untuk suatu unit organisasi berupa perusahaan, Transaksi keuangan dapat digolongkan menjadi transaksi Eksternal dan transaksi Internal. Transaksi

Eksternal adalah transaksi yang terjadi antara perusahaan dengan pihak luar sedangkan Transaksi Internal adalah transaksi yang terjadi dalam unit perusahaan.

2.2.2 Memproses data keuangan

Pemrosesan data menyangkut berbagai tindakan dan kegiatan, antara lain :

- *Mengidentifikasi* berarti menentukan apakah suatu objek dalam transaksi memenuhi kriteria untuk diukur dan dimasukkan dalam laporan keuangan.
- *Mengukur* berarti menentukan jumlah rupiahnya agar jumlah rupiah tersebut dapat diolah lebih lanjut.
- *Mencatat* berarti memasukkan angka hasil pengukuran kedalam sistem akuntansi sehingga angka kos tersebut termuat dalam laboran keuangan.
- *Mengklasifikasi* berarti memasukkan suatu jumlah rupiah kedalam kategori yang telah ditentukan.
- *Mengalokasi* berarti memecah kesatuan kos menjadi beberapa bagian berdasarkan objek tertentu.
- *Meringkas* berarti menggabungkan berbagai jumlah rupiah dari berbagai kategori khusus kedalam kategori yang lebih umum.
- *Menyajikan* berarti memuat jumlah rupiah hasil penggolongan kedalam laporan dengan format dan susunan yang telah ditentukan.

2.3 Jaringan Komputer

Jaringan Komputer yang disebut secara singkat dengan jaringan adalah kumpulan komputer dan alat-alat lain yang saling dihubungkan bersama menggunakan media komunikasi tertentu [WAG - 05]. Informasi yang melintas sepanjang media komunikasi, memungkinkan pengguna jaringan untuk saling bertukar data atau menggunakan perangkat lunak maupun perangkat keras secara berbagi. Masing-masing komputer atau alat-alat lain yang dihubungkan pada jaringan disebut node. Jaringan dapat terdiri dari puluhan, ratusan, atau bahkan ribuan node.

2.3.1 Tipe Jaringan Komputer

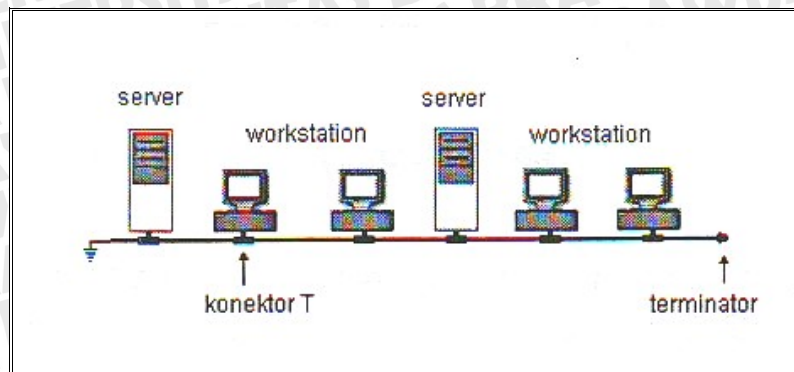
Komputer dalam jaringan dapat saling berhubungan melalui kabel, jaringan telepon. Gelombang radio, satelit, ataupun sinar infra merah. Ada tiga tipe jaringan dalam hubungannya dengan luas area yang dicakup, yaitu :

- **Local Area Network (LAN)**
Local Area Network adalah suatu jaringan yang terbatas pada daerah yang relative kecil, LAN biasanya terbatas pada daerah geografi tertentu, seperti sekolah, kantor, atau gedung. Dalam konfigurasi LAN, satu komputer dirancang sebagai *server*. *Server* menjalankan semua perangkat lunak yang digunakan untuk mengendalikan jaringan, serta menyimpan perangkat lunak yang dapat dipakai bersama oleh komputer-komputer yang dihubungkan pada jaringan.
- **Metropolitan Area Network (MAN)**
Metropolitan Area Network meliputi area geografi yang lebih luas, seperti suatu kota. Dengan interkoneksi jaringan dalam area geografi yang luas, informasi dapat disebarakan secara mudah melalui jaringan.
- **Wide Area Network (WAN)**
Wide Area Network meliputi area geografis yang lebih luas lagi, yang meliputi suatu negara atau dunia. Umumnya jaringan ditempatkan pada banyak lokasi yang berbeda. WAN digunakan untuk menghubungkan banyak LAN yang secara geografis terpisah.

2.3.2 Topologi Jaringan

Topologi merupakan gambaran bagaimana komputer dan peralatan jaringan tersusun dalam suatu jaringan. Setiap topologi yang digunakan untuk membentuk suatu jaringan mempunyai kelebihan dan kekurangan masing-masing, beberapa jenis topologi antara lain:

- **Topologi Bus**
Pada topologi Bus, komputer dan server dihubungkan pada sebuah kabel secara berderet. ujung-ujung kabel data diberi komponen elektronik yang disebut terminator, seperti digambarkan dalam Gambar 2.1.



Gambar 2.1 Topologi Bus

Sumber : *Wagito*, 2005 :17

Keuntungan dari penggunaan topologi bus, antara lain :

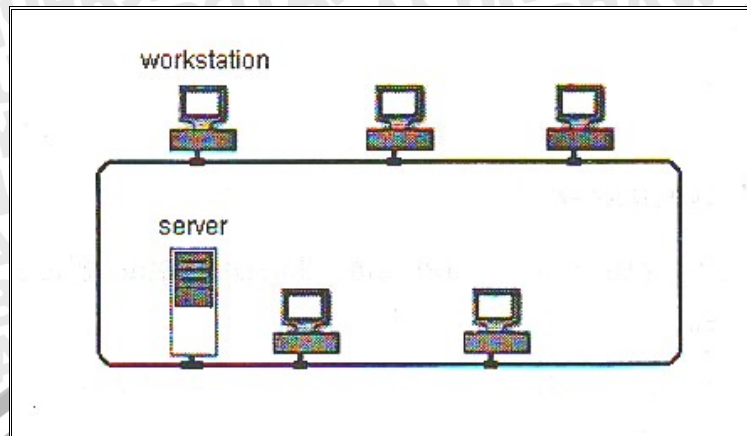
- Sangat mudah menghubungkan komputer serta peralatan lainnya pada kabel bus linear;
- Hemat dalam penggunaan kabel.

Sedangkan kekurangan dari penggunaan topologi bus, antara lain :

- Jika salah satu sambungan terputus, maka seluruh jaringan tidak dapat bekerja;
- Dibutuhkan terminator pada dua ujung kabel utama;
- Sulit untuk mengidentifikasi kesalahan jika seluruh jaringan tidak bekerja.

- Topologi Ring

Pada topologi ring, komputer dan peralatan dihubungkan pada suatu kabel utama, sehingga kabel utama membentuk suatu ring. Ujung-ujung dari kabel utama saling dihubungkan, sehingga membentuk lingkaran tak terputus, seperti digambarkan dalam Gambar 2.2.



Gambar 2.2 Topologi Ring

Sumber : Wagito, 2005 :18

Keuntungan dari penggunaan topologi ring, antara lain :

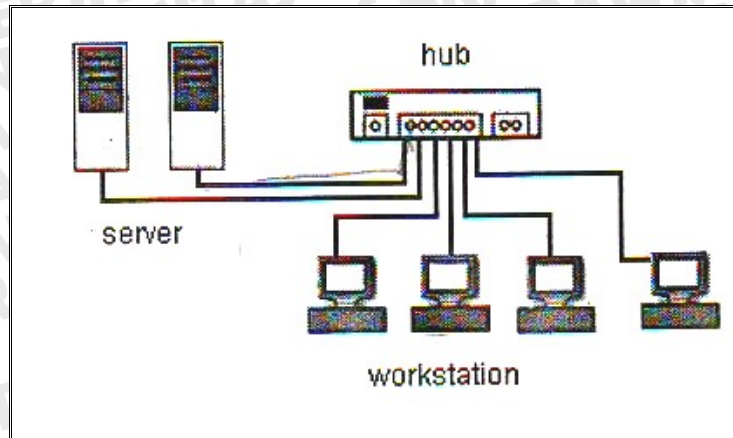
- Jika salah satu peralatan rusak, maka tidak akan mengganggu kinerja jaringan karena tiap peralatan tidak dihubungkan secara langsung tetapi melewati bus.

Sedangkan kerugian dari penggunaan topologi ring, antara lain :

- Jika kabel rusak, maka semua peralatan tidak dapat berfungsi;
- Sulit dalam pengelolaan.

- Topologi Star

Pada topologi star, setiap komputer dihubungkan secara langsung pada suatu pusat jaringan yang berupa hub seperti yang diperlihatkan pada Gambar 2.3. Hub bertindak sebagai pengelola dan pengendali semua fungsi dalam jaringan. Hub juga berfungsi sebagai *repeater* aliran data.



Gambar 2.3 Topologi Star

Sumber : Wagito, 2005:19

Keuntungan dari penggunaan topologi star, antara lain :

- Bandwith jalur komunikasi dalam kabel akan semakin lebar sehingga akan meningkatkan unjuk kerja jaringan secara keseluruhan;
- Bila terdapat gangguan di suatu jalur kabel maka gangguan hanya akan terjadi dalam komunikasi antara *workstation* yang bersangkutan dengan *server*.

Sedangkan kerugian dari penggunaan topologi star, antara lain :

- Membutuhkan kabel yang cukup panjang, sehingga investasi cukup mahal;
- Jika Hub rusak, maka fungsi jaringan tidak dapat bekerja.

2.2.3 Manfaat Jaringan Komputer

- Jaringan komputer memungkinkan penggunaan *hardware/software* yang terhubung dengan jaringan secara bersama-sama.
- Jaringan komputer dapat digunakan untuk menyediakan penyimpanan data alternatif sehingga pengguna komputer akan mendapat reliabilitas yang tinggi.
- Jaringan komputer dapat menghemat pengeluaran pada suatu instansi/perusahaan, karena penggunaan sumber daya secara bersama-sama.

2.4 Data Flow Diagram (DFD)

DFD adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut.

DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir atau dimana data tersebut akan disimpan.

DFD merupakan alat yang digunakan pada metodologi pengembangan sistem yang terstruktur. Kelebihan utama pendekatan aliran data, yaitu :

- Kebebasan dari menjalankan implementasi teknis sistem.
- Pemahaman lebih jauh mengenai keterkaitan satu sama lain dalam sistem dan subsistem.
- Mengkomunikasikan pengetahuan


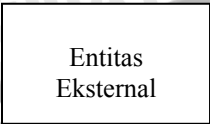
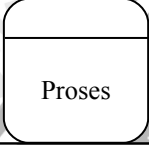
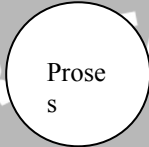
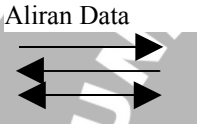

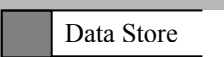

DFD terdiri dari *context diagram* dan diagram rinci (*DFD Levelled*).

- *Context diagram* berfungsi memetakan model lingkungan (menggambarkan hubungan antara entitas luar, masukan dan keluaran sistem), yang direpresentasikan dengan lingkaran tunggal yang mewakili keseluruhan sistem.
- *DFD levelled* menggambarkan sistem sebagai jaringan kerja antara fungsi yang berhubungan satu sama lain dengan aliran dan penyimpanan data, model ini hanya memodelkan sistem dari sudut pandang fungsi.

2.4.1 Simbol dalam DFD

Terdapat 2 versi dari simbol-simbol yang digunakan dalam DFD seperti yang ditunjukkan pada Tabel 2.1.

Tabel 2.1 Simbol Diagram Aliran Data

Gane / Sarson	Yourdon / De Marco	Keterangan
 Entitas Eksternal	 Entitas Eksternal	Entitas eksternal dapat berupa orang/unit terkait yang berinteraksi dengan sistem tetapi diluar sistem.
 Proses	 Proses	Orang, unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi.
 Aliran Data	 Aliran Data	Aliran data dengan arah khusus dari sumber ke tujuan.
 Data Store	 Data Store	Penyimpanan data atau tempat data yang diserahkan oleh proses.

Sumber: Anonymous, 2004 : 2

2.5 Basis Data

Basis data merupakan tempat penyimpanan kumpulan data secara elektronik. Data tersebut dapat disimpan menggunakan media penyimpanan seperti *flashdisk*, *harddisk*, atau *compact disk* (CD). Basis data sendiri dapat didefinisikan sebagai : [IRM – 03]

- Kumpulan dari item yang saling berhubungan satu sama lain yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu.
- Kumpulan data yang saling berhubungan yang disimpan secara bersama-sama tanpa adanya duplikasi data, untuk memenuhi berbagai kebutuhan.
- kumpulan file, tabel dan arsip yang saling berhubungan yang disimpan dalam media penyimpanan elektronik.

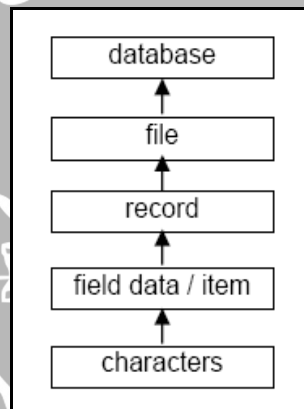
Database diperlukan karena:

- Salah satu komponen penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi.

- Menentukan kualitas informasi : akurat, tepat pada waktunya dan relevan. Informasi dapat dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.
- Mengurangi duplikasi data (*data redundancy*).
- Hubungan data dapat ditingkatkan (*data relatability*).

2.5.1 Komponen Basis Data

Basis data terdiri atas beberapa komponen untuk membentuk struktur basis data yang dibutuhkan. Tingkatan dari komponen basis data diperlihatkan dalam Gambar 2.4 :

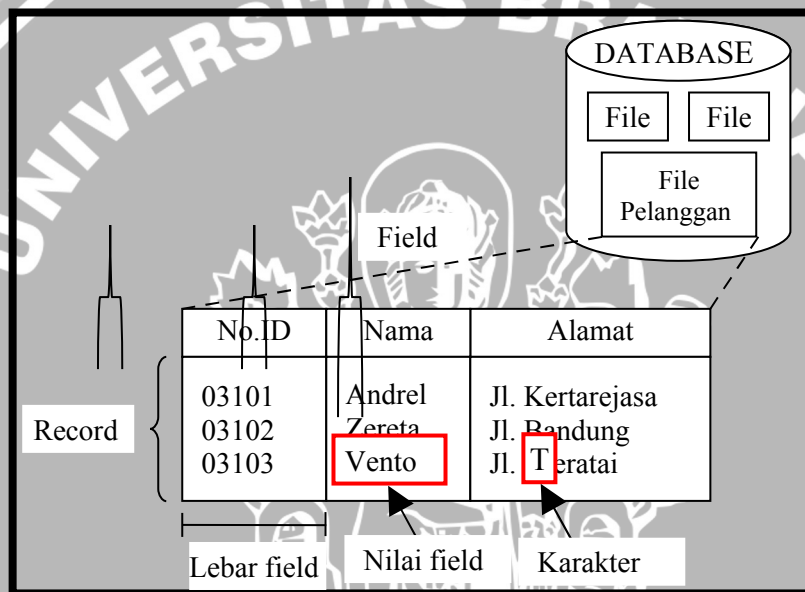


Gambar 2.4. Tingkatan Komponen Basis Data

Sumber: Faried Irmansyah, 2003:2

- *Characters* : merupakan bagian data yang terkecil, dapat berupa karakter numerik, huruf ataupun karakter-karakter khusus (special characters) yang membentuk suatu item data / field.
- *Field* : merepresentasikan suatu atribut dari record yang menunjukkan suatu item dari data, seperti misalnya nama, alamat dan lain sebagainya. Kumpulan dari field membentuk suatu record.
 - *field name*: harus diberi nama untuk membedakan field yang satu dengan lainnya
 - *field representation*: tipe field (karakter, teks, tanggal, angka, dsb), lebar field (ruang maksimum yang dapat diisi dengan karakter-karakter data).
 - *field value*: isi dari field untuk masing-masing record.

- *Record* : Kumpulan dari field membentuk suatu record. Record menggambarkan suatu unit data individu yang tertentu. Kumpulan dari record membentuk suatu file. Misalnya file personalia, tiap-tiap record dapat mewakili data tiap-tiap karyawan.
- *File*: File terdiri dari record-record yang menggambarkan satu kesatuan data yang sejenis. Misalnya file mata pelajaran berisi data tentang semua mata pelajaran yang ada.
- *Database* : Kumpulan dari file / tabel membentuk suatu database



Gambar 2.5. Komponen Basis Data.

Sumber : Faried Irmansyah, 2003:3

2.5.2 Normalisasi Basis Data

Normalisasi merupakan sebuah teknik dalam logika desain sebuah basis data, teknik pengelompokan atribut dari suatu relasi sehingga membentuk struktur relasi yang baik.

- **Normal Pertama (1st Normal Form)**

Aturan :

- Mendefinisikan atribut kunci.
- Tidak adanya group berulang.
- Semua atribut bukan kunci tergantung pada atribut kunci.

- **Normalisasi Kedua (2nd Normal Form)**

Aturan :

- Sudah memenuhi dalam bentuk normal kesatu.
- Sudah tidak ada ketergantungan parsial, dimana seluruh field hanya tergantung pada sebagian field kunci.

- **Normalisasi Ketiga (3rd Normal Form)**

Aturan :

- Sudah berada dalam bentuk normal kedua.
- Tidak ada ketergantungan transitif (dimana field bukan kunci tergantung pada field bukan kunci lainnya).

2.5.3 Diagram Entity-Relationship (ER-Diagram)

ERD merupakan notasi grafis dalam pemodelan data konseptual yang mendeskripsikan hubungan antara penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data, karena hal ini relatif kompleks. ERD menggunakan sejumlah notasi dan simbol untuk menggambarkan struktur dan hubungan antar data, pada dasarnya ada 3 macam simbol yang digunakan yaitu : [IRM – 03]

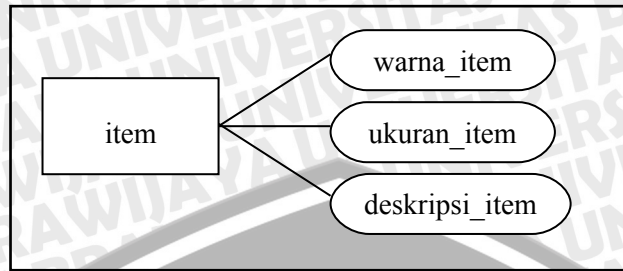
1. Entiti : adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat. Sebuah entiti digambarkan dalam bentuk segi empat



Pelanggan

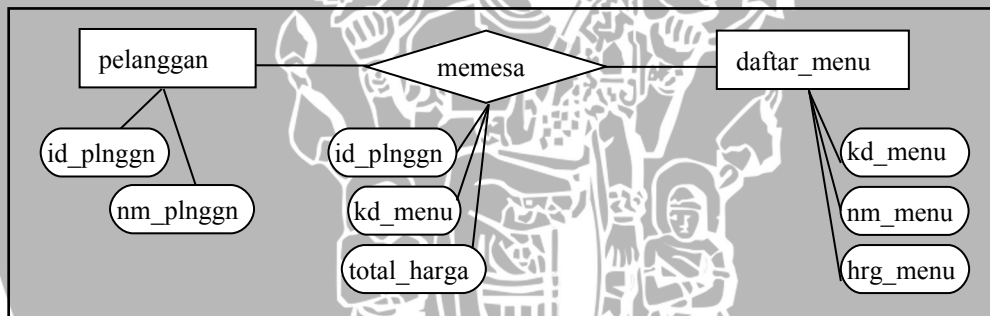
Gambar 2.6. Notasi Entiti pada ER-Diagram

2. Atribut : Entiti mempunyai elemen yang disebut atribut, dan berfungsi mendeskripsikan karakter entiti. Misalnya atribut nama pelanggan dari entiti pelanggan. Setiap ERD bisa terdapat lebih dari satu atribut. Entiti digambarkan dalam bentuk ellips.



Gambar 2.7. Notasi Atribut pada ER-Diagram

3. Hubungan : Relationship, sebagaimana halnya entiti maka dalam hubungan pun harus dibedakan antara hubungan atau bentuk hubungan antar entiti dengan isi dari hubungan itu sendiri. Misalnya dalam kasus hubungan antara entiti pelanggan dan entiti daftar_menu adalah memesan, sedangkan isi hubungannya dapat berupa total_harga. Relationship digambarkan dalam bentuk intan/diamond.



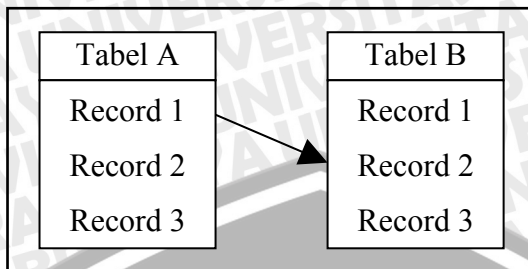
Gambar 2.8. Notasi Relationship pada ER-Diagram

Relationship atau hubungan mempunyai beberapa derajat relasi untuk menunjukkan jumlah maksimum entitas yang dapat berrelasi dengan entitas pada himpunan entitas yang lain. Derajat relasi yang terjadi diantara 2 himpunan entitas (misalkan A dan B) yaitu :

- Bentuk Relasi One to One

Bentuk relasi dikatakan One to One apabila tabel A memiliki satu anggota yang juga terdapat pada tabel B. Sebagai gambaran lihat hubungan kedua tabel berikut :



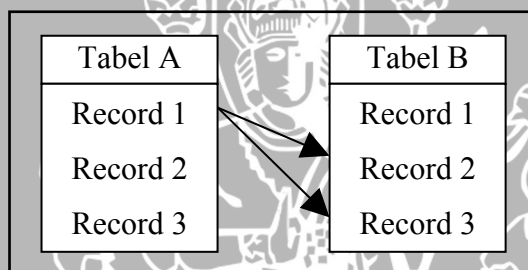


Gambar 2.9. Bentuk Relasi One to One

Sumber : *Bunafit Nugroho, 2005 : 95*

- Bentuk Relasi One to Many

Bentuk relasi dikatakan One to Many apabila salah satu anggota tabel A memiliki anggota pada tabel B lebih dari satu. Sebagai gambaran lihat hubungan kedua tabel berikut :

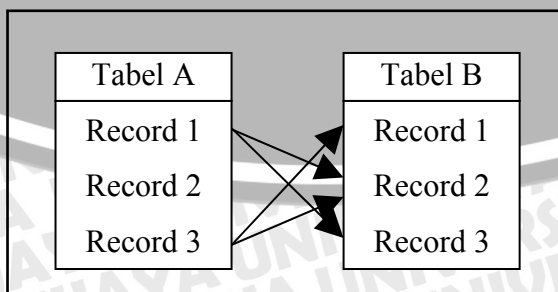


Gambar 2.10. Relasi One To Many

Sumber : *Bunafit Nugroho, 2005 : 96*

- Bentuk Relasi Many to Many

Tabel dikatakan memiliki hubungan Many to Many apabila semua baris data pada tabel A memiliki banyak anggota yang berada pada tabel B. Hubungan tersebut dapat digambarkan sebagai berikut :



Gambar 2.11. Relasi Many to Many

Sumber : *Bunafit Nugroho, 2005 : 97*

2.5.4 Relasi (Relational Database Management System / RDBMS)

Merupakan sekumpulan data yang disimpan sedemikian rupa sehingga mudah diambil informasinya bagi pengguna, dan data tersebut saling berhubungan. RDBMS merupakan suatu paket perangkat lunak yang kompleks digunakan untuk memanipulasi database.

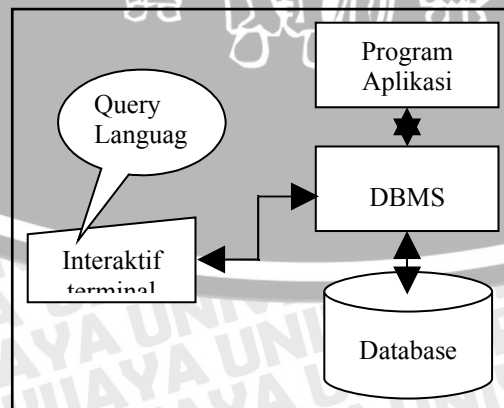
Ada tiga prinsip dalam RDBMS :

- *Data definition*, Mendefinisikan jenis data yang akan dibuat (dapat berupa angka atau huruf), cara relasi data, validasi data dan lainnya.
- *Data Manipulation*, Data yang telah dibuat dan didefinisikan tersebut akan dilakukan beberapa pengerjaan, seperti menyaring data, melakukan proses *query*, dsb.
- *Data Control*, Bagian ini berkenaan dengan cara mengendalikan data, seperti siapa saja yang bisa melihat isi data, bagaimana data bisa digunakan oleh banyak user, dsb

Semua operasi input dan output yang berhubungan dengan database harus menggunakan DBMS. Bila pemakai akan mengakses database, DBMS menyediakan penghubung (interface) antara pemakai dengan database.

Hubungan pemakai dengan database dapat dilakukan dengan dua cara :

1. Secara interaktif menggunakan bahasa pertanyaan (*query language*).
2. Dengan menggunakan program aplikasi.



Gambar 2.12. Hubungan Pemakai dengan Database

Sumber : Faried Irmansyah, 2003 : 10



2.6 SQL (Structured Query Language)

SQL adalah sebuah bahasa permintaan *database* yang terstruktur, bahasa SQL dibuat sebagai bahasa yang dapat merelasikan beberapa tabel dalam *database* manapun merelasikan antar *database*. Bahasa SQL ditulis langsung dalam sebuah program *database* sehingga seorang pengguna dapat melihat langsung permintaan yang diinginkan, sekaligus melihat hasilnya. MySQL sendiri adalah sebuah program *database server* yang memerlukan sebuah bahasa permintaan dalam melayani permintaan *user* [NUG - 04].

SQL dibagi menjadi dua bentuk *Query*, yaitu :

- DDL (*Data Definition Language*) adalah metode Query SQL yang berguna untuk mendefinisikan data pada sebuah *database*, adapun query yang dimiliki adalah :
 - CREATE : Digunakan untuk melakukan pembuatan tabel dan *database*.
 - DROP : Digunakan untuk melakukan penghapusan tabel maupun *database*.
 - ALTER : Digunakan untuk melakukan perubahan struktur tabel yang telah dibuat, baik menambah Field (add), mengganti nama Field (change), ataupun menamakannya kembali (rename), serta menghapus (drop).
- DML (*Data Manipulation Language*) adalah sebuah metode Query yang dapat digunakan apabila DDL telah terjadi, sehingga fungsi dari Query ini adalah untuk melakukan pemanipulasian *database* yang telah ada atau telah dibuat sebelumnya. Adapun query yang termasuk didalamnya adalah:
 - INSERT : Digunakan untuk melakukan penginputan / pemasukan data pada tabel *database*.
 - UPDATE : Digunakan untuk melakukan perubahan atau peremajaan terhadap data yang ada pada tabel.
 - DELETE : Digunakan untuk melakukan penghapusan data pada tabel. Penghapusan ini dapat dilakukan secara sekaligus (seluruh isi tabel) maupun hanya beberapa Recordset.

2.7 Database MySQL

MySQL adalah sebuah program *database* yang berbentuk RDBMS (*Relational Database Management System*) yang mampu menerima dan mengirimkan datanya dengan sangat cepat, *multi user* serta menggunakan perintah standar SQL (*Structured Query Language*), MySQL yang kita gunakan adalah MySQL *FreeSoftware* yang berada dibawah lisensi GNU/GPL (*General Public License*) merupakan sebuah *database server* yang *free*, artinya kita bebas menggunakan *database* ini untuk keperluan pribadi atau usaha tanpa harus membeli atau membayar lisensinya. [NUG - 04]

Selain itu *database* ini memiliki beberapa kelebihan dibanding *database* lain, diantaranya adalah :

- MySQL sebagai Database Management System (DBMS).
- MySQL sebagai Relational Database Management System (RDBMS).
- MySQL adalah sebuah *software database* yang *OpenSource*, artinya program ini bersifat *free* digunakan oleh siapa saja tanpa harus membeli dan membayar lisensi kepada pembuatnya.
- MySQL merupakan sebuah *database server*, jadi dengan menggunakan *database* ini dapat menghubungkannya ke media *Internet* sehingga dapat diakses dari jauh.
- MySQL merupakan sebuah *database client*. Selain menjadi *server* yang melayani permintaan, MySQL juga dapat melakukan *query* yang mengakses *database* pada *server*. Jadi MySQL dapat juga berperan sebagai *Client*.
- MySQL mampu menerima *query* yang bertumpuk dalam satu permintaan atau yang disebut *Multi-Threading*.
- MySQL merupakan *database* yang mampu menyimpan data berkapasitas sangat besar hingga berukuran *Gigabyte* sekalipun.
- MySQL didukung oleh *driver ODBC*, artinya *database* MySQL dapat diakses menggunakan aplikasi apa saja termasuk berupa *visual* seperti *Delphi* maupun *Visual Basic*.
- MySQL adalah *database* menggunakan *enkripsi password*. Jadi *database* ini cukup aman karena memiliki *password* untuk mengaksesnya.

- MySQL merupakan *Server database* yang *multi user*, artinya *database* ini tidak hanya digunakan oleh sepihak orang akan tetapi merupakan *database* yang dapat digunakan oleh banyak pengguna.
- MySQL dapat menciptakan lebih dari 16 kunci pertabel, dan dalam satu kunci memungkinkan berisi belasan *Field* (kolom).
- MySQL memiliki kecepatan dalam pembuatan tabel maupun peng-update-an tabel.
- MySQL menggunakan bahasa permintaan standar yang bernama SQL (*Structured Query Language*) yaitu sebuah bahasa permintaan yang distandarkan pada beberapa *database server* seperti *Oracle*, *PostgreSQL*, dll.

2.8 Visual Basic

Visual Basic adalah salah suatu pemrograman berbasis visual yang dikembangkan oleh Microsoft sejak tahun 1991. Visual Basic merupakan pengembangan dari bahasa pemrograman BASIC (*Beginner's All-Purpose Symbolic Instruction Code*) yang dikembangkan pada tahun 1950-an.

Visual Basic menggunakan model *Event-Driven* dalam menjalankan perintah-perintah yang diberikan. Pada pemrograman prosedural seperti C++ atau Pascal, kode-kode yang dibangun dijalankan dengan alur yang teratur. Dimana kode-kode dieksekusi mulai baris pertama dan mengikuti langkah yang telah ditentukan pada aplikasi. Kode-kode dalam Visual Basic tidak mengikuti alur tersebut, akan tetapi kode-kode dieksekusi pada bagian lain sesuai respon terhadap event yang diberikan, dimana event dijalankan apabila mendapatkan respon dari user yang menggunakan aplikasi tersebut.

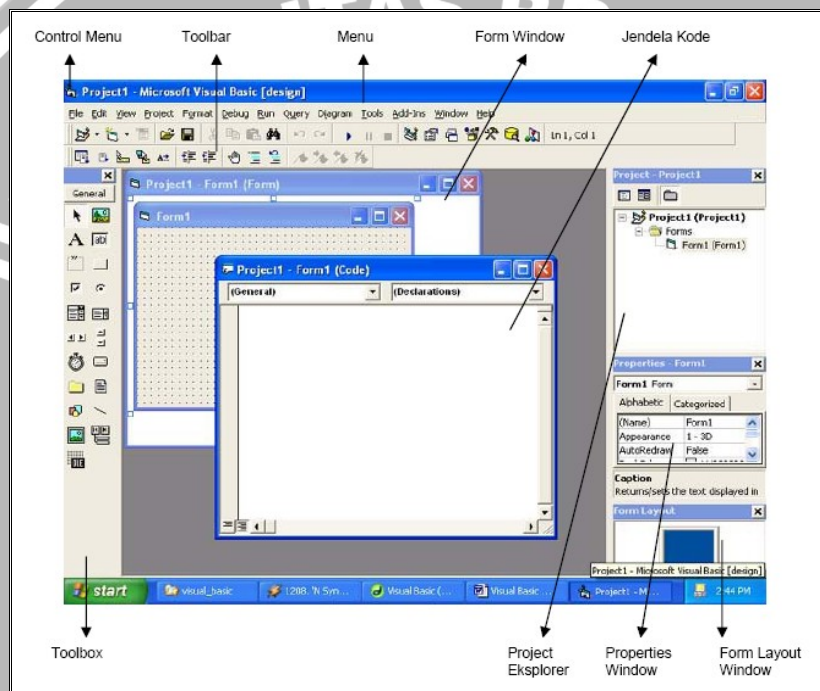
Beberapa keistimewaan utama dari Visual Basic 6 diantaranya seperti :

- Menggunakan *platform* pembuatan programan yang diberi nama *Developer Studio*, yang memiliki tampilan dan sarana yang sama dengan Visual C++ dan Visual J++. sehingga dapat berimigrasi atau belajar cepat bahasa pemrograman lainnya dengan mudah dan cepat tanpa harus belajar dari nol lagi.

- Memiliki beberapa tambahan sarana *wizard* yang baru. *Wizard* adalah sarana yang mempermudah dalam pembuatan aplikasi dengan mengotomatisasi tugas-tugas tertentu.
- Akses data lebih cepat dan handal untuk membuat aplikasi *database* berkemampuan tinggi.

2.8.1 IDE (*Integrated Development Environment*) Visual Basic

Di dalam lingkungan Visual Basic, terdapat berbagai macam komponen ditunjukkan dalam Gambar 2.13 yang terdiri dari :



Gambar 2.13. IDE (*Integrated Development Environment*) Visual Basic

Sumber : IPB, 2001 : 27

1. *Control Menu*

Control Menu adalah menu yang digunakan terutama untuk memanipulasi jendela Visual Basic. Dari menu ini anda dapat mengubah ukuran, memindahkannya, atau menutup jendela.

2. *Menu*

Menu Visual Basic berisi semua perintah Visual Basic yang dapat dipilih untuk melakukan tugas tertentu. Isi dari menu ini sebagian hampir sama dengan program-program Windows pada umumnya.

3. *Toolbar*

Toolbar adalah tombol-tombol (*shortcut*) yang mewakili suatu perintah tertentu dari Visual Basic.

4. *Form Window*

Form Window atau jendela form adalah daerah kerja utama tempat membuat program-program aplikasi Visual Basic.

5. *Toolbox*

Toolbox adalah sebuah “kotak piranti” yang mengandung semua objek atau ‘kontrol’ yang dibutuhkan untuk membentuk suatu program aplikasi. Kontrol adalah suatu objek yang akan menjadi penghubung antara program aplikasi dan user-nya, dan yang kesemuanya harus diletakkan di dalam jendela form.

6. *Project Explorer*

Jendela *Project Explorer* adalah jendela yang mengandung semua file di dalam aplikasi Visual Basic. Setiap aplikasi dalam Visual Basic disebut dengan istilah *project* (proyek), dan setiap proyek bisa mengandung lebih dari satu file. Pada *Project Explorer* ditampilkan semua file yang terdapat pada aplikasi (proyek), misalnya form, modul, class, dan sebagainya.

7. *Jendela Properties*

Jendela Properties adalah jendela yang mengandung semua informasi mengenai objek yang terdapat pada aplikasi Visual Basic. Properti adalah sifat dari sebuah objek, misalnya seperti nama, warna, ukuran, posisi, dan sebagainya.

8. *Form Layout Window*

Form Layout Window adalah jendela yang menggambarkan posisi dari form yang ditampilkan pada layar monitor. Posisi *form* pada Form Layout Window inilah yang merupakan petunjuk tempat aplikasi akan ditampilkan pada layar monitor saat dijalankan.

9. *Jendela Code*

Jendela *Code* adalah salah satu jendela yang penting di dalam Visual Basic. Jendela ini berisi kode-kode program yang merupakan instruksi-instruksi untuk aplikasi Visual Basic yang dibuat.

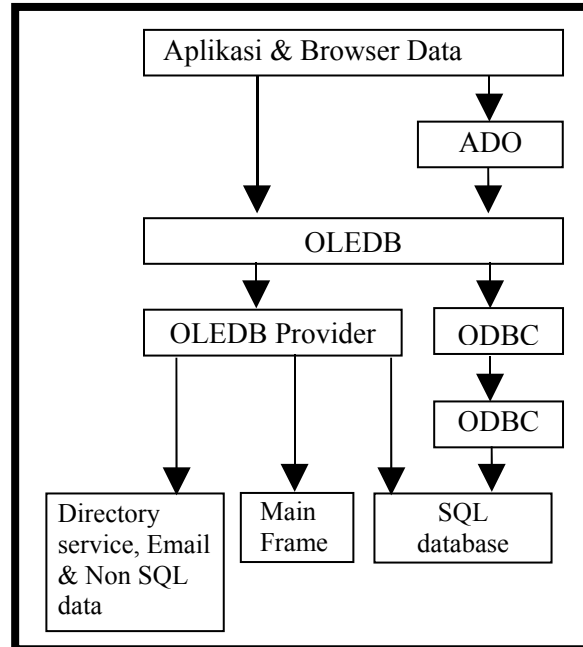
2.8.2 Koneksi Visual Basic dengan Database MySQL

Visual Basic memiliki 3 antarmuka untuk akses data, yaitu : *ActiveX Data Objects* (ADO), *Remote Data Objects* (RDO), dan *Data Access Objects* (DAO). Antarmuka akses data merupakan model objek yang menggambarkan berbagai macam sisi dari pengaksesan data. DAO didesain untuk mengakses database desktop, sedangkan RDO didesain untuk mengakses database client server. DAO dan RDO merupakan versi lama yang digunakan untuk mengakses database pada Visual Basic, sedangkan kontrol data baru dalam Visual Basic 6.0 menggunakan ADO sebagai antarmuka akses data.

ADO memiliki arsitektur terbuka berbasis COM (*Common Object Model*) dan DCOM (*Distributed COM*) sehingga memiliki beberapa keuntungan, antara lain : [SUP – 06]

1. Dapat bekerja pada banyak proses dan juga pada computer terpisah
2. Membuat OLEDB provider sendiri (dengan komponen MDAC (*Microsoft Data Access Component*))

Gambar arsitektur koneksi ADO ditunjukkan dalam gambar 2.14 :



Gambar 2.14. Arsitektur Koneksi Data ADO

Sumber : Yuniar Supardi, 2006 : 80

ADO memiliki 3 bagian utama objek, antara lain :

1. *Connection* adalah objek untuk koneksi ke sumber data, yang mencakup path, direktori, file database, OLEDB Provider, DSN (*Data Source Name*).
2. *Command* adalah objek untuk mengeksekusi / menjalankan perintah yang telah didefinisikan berupa perintah SQL.
3. *Recordset* adalah objek untuk mengatur hasil, salah satu properti pentingnya adalah *CursorType*, yang digunakan untuk menentukan posisi data yang dihasilkan oleh objek recordset.

2.9 Barcode

Barcode merupakan cara lain dalam peng-enkodean angka dan huruf dengan menggunakan kombinasi bar dan spasi yang bermacam-macam. Selain itu *barcode* merupakan salah satu cara untuk proses pengentrian data ke dalam komputer. *Barcode* tidak terdiri atas data-data deskriptif melainkan berupa nomor-nomor referensi yang digunakan oleh komputer untuk mengetahui record yang berisi data deskriptif atau informasi penting lainnya.

2.9.1 Jenis-jenis barcode

- **Kode 39 / 3of 9,**

Dapat mengkodekan karakter *alphanumeric* yaitu angka desimal atau huruf besar serta tambahan karakter spesial-.*\$/%+. Satu karakter dalam Code 39 terdiri dari 9 elemen yaitu 5 bar (garis vertikal hitam) dan 4 spasi (garis vertikal putih) yang disusun bergantian antara bar dan spasi. 3 dari 9 elemen tersebut memiliki ketebalan lebih tebal dari yang lainnya oleh karenanya kode ini biasa disebut juga *code 3 of 9* , 3 elemen yang lebih tebal tersebut terdiri dari 2 bar dan 1 spasi. Elemen yang lebar mewakili digit biner 1 dan elemen yang sempit mewakili digit biner 0.



Gambar 2.15. Barcode Code 39

Sumber : *Anonymous* , 2006 : 2

- **Extended kode 39,**

Merupakan pengembangan dari kode 39 yang dapat mengkodekan seluruh karakter ASCII dengan cara mengkodekannya dengan pasangan kode 39.

- **Interleaved 2 of 5 (ITF),**

ITF *barcode* hanya dapat mengkodekan angka saja dan sering digunakan pada produk-produk yang memiliki kemasan dengan permukaan yang tidak rata (misalnya *corrugated box*), hal ini disebabkan struktur dan cara pengkodean ITF yang unik.

Setiap karakter pada ITF *barcode* dikodekan dengan 5 elemen yaitu 2 elemen tebal dan 3 elemen sempit, dimana elemen tebal mewakili digit biner 1 sedangkan elemen tipis mewakili digit biner 0 dengan perbandingan ketebalan antara elemen tebal dengan elemen tipis 2:1 s/d 3:1.

Keunikan dari ITF adalah pengkodean karakternya apakah menggunakan bar atau menggunakan spasi tergantung pada posisi sesuai dengan namanya *interleaved*, atau lebih jelasnya sebagai berikut :

Karakter pertama dikodekan menggunakan bar setelah *start character*, sedangkan karakter kedua dikodekan menggunakan spasi secara *interleaved* pada karakter pertama, karena sifat berpasang-pasangan itulah panjang message termasuk *check character* haruslah genap jika jumlahnya ganjil maka harus ditambahkan karakter 0 pada awal message.

Untuk menghindari terjadi *partial scan* atau pembacaan *barcode* yang terpotong yang dapat menyebabkan kesalahan informasi, maka seringkali ITF *barcode* diberi tambahan *bearer bar* yaitu bar sepanjang barcode yang ditempatkan di atas dan dibawah barcode seperti tampak dibawah ini.

3 06 21200 66285 3



Gambar 2.16 Barcode ITF
Sumber : *Anonymous*, 2006 : 3

- **Kode 128,**

Adalah *barcode* dengan kerapatan tinggi, dapat mengkodekan keseluruhan simbol ASCII (128 karakter) dalam luasan yang paling minim dibandingkan dengan *barcode* jenis lain.

Hal ini disebabkan karena Kode 128 menggunakan 4 ketebalan elemen (bar atau spasi) yang berbeda (jenis yang lain kebanyakan menggunakan 2 ketebalan elemen yang berbeda). Setiap karakter pada Kode 128 dikodekan oleh 3 bar dan 3 spasi (atau 6 elemen) dengan ketebalan masing-masing elemen 1 sampai 4 kali ketebalan minimum (*module*), jika dihitung dengan satuan *module* maka tiap karakter Kode 128 terdiri dari 11 *module* kecuali untuk *stop character* yang terdiri dari 4 bar 3 spasi (13 *module*). Jumlah total *module* untuk bar selalu genap sedangkan untuk spasi selalu ganjil, selain itu Kode 128 memiliki 3 *start character* yang berbeda sehingga Kode 128 memiliki 3 sub set karakter yang bersesuaian dengan *start character*nya.

Kode 128 memiliki fitur untuk dapat bergeser dari sub set yang satu ke sub set yang lainnya dengan menggunakan karakter CODE dan SHIFT, CODE X menyebabkan seluruh message bergeser menjadi sub set X (misalnya CODE A pada sub set B membuat message menjadi sub set A), sedangkan SHIFT menyebabkan satu karakter didepannya bergeser sub set (ini hanya berlaku untuk sub set A ke sub set B atau sebaliknya). Struktur Kode 128 barcode seperti terlihat dibawah ini :



Gambar 2.17 Barcode Kode 128

Sumber : *Anonymous*, 2006 : 4

2.9.2 Barcode Reader

Komputer tidak dapat membaca *barcode* secara langsung. Untuk mengetahui informasi yang terdapat pada *barcode*, *barcode* harus didekodekan kedalam format data yang dapat diproses oleh komputer. Alat yang dapat

membaca informasi yang terdapat pada *barcode* disebut *barcode reader*, atau secara umum disebut *barcode scanner*.

Barcode reader terdiri atas :

- *Scanner* berfungsi untuk men-*scan* symbol *barcode* dan meng-*capture* atau menangkap bar dan spasi dari *barcode* dan mengirimnya ke *decoder*.
- *Decoder* berfungsi sebagai penerjemah bar dan spasi kedalam bentuk karakter ASCII dan mentransmisikan data ke komputer dalam format data sederhana.
- Kabel berfungsi sebagai antarmuka antara dekoder dan komputer.

2.9.3 Tipe-tipe Barcode Reader

Saat ini terdapat 4 tipe *barcode reader* berbeda yang tersedia diantaranya : *Barcode reader* berbentuk pena atau tongkat, *Laser barcode scanner*, *CCD (Charge Couple Devices) barcode reader* and Kamera berbasis *barcode reader*. Masing-masing tipe ini digunakan dengan teknologi yang berbeda dalam pembacaan dan penguraian *barcode*.

- ***Barcode reader* berbentuk pena atau tongkat**

Merupakan *barcode reader* yang paling sederhana dan tidak begitu mahal. Paling tahan lama dan tidak terdiri atas bagian-bagian yang bergerak. Bagaimanapun, ketika men-*scan*, tongkat atau pena harus bersentuhan dengan *barcode*, jika *barcode* terbaca lebih dari satu kali, memungkinkan terjadinya kerusakan, bahkan tidak dapat dibaca. Selain itu, tongkat atau pena bertenaga manusia, maksudnya harus dipegang pada sudut yang tepat dan digerakkan pada kecepatan yang tepat. Untuk alasan ini, *barcode scanner* berbentuk pena atau tongkat merupakan pilihan tepat ketika biaya menjadi faktor yang menentukan.



Gambar 2.18. Barcode reader berbentuk pena atau tongkat

Sumber : www.semicronsistemas.com

Barcode reader berbentuk pena memiliki sumber cahaya dan foto dioda yang diletakkan pada ujung pena atau tongkat. Untuk membaca *barcode*, dengan men-drag ujung dari pena melewati keseluruhan kode-kode bar dengan gerakan sejajar *barcode*. Foto dioda mengukur intensitas cahaya yang direfleksikan kembali yang berasal dari sumber cahaya dan membangkitkan gelombang yang menghubungkan lebar dari bar dan spasi pada *barcode*. *Barcode reader* mengirim *waveform* ke dekoder, yang akan menguraikan *waveform* dan mengirimnya ke komputer kedalam format data sederhana.

- ***Laser barcode scanners***

Merupakan *barcode scanner* paling populer. *Laser barcode scanner* tidak perlu menyentuh atau mendekati *barcode* untuk pembacaan *barcode*. Rentang standar *barcode scanner laser* dapat membaca *barcode* sekitar 6 hingga 24 inchi, dan rentang yang panjang sekitar 2 hingga 8 feet. Rentang ekstra panjang mencapai 30 feet.



Gambar 2.19 Laser Barcode Scanner

Sumber : www.semicronsyste.ms.com

Cara kerja *Laser barcode scanner* hampir sama dengan *barcode reader* berbentuk pena. Hanya saja perbedaannya terletak pada sorotan laser sebagai sumber cahaya, dan menggunakan *reciprocating mirror* atau perputaran prisma untuk men-scan sorotan laser kembali dan selanjutnya melewati *barcode*. Seperti halnya *barcode reader* berbentuk pena, foto dioda digunakan untuk mengukur intensitas cahaya yang direfleksikan kembali dari *barcode*.

- **CCD (*Charge Couple Devices*) barcode scanner**

CCD *barcode scanner* menggunakan array cahaya sensor yang terdapat pada bagian depan barcode reader. Voltase koresponden *waveform* terhadap bar dan spasi pada *barcode* dibangkitkan dan dikirim ke dekoder, yang menguraikan data dan mengirimnya ke komputer. Perbedaan utama antara CCD *barcode scanner*, *barcode scanner* berbentuk pena, dan *Laser barcode scanner* adalah CCD *barcode scanner* mengukur pancaran cahaya dari *barcode* sedangkan *barcode scanner* berbentuk pena atau laser mengukur cahaya yang direfleksikan yang berasal dari *scanner* itu sendiri.



Gambar 2.20. CCD barcode scanner

Sumber : www.semicronsystems.com

CCD *barcode scanner* dapat membaca *barcode* secara cepat dan mudah. Tetapi memiliki 2 keterbatasan. Pertama, memiliki rentang baca yang pendek, dan tidak boleh lebih dari 1-3 inchi dari *barcode*. Selanjutnya, CCD memiliki keterbatasan lebar dan tidak akan membaca *barcode* yang lebih lebar dari permukaan *scanner*. CCD *barcode scanner* paling banyak digunakan untuk aplikasi penjualan.

- **Kamera berbasis *barcode reader***

Menggunakan video kamera kecil untuk menangkap gambar *barcode*. *Barcode reader* kemudian menggunakan teknik pemrosesan gambar digital untuk menguraikannya.