

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **II.1. Konsep Dasar Sistem**

Menurut Tata Sutabri (2005 : 2) Konsep yang terkandung di dalam definisi sistem adalah konsep sinergi. Konsep ini mengandaikan bahwa di dalam suatu sistem output dari suatu organisasi di harapkan lebih besar daripada output individual atau output masing-masing bagian. Kegiatan bersama dagian yang terpisah, tetapi saling berhubungan secara bersama-sama akan menghasilkan efek total yang lebih besar daripada jumlah bagian secara individu dan terpisah. Ini berarti bahwa 2 ditambah 2 tidak sama dengan 4, tetapi memungkinkan sama dengan 5 atau lebih. Oleh karena itu, sistem organisasi mengutamakan pekerjaan-pekerjaan di dalam tim. Selain itu, cara pandang sistem mensyaratkan suatu pelaksanaan pekerjaan secara integratif baik menyangkut manusia, perkakas, metode, maupun sumber daya yang dimanfaatkan. Karena itu, ada berbagai macam cara untuk mengkategorikan suatu sistem. Ada sistem terbuka atau tertutup, sistem manusia, sistem mesin, atau gabungan antara sistem manusia dan mesin, sistem deterministik atau probabilistik dan masih banyak lagi.

Istilah "Sistem" sekarang ini banyak dipakai, banyak orang berbicara mengenai sistem perbankan, sistem akuntansi, sistem inventori, sistem persediaan, sistem pemasaran, sistem pendidikan, sistem perangkat lunak, sistem tata surya, sistem teologi, dan masih banyak lagi. Sebuah sistem terdiri atas bagian-bagian atau komponen yang terpadu untuk suatu tujuan. Model dasar dari bentuk sistem

ini adalah adanya masukan, pengolahan, dan keluaran. Akan tetapi sistem ini dapat dikembangkan hingga menyertakan media penyimpanan. Sistem dapat terbuka dan tertutup, akan tetapi sistem informasi biasanya adalah sistem terbuka, artinya sistem tersebut dapat menerima beberapa masukan dari lingkungan luarnya.

### **II.1.1. Pengertian Sistem**

Menurut Tata Sutabri (2005 : 8) suatu sistem pada dasarnya adalah sekelompok unsure yang erat hubungannya satu dengan yang lain, yang berfungsi bersama-sama untuk mencapai tujuan tertentu.

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu (Jogiyanto Hartono ; 2005 : 2).

### **II.1.2. Karakteristik Sistem**

Menurut Tata Sutabri (2005 : 11) Model umum sebuah sistem adalah *input*, *proses*, dan *output*. Hal ini merupakan konsep sebuah sistem yang sangat sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Selain itu sebuah sistem memiliki karakteristik atau sifat-sifat tertentu, yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem.

Adapun karakteristik yang dimaksud adalah sebagai berikut :

#### **a. Komponen Sistem (*Components*)**

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan.

Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem. Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar, yang disebut “supra sistem”.

b. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisah-pisahkan.

c. Lingkungan Luar Sistem (*Environment*)

Bentuk apapun yang ada di luar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem.

d. Penghubung Sistem (*Interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau *interface*. Penghubung ini memungkinkan sumber-sumber daya yang mengalir dari satu subsistem ke subsistem lain.

e. Masukan Sistem (*Input*)

Energi yang dimasukkan kedalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

f. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain.

g. Pengolah Sistem (*Process*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

h. Sasaran Sistem (*Objective*)

Suatu sistem memiliki suatu tujuan dan sasaran yang pasti dan bersifat deterministik.

## **II.2. Konsep Dasar Informasi**

Menurut Tata Sutabri (2005 : 23) Informasi adalah sebuah istilah yang tidak tepat dalam pemakaiannya secara umum. Informasi dapat mengenai data mentah, data tersusun, kapasitas sebuah saluran komunikasi, dan lain sebagainya. Informasi ibarat darah yang mengalir di dalam tubuh suatu organisasi sehingga informasi ini sangat penting di dalam suatu organisasi. Suatu sistem yang kurang mendapatkan informasi akan menjadi luruh, kerdil, dan akhirnya mati.

### **II.2.1. Pengertian Informasi**

Menurut Jogiyanto Hartono (2005 : 8) Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

Informasi adalah data yang telah diklasifikasi atau diolah atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi mengolah data menjadi informasi atau tepatnya mengolah

data dari bentuk tak berguna menjadi berguna bagi penerimanya. Bila tidak ada pilihan atau keputusan, maka informasi menjadi tidak diperlukan. Keputusan dapat berkisar dari keputusan berulang sederhana sampai keputusan strategis jangka panjang. Nilai informasi dilukiskan paling berarti dalam konteks sebuah keputusan (Tata Sutabri ; 2005 : 23)

Kualitas Informasi :

a. Akurat (*accurate*)

Informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai penerima informasi kemungkinan banyak terjadi gangguan (*noise*) yang dapat mengubah atau merusak informasi tersebut.

b. Tepat waktu (*Timelines*)

Informasi yang datang pada si penerima tidak boleh terlambat. Informasi yang sudah usang tidak akan mempunyai nilai lagi karena informasi merupakan landasan dalam pengambilan keputusan. Bila pengambilan keputusan terlambat, maka dapat berakibat fatal bagi organisasi.

c. Relevan (*Relevance*)

Informasi tersebut mempunyai manfaat untuk pemakainya. Relevansi informasi untuk orang satu dengan yang lain berbeda.

### **II.3. Pengertian Sistem Informasi**

Menurut Tata Sutabri (2005 : 42) Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan data yang mendukung fungsi operasional yang bersifat manajerial dengan kegiatan strategi dari satu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan.

Sistem informasi adalah suatu sistem didalam 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 di perlukan (Jogiyanto Hartono ; 2005 : 11).

### **II.4. Sistem Informasi Geografis**

#### **II.4.1. Sejarah Sistem Informasi Geografis**

Menurut Supriadi (2008 : 1) Sebelum tahun 1940-an analisis geografis melakukan tumpang tindih (*overlay*) beberapa jenis peta pada area tertentu. Namun sejak tahun 1950-an dikembangkan sistem digital untuk melakukan analisis dalam peranan memecahkan permasalahan keruangan. Hingga kini berbagai peranan untuk mengatasi berbagai aspek permasalahan yang berkaitan dengan ruang.

Perkembangan di bidang teknologi komputer telah membawa manfaat yang sangat besar bagi penyebaran informasi. Dengan internet misalnya, kita

dapat melihat tempat-tempat yang indah disegala penjuru dunia bila tampilannya memanfaatkan sistem informasi geografi.

#### **II.4.2. Pengertian Sistem Informasi Geografis**

Menurut Supriadi (2008 : 1) Sistem Informasi Geografis (SIG) adalah bahagian daripada sistem informasi yang diaplikasikan untuk data geografi atau alat database untuk analisis dan pemetaan sesuatu yang terdapat dan terjadi di bumi.

Sistem Informasi Geografis (SIG) merupakan salah satu model sistem informasi yang banyak digunakan untuk membuat berbagai keputusan, perencanaan dan analisis.

SIG memiliki perbedaan pokok dengan sistem informasi lain. Perbedaan ini justru menjadi ciri karakteristiknya. Pada sebuah sistem informasi selain SIG, basis data atributal adalah fokus dari pekerjaan sistem, sedangkan SIG mengaitkan data atributal dengan data spasial. SIG member analisis keruangan terhadap data atribut tersebut. SIG menjelaskan dimana, bagaimana dan apa yang akan terjadi secara keruangan yang diwujudkan dalam gambaran peta dengan berbagai penjelasan secara deskriptif, tabular dan grafis.

Dari kemampuannya tersebut, SIG memberi dua jenis model informasi, yaitu dalam bentuk spasial dan deskriptif. Hubungan antara bentuk spasial dan deskriptif dijelaskan secara topologis (Eko Budiyanto ; 2004 : 1).

### II.4.3. Perolehan Data Sistem Informasi Geografis

Menurut Eko Budiyanto (2004 : 5) Sistem Informasi Geografis memanfaatkan berbagai macam sumber data baik spasial ataupun atribut dalam bentuk tabular ataupun deskriptif. Berbagai bentuk data ini digunakan secara bersama-sama ataupun sendiri-sendiri dalam proses analisisnya. Sistem informasi geografis bekerja dengan tipe data raster dan data vector. Data raster secara umum dikenal sebagai image atau citra atau gambar. Data raster terdiri dari kumpulan piksel yang diwujudkan dalam nilai-nilai spectral. Nilai spectral terendah adalah nilai 0 yang secara visual akan tampak sebagai warna hitam. Nilai spectral tertinggi adalah 255 yang secara visual tampak sebagai warna putih. Data vector dalam sistem informasi geografis dikenal beberapa tipe, yaitu tipe titik, tipe garis, dan tipe polygon. Tipe titik (point) digunakan untuk menggambarkan fenomena seperti kota, mata air, puncak, gunung, dan sebagainya. Tipe garis (line) digunakan untuk menggambarkan fenomena yang berupa garis seperti jalan, rel kereta api, dan sungai. Tipe polygon sering digunakan untuk menggambarkan fenomena berupa wilayah seperti penggunaan lahan, administrasi, penutup lahan, dan sebagainya.

## II.5. *MapInfo*

### II.5.1. Pengertian *MapInfo*

Menurut Eko Budiyanto (2004 : 19) MapInfo adalah *software* pengolah data spasial yang banyak digunakan dalam analisis Sistem Informasi Geografis. *Software* ini memiliki kemampuan seperti *software-software* pengolah data spasial lain seperti *MapInfo* atau *ArcView*. *MapInfo* merupakan *software* pengolah data

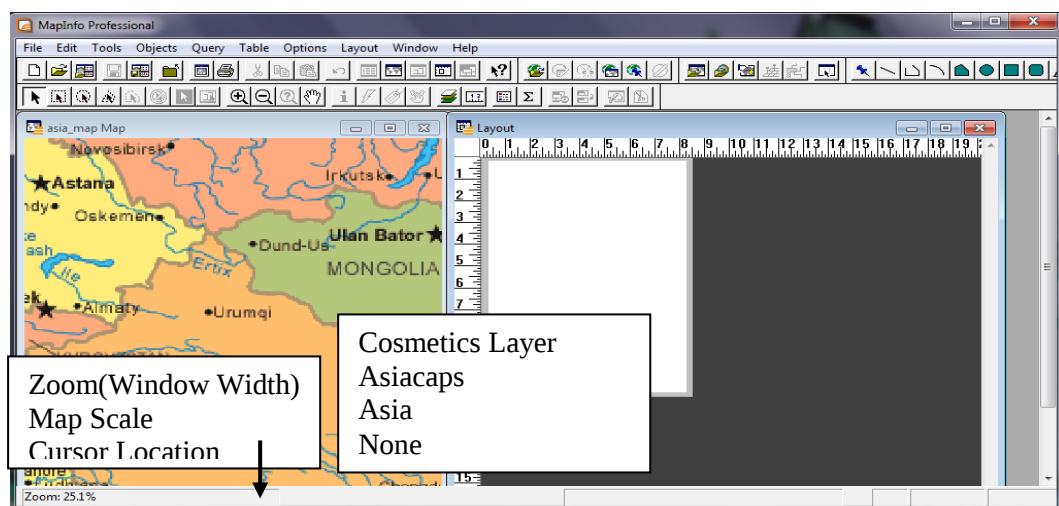
spasial yang terpadu dengan data tabel. Melalui *software MapInfo* operator dapat membuat, menampilkan, serta mengadakan perubahan terhadap data spasial atau peta.

*MapInfo* bersifat fleksibel dalam penampilan dan perubahan data yang antara lain mencakup :

1. Pembukaan banyak tabel dalam waktu yang bersamaan.
2. Pengendali *property layer* secara individual.
3. Mampu membuat dan memodifikasi peta-peta tematik yang ada.
4. Pencarian informasi terkait dengan data spasial.
5. Sistem kendali proyek peta.

## II.5.2. User interface MapInfo

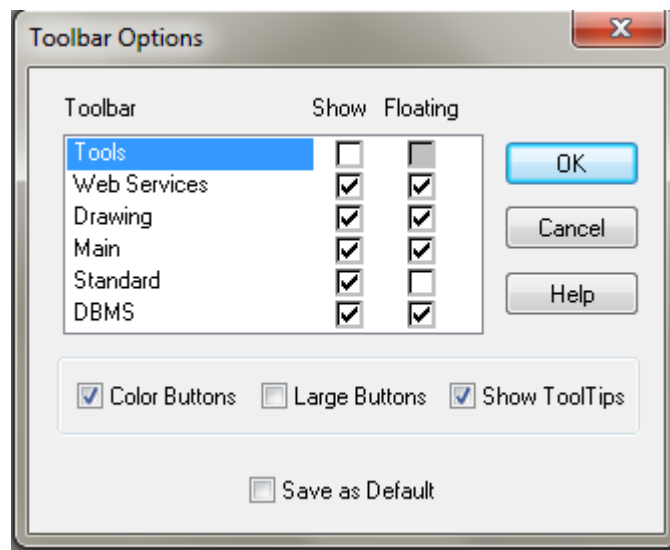
Menurut Eddy Prahasta (2006 : 7) *Mapinfo professional* hadir bersama dengan beberapa *user interface* yang diimplementasikan dalam berbagai bentuk. Berikut adalah sebagian di antaranya yang sering dijumpai oleh pengguna.



**Gambar II.1. Contoh Tampilan User Interface Mapinfo Profesional**

Sumber : Eddy Pahasta ( 2006 : 7)

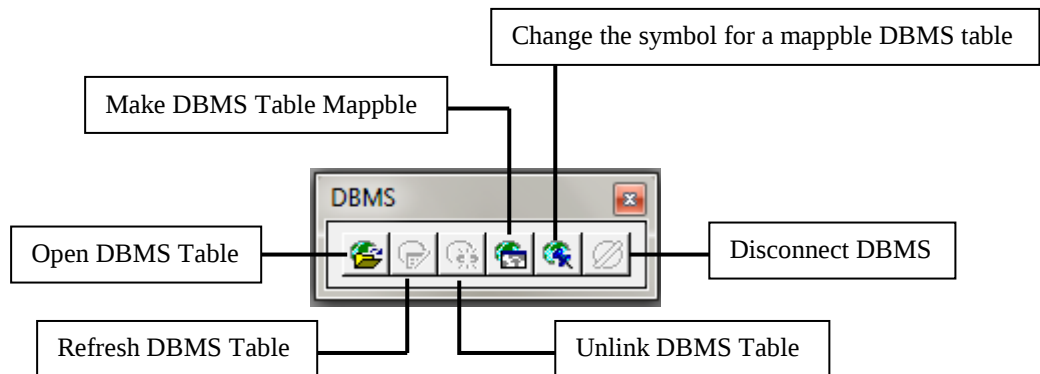
Selain itu, di dalam *MapInfo*, pengguna juga dapat menjumpai sejumlah *user interface* yang di implementasikan dalam bentuk kotak dialog *floating* yang berisi beberapa *tool (icon)* di dalamnya kotak dialog ini dapat diaktifkan dengan cara menggunakan menu “*Option | Toolbars* “ sehingga muncul kotak dialog “*Toolbar Options*”



**Gambar II.2. Tampilan Kotak Dialog “Toolbar Options”**

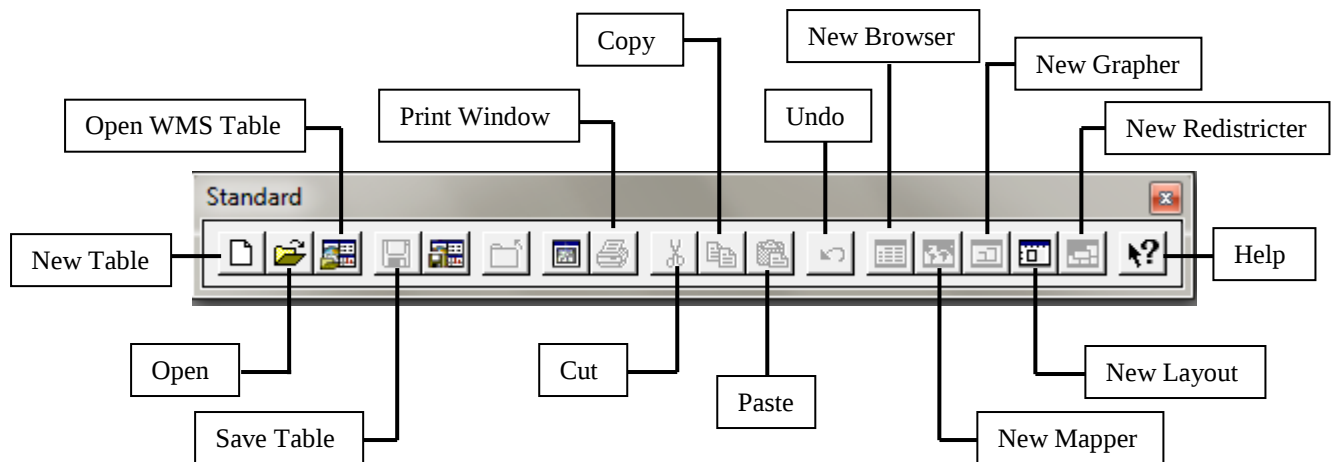
Sumber : Eddy Pahasta ( 2006 : 8)

Kemudian pengguna dengan menggunakan kotak “*Toolbar Options*” di atas dapat memilih sekaligus beberapa kotak dialog *floating* yang akan diaktifkan pada saat yang sama.



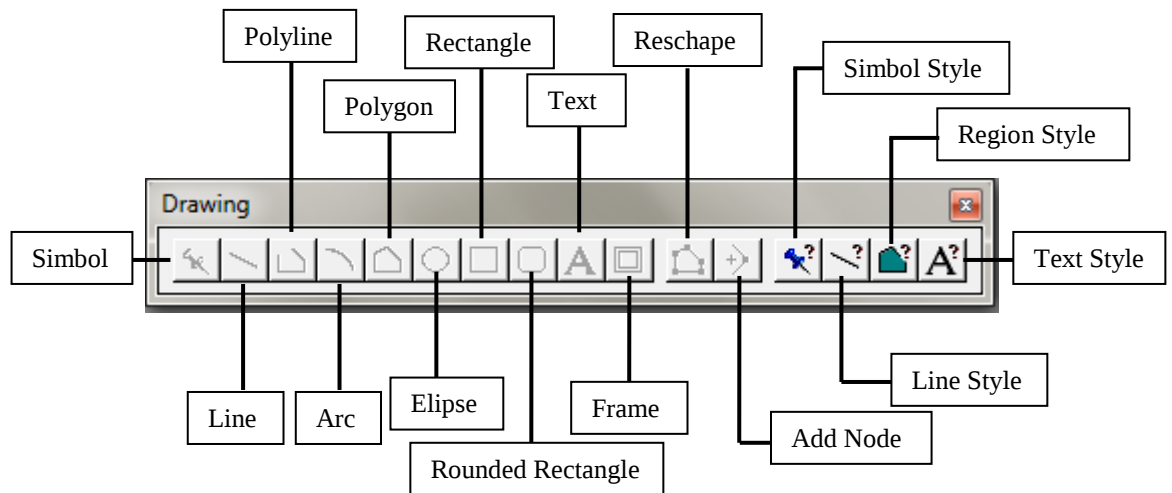
**Gambar II.3. Tampilan Floating Toolbar “DBMS”**

Sumber : Eddy Pahasta ( 2006 : 9)



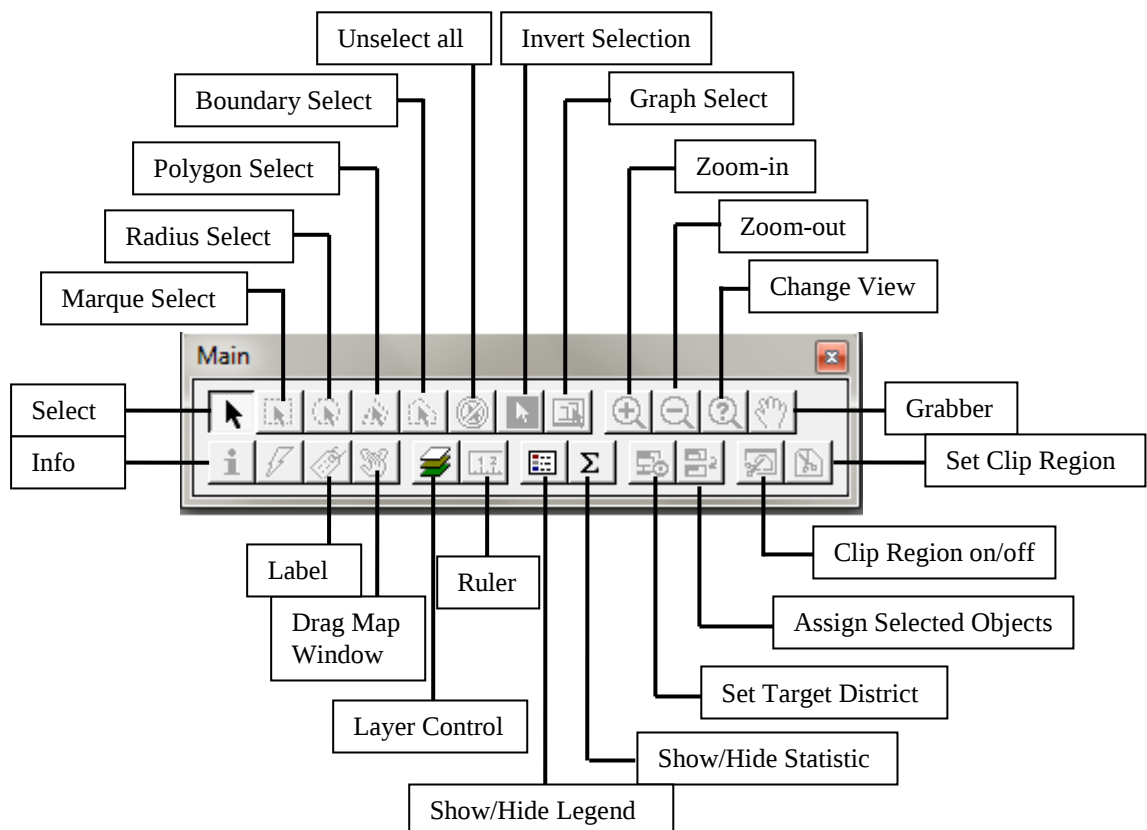
**Gambar II.4. Tampilan Floating Toolbar “Standar”**

Sumber : Eddy Pahasta ( 2006 : 9)



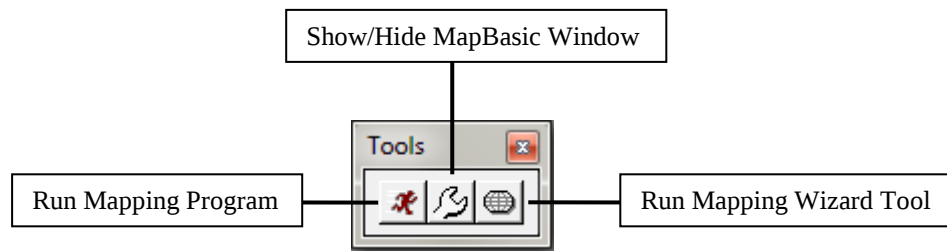
**Gambar II.5. Tampilan Floating Toolbar “Drawing”**

Sumber : Eddy Pahasta ( 2006 : 9)



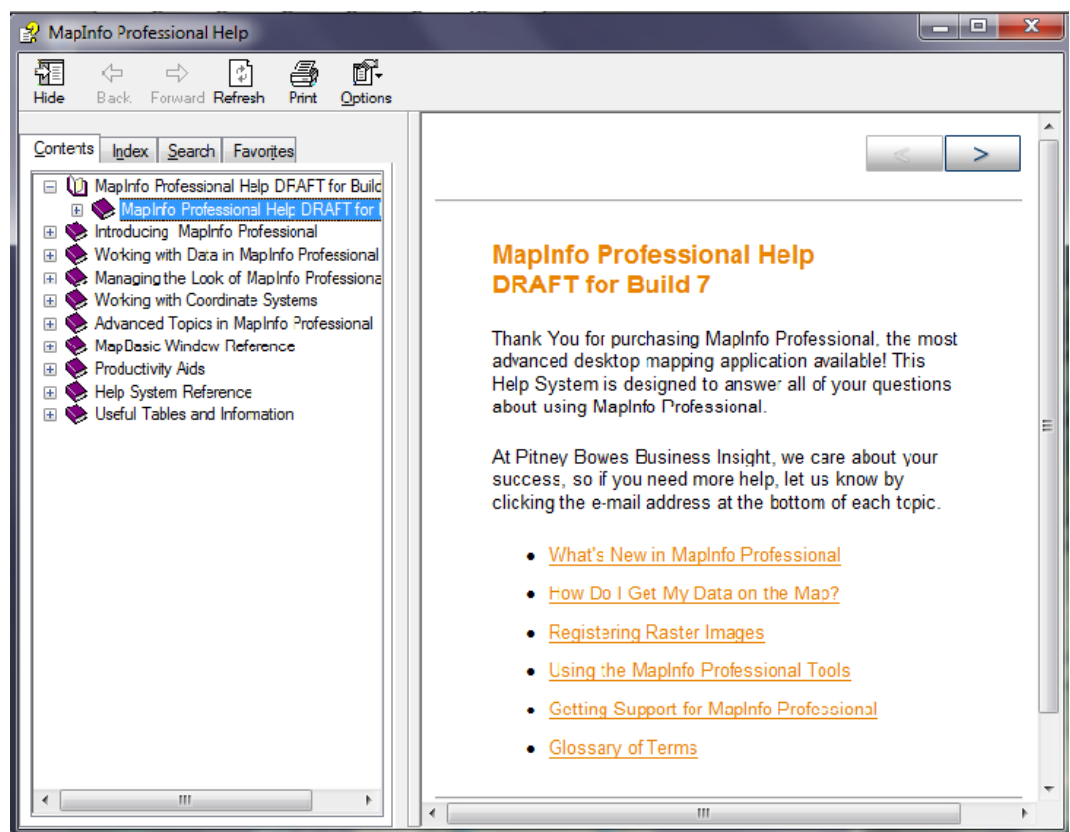
**Gambar II.6. Tampilan Floating Toolbar “Main”**

Sumber : Eddy Pahasta ( 2006 : 10)



**Gambar II.7. Tampilan Floating Toolbar “Tools”**

Sumber : Eddy Pahasta ( 2006 : 10)



**Gambar II.8. Tampilan Kotak Dialog “Help”**

Sumber : Eddy Pahasta ( 2006 : 11)

## II.6. Peta

### II.6.1. Konsep Dasar Peta

Menurut Supriadi (2008 : 15) *Output* terpenting dari Sistem Informasi Geografis adalah peta yang menggambarkan jenis, penampakan dan informasi yang beragam. Peta menampilkan rona geografis atau fenomena keruangan lainnya melalui gambar yang menyangkut informasi tentang lokasi dan atributnya.

Tujuan dasar pemetaan adalah untuk menyediakan :

1. Penjelasan fenomena geografis.
2. Informasi keruangan dan bukan keruangan.
3. *Feature* peta seperti titik, garis dan area.

### II.6.2. Ciri (*Feature*) Peta

Menurut Supriadi (2008 : 15) Informasi lokasi biasanya ditampilkan berupa titik (*point*), garis (*line*), area (*polygon*).

1. Titik (*points*) untuk menggambarkan lokasi tunggal yang terlalu kecil untuk diwujudkan sebagai garis atau area.
2. Garis (*line*) untuk menggambarkan alur, jalur pipa, garis kontur. *Feature* garis (*line*), merupakan sekumpulan hubungan koordinat ditampilkan sebagai garis yang terlalu pipih untuk ditampilkan sebagai sebuah area, seperti jalan atau garis kontur.
3. Area (*polygon*) untuk menggambarkan danau, daerah dan lokasi sensus. *Feature* area (*polygon*) merupakan gambar tertutup yang membatasi daerah yang homogen.

### II.6.3. Jenis Peta

Menurut Supriadi (2008 : 16) Peta merupakan gambaran sederhana akan keberadaan di muka bumi. Peta dapat dibagi dalam dua kelompok besar yaitu :

#### 1. Peta Topografi (*topographical map*)

Merupakan peta yang menunjukkan Penampakan alamiah dan buatan manusia di bumi yang diwujudkan oleh kontur, lahan, jalan, kereta api dan berbagai *feature* penting lainnya. Kondisi muka bumi ini dapat ditayangkan dalam berbagai jenis peta, seperti peta topografi dan peta kontur. Peta topografi menunjukkan penampakan fisik permukaan bumi, seperti jalan, sungai dan bangunan. Sedangkan peta kontur menampakan garis yang menghubungkan titik tertentu yang memiliki kesamaan nilai, misalnya ketinggian tempat dari permukaan laut dan isobar (menunjukkan tekanan udara).

#### 2. Peta Tematik (*Thematic Map*)

Merupakan sumber yang penting dari informasi SIG. Peta tematik merupakan sarana untuk menyampaikan konsep geografis dengan thema tertentu, seperti kepadatan populasi, iklim, jenis tanah, geologi, kesesuaian lahan, pergerakan barang, dan manusia serta penggunaan lahan.

### II.7. UML (*Unified Modeling Language*)

Menurut Munawar (2005 : 17) *Unified Modeling Language* adalah salah satu alat bantu yang sangat handal di dunia pengembangan sistem yang berorientasi obyek. Hal ini disebabkan karena *UML* menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang sistem untuk membuat

cetak biru atas visi mereka dalam bentuk yang baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (*sharing*) dan mengkomunikasikan rancangan mereka dengan yang lain.

Ada beberapa diagram dalam *Unified Modeling Language*, diantaranya adalah :

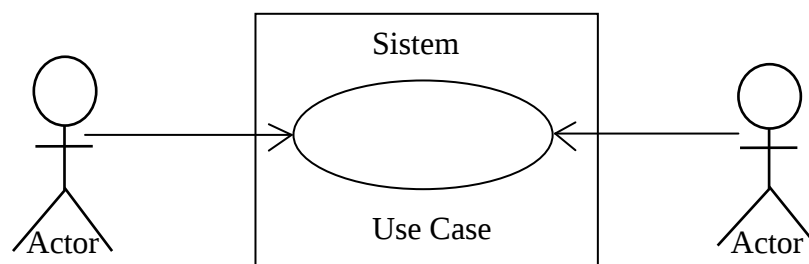
### 1. *Use Case Diagram*

*Use Case* adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna. *Use Case* bekerja dengan cara mendeskripsikan tipikal interaksi antara *user* (pengguna) sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai.

Dalam pembicaraan tentang *Use Case*, pengguna biasanya disebut dengan aktor. Aktor adalah sebuah peran yang bisa dimainkan oleh pengguna dalam interaksinya dengan sistem.

Diagram *Use Case* menunjukkan 3 aspek dari sistem yaitu aktor, *use case*, dan sistem/sub sistem boundary. *Actor* mewakili peran orang, sistem yang lain atau alat ketika berkomunikasi dengan *use case*.

Adapun ilustrasi *actor*, *use case*, dan *boundary* dapat dilihat pada gambar II.9. berikut ini :



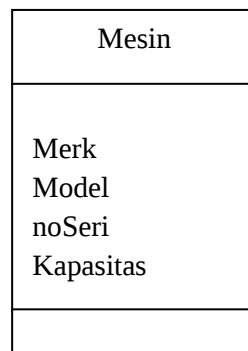
**Gambar II.9. Use Case Model**

Sumber : Munawar ( 2005 : 64 )

## 2. Class Diagram

*Class Diagram* sangat membantu dalam visualisasi struktur kelas dari suatu sistem. Hal ini disebabkan karena *class* adalah deskripsi kelompok obyek-obyek dengan *property*, perilaku (*operasi*) dan relasi yang sama.

Adapun contoh *class* dan atributnya dapat dilihat pada gambar II.10. berikut ini :



**Gambar II.10. Class dan atributnya**

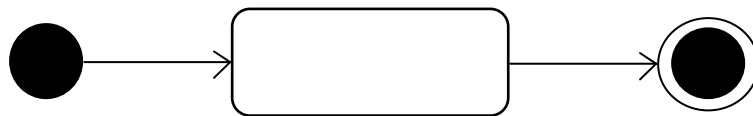
Sumber : Munawar (2005 : 36)

## 3. Statechart Diagram

*Interaction diagram* dan *statechart* menampilkan dua pandangan yang saling melengkapi tentang perilaku dinamis sebuah sistem. *Interaction diagram* menunjukkan pesan-pesan yang dilewatkan diantara obyek-obyek di dalam sistem selama periode waktu yang pendek. Sedangkan *statechart* diagram menelusuri individu-individu obyek melalui keseluruhan daur hidupnya, menspesifikasikan semua urutan yang mungkin dari pesan-pesan yang akan diterima obyek tersebut, bersama-sama dengan tanggapan atas pesan-pesan tersebut.

*State diagram* menyediakan variasi simbol dan sejumlah ide untuk pemodelan. Tipe diagram ini mempunyai potensi untuk menjadi sangat kompleks dalam waktu yang singkat.

Adapun simbol-simbol yang terdapat pada *statechart diagram* dapat dilihat pada gambar II.11. berikut ini :



**Gambar II.11. Simbol-simbol Statechart Diagram**

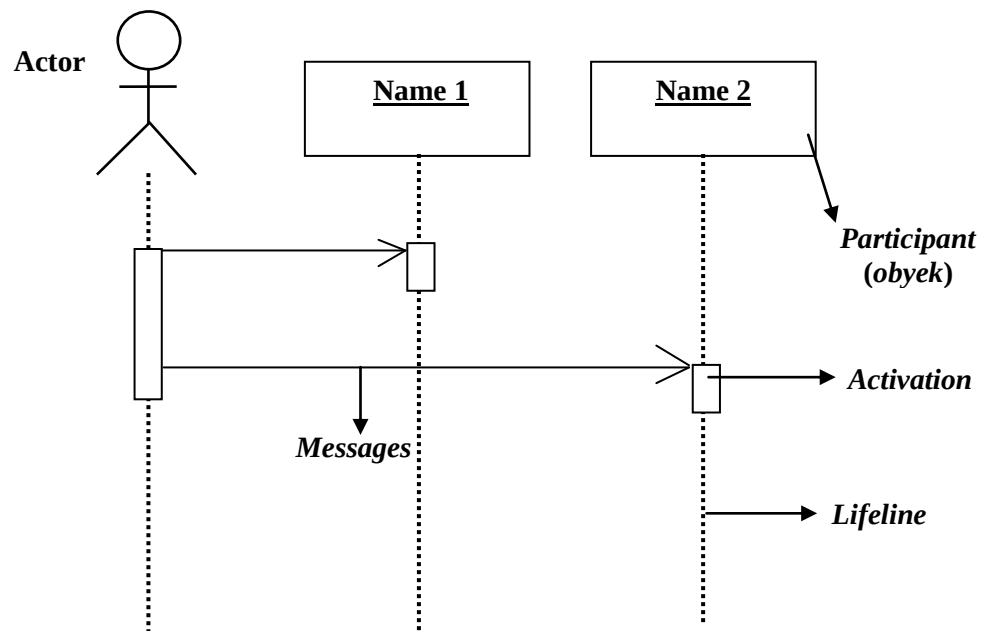
Sumber : Munawar (2005 : 74)

#### 4. *Sequence Diagram*

*Sequence diagram* digunakan untuk menggambarkan perilaku pada sebuah skenario. Dengan ini menunjukkan sejumlah contoh obyek dan *message* (pesan) yang diletakkan diantara obyek-obyek ini di dalam *use case*.

Komponen utama *sequence diagram* terdiri atas obyek yang dituliskan dengan kotak segiempat bernama. *Message* diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan *progress vertical*.

Adapun simbol-simbol yang terdapat pada *sequence diagram* dapat dilihat pada gambar II.12. berikut ini :



**Gambar II.12. Simbol-simbol sequence diagram**

Sumber : Munawar ( 2005 : 89 )

## 5. Collaboration Diagram

*Collaboration diagram* adalah perluasan dari obyek diagram. (Obyek diagram menunjukkan obyek-obyek dan hubungannya satu dengan yang lain). *Collaboration diagram* menunjukkan *message-message* obyek yang dikirimkan satu sama lain.

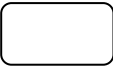
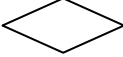
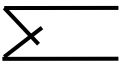
Antara *collaboration diagram* dan *sequence diagram* bisa saling mengisi. Dengan demikian pada *collaboration diagram* kita bisa tambahkan nomor urut pada label sebuah *message* untuk menunjukkan urutan informasi. Titik dua (:) perlu digunakan untuk memisahkan nomor dengan *message*.

## 6. Activity Diagram

*Activity diagram* adalah teknik untuk mendeskripsikan logika *procedural*, proses bisnis dan aliran kerja dalam banyak kasus. *Activity diagram* mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity diagram* bisa mendukung perilaku *parallel* sedangkan *flowchart* tidak bisa.

Adapun simbol-simbol yang terdapat pada *Activity diagram* dapat dilihat pada Tabel II.1. berikut ini :

**Tabel II.1. Simbol-Simbol Activity Diagram**

| Simbol                                                                              | Keterangan                                                                                                                            |
|-------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|
|  | Titik Awal                                                                                                                            |
|  | Titik Akhir                                                                                                                           |
|  | Activity                                                                                                                              |
|  | Pilihan untuk pengambilan keputusan                                                                                                   |
|  | Fork ; Digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu. |
|  | Rake ; Menunjukkan adanya dekomposisi.                                                                                                |
|  | Tanda waktu                                                                                                                           |
|  | Tanda pengiriman                                                                                                                      |
|  | Tanda penerimaan                                                                                                                      |
|  | Aliran akhir (Flow Final)                                                                                                             |

Sumber : Munawar (2005 : 109)

## 7. Component Diagram

*Component* adalah implementasi *software* dari sebuah *class*. *Class* mewakili abstraksi dari serangkaian *attribute* dan *operation*. Hal terpenting tentang *class* dan *component* adalah sebuah *component* bisa jadi merupakan implementasi lebih dari sebuah *class*.

Dengan memodelkan *component* dan relasinya maka :

1. Klien bisa melihat struktur sistem yang sudah selesai.
2. Pengembang mempunyai struktur untuk panduan kerja.
3. Dokumentator bisa memahami apa yang mereka tulis.
4. Siap untuk digunakan kembali untuk proyek lain.

Adapun Notasi *Component* diagram dapat dilihat pada gambar II.13. berikut ini :



**Gambar II.13. Notasi Component**

Sumber : Munawar (2005 : 122)

## 8. Deployment Diagram

*Deployment Diagram* menunjukkan tata letak sebuah sistem secara fisik, menampakkan bagian-bagian *software* yang berjalan pada bagian-bagian *hardware*.

Bagian utama *hardware* / perangkat keras adalah node; yaitu nama umum untuk semua jenis sumber komputasi. Ada 2 tipe node yang mungkin,

*Processor* adalah node yang bisa mengeksekusi sebuah *component*, sedangkan *device* tidak. *Device* adalah perangkat keras tipikalnya menjadi *interface* dengan dunia luar.

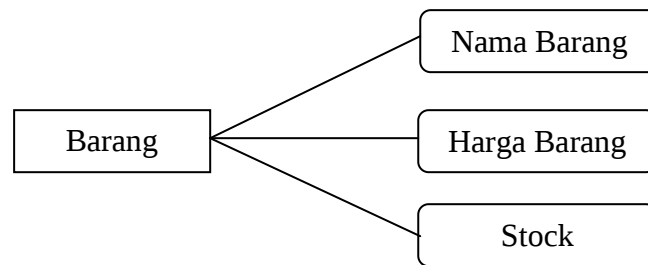
## II.8. *Entity Relationship Diagram (ERD)*

Menurut Kusrini S. kom (2007 : 99) ERD merupakan notasi grafis dalam pemodelan data konseptual yang mendeskripsikan hubungan antar penyimpanan. ERD digunakan untuk memodelkan struktur data dan hubungan antar data, karena hal ini *relative* kompleks. Dengan *ERD* kita dapat menguji model dengan mengabaikan proses yang harus dilakukan. dengan *ERD* kita mencoba menjawab pertanyaan. “ Data apa yang kita perlukan ? Bagaimana data yang satu berhubungan dengan yang lain? “

*ERD* menggunakan sejumlah notasi dan symbol untuk menggambarkan struktur dan hubungan antar data. Pada dasarnya ada 3 macam symbol yang digunakan, yaitu :

1. *Entity* adalah suatu objek yang dapat diidentifikasi dalam lingkungan pemakai, sesuatu yang penting bagi pemakai dalam konteks sistem yang akan dibuat. Sebagai contoh adalah barang, pemasok, pekerja dan lain-lain. Seandainya A adalah barang maka A adalah isi dari barang, sedangkan jika B adalah seorang pelanggan maka B adalah isi dari pelanggan. Karena itu harus dibedakan antara entitas sebagai bentuk umum dari deskripsi tertentu dan isi entitas seperti A dan B dalam contoh di atas. Entitas digambarkan dalam bentuk persegi empat.

2. *Attribut Entitas* mempunyai elemen yang disebut atribut dan berfungsi mendeskripsikan karakter entitas, misalnya atribut nama barang dari entitas barang. setiap *ERD* bias berisi lebih dari satu atribut. Entitas digambarkan dalam bentuk elips.



**Gambar II.14. Atribut Model ERD**

Sumber : Kusrini S. kom (2007 : 99)

3. Hubungan – *relationship* sebagaimana halnya entitas, hubungan pun harus di bedakan antara hubungan atau bentuk hubungan antara entitas dengan isi dari hubungan itu sendiri. Misalnya dalam kasus hubungan antara entitas barang dan entitas pelanggan adalah menjual barang, sedangkan isi hubungannya dapat berupa tanggal jual atau yang lainnya. Relationship digambarkan dalam bentuk intan (*diamonds*).

## **II.9. PHP (*Hypertext Preprocessor*)**

### **II.9.1. Sejarah Singkat PHP**

Menurut mAster.com (2012 : 5) PHP diciptakan pertama kali Oleh Rasmus Lerdorf pada tahun 1994. Awalnya, PHP digunakan untuk mencatat jumlah serta untuk mengetahui siapa saja pengunjung pada *Homepage*-nya. Rasmus Lerdorf adalah salah seorang pendukung *open source*. Oleh Karena itu, ia

mengeluarkan *Personal Home Page Toll* versi 1.0 secara gratis, kemudian menambah kemampuan *PHP* 1.0 dan meluncurkan *PHP* 2.0.

Pada tahun 1996, *PHP* telah banyak digunakan dalam *website* di dunia. Sebuah kelompok pengembang *software* yang terdiri dari Rasmus, Andi Gutman, Stig Bakken, Shane Caraveo, dan Jim Winstead bekerja sama untuk menyempurnakan *PHP* 2.0. Akhirnya, pada tahun 1998, *PHP* 3.0 diluncurkan. Penyempurnaan terus dilakukan sehingga pada tahun 2000 dikeluarkan *PHP* 4.0. Tidak berhenti sampai disini kemampuan *PHP* terus bertambah, dan sampai saat ini versi terbaru *PHP* yang telah dikeluarkan adalah *PHP* 5.0.

### **II.9.2. Kelebihan *PHP***

Menurut mAster.com (2012 : 6) *PHP* memiliki banyak kelebihan yang tidak dimiliki oleh bahasa script sejenis. *PHP* difokuskan pada pembuatan script *server-side*, yang bisa melakukan apa saja yang dapat dilakukan oleh *CGI* (*Common Gateway Interface*), seperti mengumpulkan data dari *form*, menghasilkan isi halaman *web dinamis*, dan kemampuan mengirim serta menerima *cookies*, bahkan lebih dari pada kemampuan *CGI* (*Common Gateway Interface*).

*PHP* dapat digunakan pada semua sistem operasi, antara lain *Linux*, *Unix*, *Microsoft Windows*, *Mac OS*, *RISC OS*. *PHP* juga mendukung banyak *Web Server*, seperti *Apache*, *Microsoft Internet Information Server*, *Personal Web Server*, *Netcape* an *iPlanet server*, *oreilly website pro server*, *audium*, *Xitami*, *OmniHTTPd*, dan masih banyak lagi lainnya, bahkan *PHP* dapat bekerja sebagai suatu *CGI processor*.

### II.9.3. *PHP Programming*

Menurut mAster.com (2012 : 7) Sintak program atau *script PHP* ditulis dalam apitan tanda khusus *PHP*. Ada tiga macam pasangan tag *PHP* yang dapat digunakan untuk menandai blok *script PHP*, yakni sebagai berikut :

1. `<?php ... ?>`
2. `<script language="PHP"> ... </script>`
3. `<? ... ?>`

Cara 1 dan 2 merupakan cara yang paling umum digunakan sekalipun cara 3 tampak lebih praktis karena cara 3 tidak selalu diaktifkan pada konfigurasi file *php*. Ini yang terdapat pada direktori *c:\apache\php*.

### II.10. *Sql Server*

Menurut Emma Utami dan Sukrisno (2008 : 1) *SQL (Structured Query Language)* pada dasarnya adalah bahasa komputer standar yang ditetapkan untuk mengakses dan memanipulasi sistem *database*. Sebuah *database* berisi satu *table* atau lebih dan memiliki nama yang berbeda untuk masing-masing tabel. Masing-masing tabel memiliki satu kolom (*field*) atau lebih dan memiliki baris (*record*). *Query* digunakan untuk mengakses dan mengolah *database*.

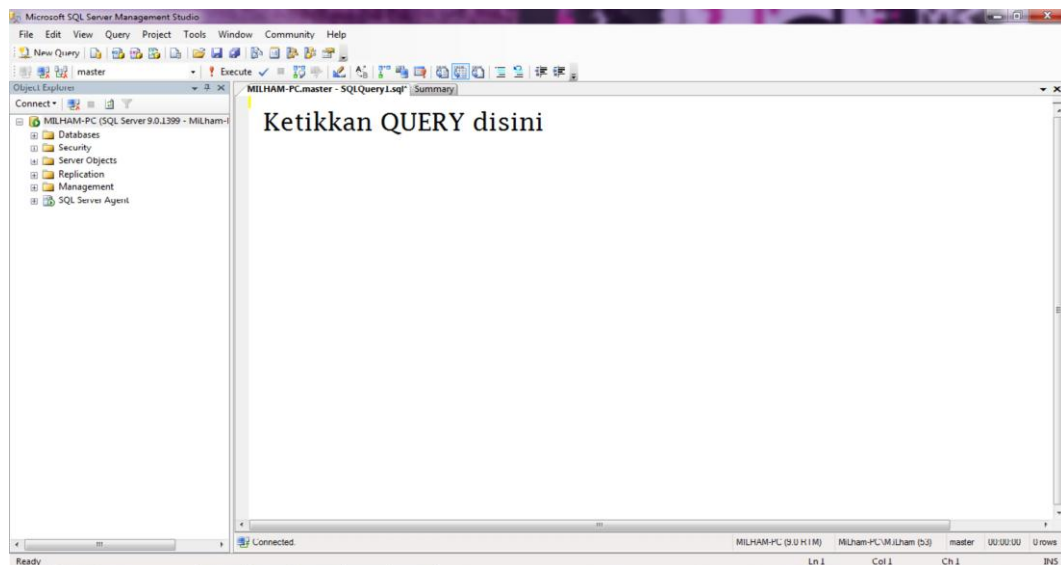
*SQL* terdiri dari 5 (lima) bagian utama, yaitu :

1. Retrieving data: perintah untuk menampilkan data dari *database* (*SELECT*).
2. *Data Definition Language (DDL)*: bahasa yang digunakan untuk membuat atau menghapus tabel atau *database* itu sendiri (*CREATE, DROP, ALTER*).

3. *Data Manipulation Language (DML)*: merupakan bahasa untuk memanipulasi/mengubah isi tabel (*INSERT, DELETE, UPDATE*).
4. *Data Control Language (DCL)*: bahasa yang berhubungan dengan pengendalian akses ke *database* (*GRANT, REVOKE*).
5. *Data Transaction Language (DTL)*: bahasa yang digunakan untuk mengelola transaksi *database* (*COMMIT, ROLLBACK*).

Untuk menjalankan perintah *SQL* kita gunakan *Query Analyzer* yang merupakan *tool* yang telah disediakan oleh *SQL server 2000*.

Berikut adalah tampilan untuk mengetikkan *Query* pada gambar II.16. berikut ini :



**Gambar II.16. Tampilan Untuk mengetikkan Query**

Sumber : Emma Utami dan Sukrisno (2008 : 2)