

BAB II

TINJAUAN PUSTAKA

II.1. Sistem Informasi Geografis.

SIG atau Sistem Informasi Geografis merupakan suatu cabang ilmu pengetahuan yang dibangun atas dasar perpaduan beberapa disiplin ilmu seperti: geografi, ilmu komputer, matematika dan statistik. SIG adalah sebuah sistem yang menangani data dan informasi mengenai kebumihan, baik yang memiliki unsur ruang ataupun deskriptif, dimana sistem ini berfungsi menangkap, menyimpan, memeriksa, memanipulasi, menganalisa dan menayangkan semua data dan informasi.

Dalam dunia sistem informasi terdapat banyak model sistem informasi yang bertujuan akhir memberi berbagai macam informasi. Pentingnya informasi ini memberi banyak inspirasi terhadap pembuat model untuk merancang sistem-sistem yang mendekati dunia nyata dengan hasil sedekat mungkin dengan aslinya. Dari sekian banyak model sistem informasi, 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 memberi 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.

Pengertian SIG dikemukakan oleh beberapa tokoh, diantaranya:

1. SIG adalah suatu sistem informasi yang dapat memadukan antara data grafis dengan data teks (atribut) objek yang dihubungkan secara geografis di bumi (*georeference*). SIG juga dapat menggabungkan data, mengatur data dan melakukan analisis data.

Sumber : (Andi, 2005: 1).

2. Sistem pemetaan berbasis komputer sering disebut sebagai GIS (*Geographic Information System*) atau SIG (Sistem Informasi Geografis) yang merupakan sebuah sistem yang dirancang untuk bekerja dengan data yang terreferensi secara spasial atau koordinat-koordinat geografi. GIS memiliki kemampuan melakukan pengolahan data dan melakukan operasi-operasi tertentu dengan menampilkan dan menganalisa data.

Sumber : (Winarno *et al.*, 2009).

3. Sistem Informasi Geografis adalah sistem informasi khusus yang mengelola data yang memiliki informasi spasial (bereferensi keruangan). Atau dalam arti yang lebih sempit adalah sistem komputer yang memiliki kemampuan untuk membangun, menyimpan, mengelola dan menampilkan informasi bereferensi geografis, misalnya data yang diidentifikasi menurut lokasinya, dalam sebuah database.

Sumber : (Riyanto *et al.*, 2009: 35).

II.1.1. Komponen Sistem Informasi Geografis.

Sistem Informasi Geografis (SIG) merupakan sebuah sistem yang kompleks, sehingga terdapat komponen-komponen yang saling berkaitan dan mendukung. Terdapat 5 (lima) komponen yang ada di dalam SIG yaitu:

1. *Hardware*

SIG membutuhkan komputer untuk penyimpanan dan pemrosesan data dengan spesifikasi yang lebih tinggi dibandingkan dengan sistem informasi lainnya, karena data yang digunakan baik data vektor maupun data raster membutuhkan ruang yang besar serta membutuhkan memori besar dan prosesor yang cepat untuk analisa.

2. *Software*

SIG harus menyediakan fungsi dan *tools* yang mampu melakukan penyimpanan data, analisis dan menampilkan informasi geografis.

3. *Data*

Data terdiri dari data spasial dan data atribut. Data spasial adalah data yang terdiri dari lokasi eksplisit dari suatu geografi yang diset ke dalam bentuk koordinat. Data atribut adalah gambaran data yang terdiri atas informasi yang relevan terhadap suatu lokasi, seperti kedalaman, ketinggian dan lainnya.

4. *Metode*

SIG harus memiliki keserasian antara rencana, desain yang baik dan aturan dunia nyata. Metode yang tepat akan memberikan ketepatan model dan implementasi yang tergantung pada permasalahan yang ada.

5. Manusia

Manusia melakukan perawatan dan pemanfaatan sistem secara baik dan benar agar tujuan sistem tercapai.

Sumber : (Muslim, 2005).

II.1.2. Subsistem Sistem Informasi Geografis.

Beberapa subsistem dalam Sistem Informasi Geografis antara lain :

1. Data *input*

Subsistem ini bertugas untuk mengumpulkan dan mempersiapkan data spasial dan atribut dari berbagai sumber serta dapat mempresentasikan kondisi sebenarnya dalam aplikasi SIG yang dibuat.

2. Data *output*

Subsistem ini menampilkan atau menghasilkan keluaran seluruh atau sebagian basis data baik dalam bentuk *softcopy* maupun bentuk *hardcopy* seperti: table, grafik, peta dan lain-lain.

3. Data management

Subsistem ini mengorganisasikan baik data spasial maupun atribut ke dalam sebuah basis data sedemikian rupa sehingga mudah dipanggil, *diupdate* dan diedit.

4. Data manipulasi dan analisis

Subsistem ini menentukan informasi-informasi yang dapat dihasilkan oleh SIG. Selain itu, subsistem ini juga melakukan manipulasi dan pemodelan data untuk menghasilkan informasi yang diharapkan. Sehingga data dapat

digunakan sesuai dengan kebutuhan dalam pembuatan SIG. Sumber : (Putra, 2012)

II.1.3. Model Data SIG.

Data digital geografis diorganisir menjadi dua bagian, yaitu: data spasial dan data atribut/tabular.

II.1.3.1. Data Spasial

Data yang menyimpan kenampakan-kenampakan permukaan bumi, seperti: jalan, sungai, pemukiman, jenis tanah dan lain-lain. Model data spasial dibedakan menjadi dua yaitu: model data vektor dan model data raster.

1. Data vektor terbentuk dari tiga jenis geometri yakni titik (*point*), garis (*line*), dan area (*polygon*). Oleh karena itu, objek-objek di permukaan bumi perlu divisualisasikan dalam ketiga geometri tersebut agar bisa diproses dengan SIG. Contoh visualisasi dunia nyata menjadi elemen gambar ketiga geometri tersebut antara lain fasilitas sebagai titik, jalan dan sungai sebagai garis, dan daerah administrasi tertentu sebagai area. Berikut ini penjelasan lebih dalam mengenai ketiga geometri tersebut.
 - a. Titik (*point*) meliputi semua objek grafis atau geografis yang dikaitkan dengan pasangan koordinat (x,y). Selain memuat informasi koordinat, data titik juga bisa saja merupakan suatu simbol yang memiliki keterkaitan dengan informasi lain. Sebuah objek titik memiliki satu baris dalam tabel atribut. Karakteristik-karakteristik dari titik ini dijelaskan oleh kolom-kolom yang dibentuk pada tabel atribut. Contoh-contoh objek dunia nyata yang biasa direpresentasikan sebagai titik antara lain kota, pelabuhan,

bandara, rumah sakit, sekolah, dan sebagainya. Perlu diingat bahwa representasi ini sifatnya tidak mutlak melainkan relatif terhadap skala peta. Dalam skala peta yang lebih besar, kota dan bandara bisa saja direpresentasikan sebagai area/luasan (*polygon*).

- b. Garis (*line*) merupakan semua unsur-unsur linier yang dibangun dengan menggunakan segmen-segmen garis lurus yang dibentuk oleh dua titik koordinat atau lebih. Entitas garis yang paling sederhana memerlukan ruang untuk menyimpan titik awal dan titik akhir (dua pasangan koordinat x,y) beserta informasi lain mengenai simbol yang digunakan untuk merepresentasikannya. Garis tunggal yang terbentuk dari titik awal dan titik akhir saja disebut sebagai *line*. Sedangkan garis bersegmen banyak yang terbentuk dari banyak titik (*vertex*) disebut *polyline*. Dalam SIG, baik *line* maupun *polyline* dianggap sebagai suatu entitas yang sama yakni *polyline*. Setiap satu entitas *polyline* memiliki satu baris dalam tabel atribut. Karakteristik dari entitas ini disimpan dalam kolom-kolom tabel atribut. Objek-objek dunia nyata yang sering direpresentasikan sebagai *polyline* antara lain jalan, sungai, jaringan air bersih, jaringan listrik, jaringan telepon, dan sebagainya.
- c. Area (*polygon*) merupakan suatu objek tertutup yang memiliki luasan. *Polygon* dapat direpresentasikan dengan berbagai cara di dalam model data vektor. Karena kebanyakan peta tematik yang digunakan dalam SIG berurusan dengan *polygon*, metode-metode representasi dan manipulasi entitas ini banyak mendapat perhatian. Seperti halnya titik

dan *polyline*, satu objek poligon juga diwakili oleh satu baris pada tabel atribut. Poligon biasanya digunakan untuk merepresentasikan objek dunia nyata yang memiliki luasan seperti wilayah administrasi, danau, jenis tanah, dan sebagainya.

2. Data raster (atau disebut juga dengan sel grid) adalah data yang dihasilkan dari sistem penginderaan jauh. Pada data raster, obyek geografis direpresentasikan sebagai struktur sel grid yang disebut dengan pixel. Pada data raster, resolusi (definisi visual) tergantung pada ukuran pixelnya. Dengan kata lain, resolusi pixel menggambarkan ukuran sebenarnya di permukaan bumi yang diwakili oleh setiap pixel pada citra. Makin kecil ukuran atau luas permukaan bumi yang dapat direpresentasikan oleh setiap pixelnya, makin tinggi resolusi spasialnya. Data raster sangat baik untuk merepresentasikan batas-batas yang berubah secara gradual, seperti jenis tanah, kelembaban tanah, vegetasi, suhu tanah, dan sebagainya. Keuntungan utama dari format data vektor adalah ketepatan dalam merepresentasikan fitur titik, batasan dan garis lurus. Hal ini sangat berguna untuk analisa yang membutuhkan ketepatan posisi, misalnya pada basis data batas-batas kadaster. Contoh penggunaan lainnya adalah untuk mendefinisikan hubungan spasial dari beberapa fitur.

II.1.3.2. Data Atribut.

Data Atribut adalah data yang menyimpan keterangan dari kenampakan-kenampakan permukaan bumi tersebut. Model data tersebut tersimpan ke dalam bentuk baris dan kolom. Data atribut suatu objek dapat berupa data kualitatif dan data kuantitatif.

1. Data Kualitatif

Data kualitatif adalah data hasil pengamatan yang dinyatakan dalam bentuk deskriptif. Data kualitatif dapat diperoleh dari pengisian angket, wawancara, dan tanya jawab. Data kualitatif berfungsi untuk memperlihatkan perbedaan jenis atau rupa.

2. Data Kuantitatif

Data kuantitatif adalah data hasil pengamatan yang dinyatakan dalam bilangan.

Data kuantitatif berfungsi untuk memperlihatkan perbedaan nilai dari objek.

II.1.4. MapInfo Profesional 10.0.

MapInfo pertama kali didirikan oleh empat mahasiswa diantaranya yaitu : Lazio Bardos, Andrew Dressel, John Haller dan Sean O'Sullivan beserta seorang pembimbingnya (Michael Marvin) pada Institute Politeknik Rensselaer pada tahun 1986 di Troy, New York. Perusahaan yang sebelumnya bernama "Navigational Technologies Inc." . Pada asalnya dimaksudkan sebagai salah satu institusi swasta komersial yang mengintegrasikan perangkat lunak (system informasi) geografis dengan sensor-sensor roda dan giroskop pada sistem navigasi kendaraan (in-car). Walaupun demikian, setelah menyadari bahwa pada saat itu sebenarnya adalah tidak mudah untuk berspekulasi usaha pada bidang system navigasi (karena beberapa keterbatasan perangkat keras), maka kelompok usaha ini segera mengalihkan usahanya ke bidang visualisasi (peta digital) di atas platform system computer desktop membangun usaha yang sama sekali baru. Setelah hampir dua dasarwarsa melintang di dunia bisnis digital yang digelutinya

juga sebagai antisipasi perkembangan teknologi, ekonomi dan bisnis informasi yang melanda garapan utama pada perusahaan MapInfo corp. diantaranya adalah :

1. CRM : Customer Relationship Management

Solusi-solusi CRM MapInfo mentransformasikan lokasi-lokasi ke dalam (keuntungan) bisnis hingga memungkinkan bisnis untuk mencari, melayani, dan mengembangkan client nya dengan cara membangun relasi-relasi yang signifikan dan berorientasi jangka panjang.

2. LBS : Location Based Service

MapInfo adalah kekuatan di balik aplikasi-aplikasi pelayanan yang berbasis lokasi mobile location service.

3. LBI : Location Based Intelligence

MapInfo hingga saat ini leading di dalam penyediaan data dan perangkat lunak yang digunakan untuk aplikasi-aplikasi location based intelligence.

4. Aplikasi-aplikasi sector publik

Mapinfo menggunakan location intelligence untuk melindungi manusia beserta asset-assetnya. Sektor public, kepolisian dan pemerintah (Amerika Serikat) masih menggunakan kekuatan teknologi location intelligence milik MapInfo untuk sharing dan kolaborasi informasi. Dengan MapInfo, sector public mendapatkan berbagai perspektif informasi dengan cara visualisasi, analisis dan penyebaran.

Oleh karena itu komitmentnya di dalam bidang nya diatas, pada saat ini MapInfo menjadi salah satu produk perangkat lunak SIG yang paling sukses dipasaran, diantaranya sebagai berikut :

1. MapInfo tersedia di dalam 16 versi bahasa yang berbeda.
2. MapInfo telah terjual ratusan ribu copy di dunia.
3. MapInfo menjadi perangkat lunak standart untuk pemetaan di lingkungan pemerintahan Australia.
4. MapInfo secara defakto menjadi perangkat lunak standart untuk aplikasi-aplikasi telekomunikasi.

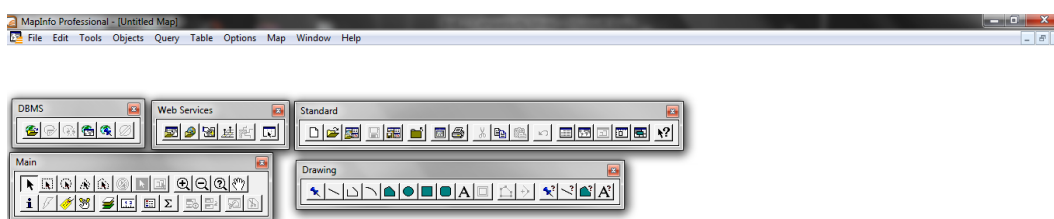
Sehubungan dengan hal diatas, MapInfo merupakan salah satu perangkat lunak pemetaan (SIG) desktop yang dikembangkan dan kemudian dipasarkan untuk memenuhi (sebagian besar) kebutuhan-kebutuhan di lingkungan bisnis.

Sumber : (Ir. Eddy Prahasta MT, 2004, 01-03)

MapInfo adalah software pengolah data spasial dan sistem informasi geografis. Software ini memiliki kemampuan pemetaan digital dan analisis-analisis yang diperlukan dalam sistem informasi geografis.

Sumber : (Eko Budiyanto, 2004, 01-02).

Tampilan umum pada MapInfo :

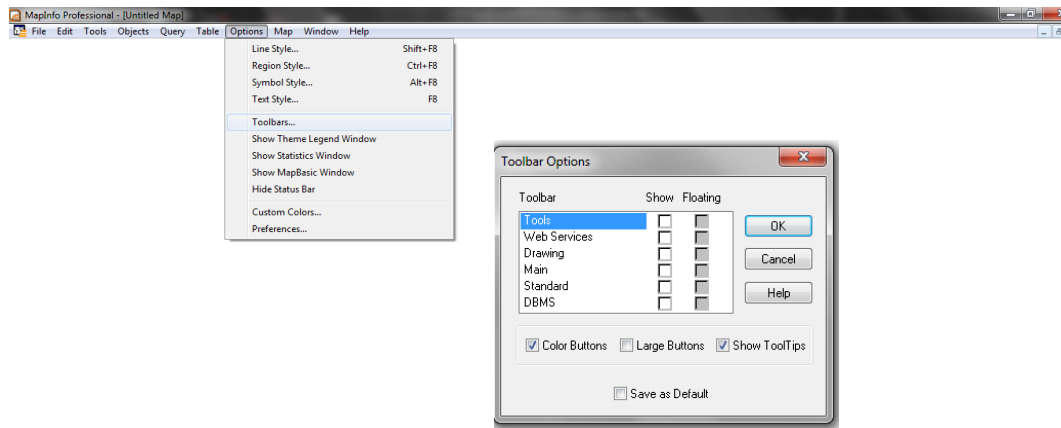


Gambar II.1. Tampilan Umum Aplikasi MapInfo Profesional

Sumber : (Ir. Eddy Prahasta MT, 2004, 07)

Selain di dalam MapInfo Profesional, pengguna juga dapat menjumpai beberapa interface yang di implementasikan dalam bentuk kotak dialog floating yang berisikan beberapa tool-tool (icon) di dalamnya. Kotak dialog ini dapat diaktifkan

dengan cara menggunakan menu “option | Toolbars” hingga muncul kotak dialog “Toolbar Options”. Seperti gambar berikut,

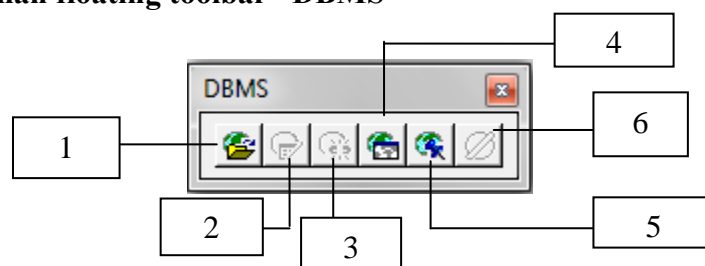


Gambar II.2. Tampilan Gambar Toolbar Options

Sumber : (Ir. Eddy Prahasta MT, 2004, 08)

Kemudian, pengguna dengan menggunakan kotak dialog “Toolbar Options” diatas dapat memilih sekaligus beberapa kotak dialog floating yang akan diaktifkan pada saat yang sama. Berikut beberapa tampilan floating toolbar, diantaranya sebagai berikut :

1. Tampilan floating toolbar “DBMS”



Gambar II.3. Tampilan Gambar Floating Toolbar DBMS

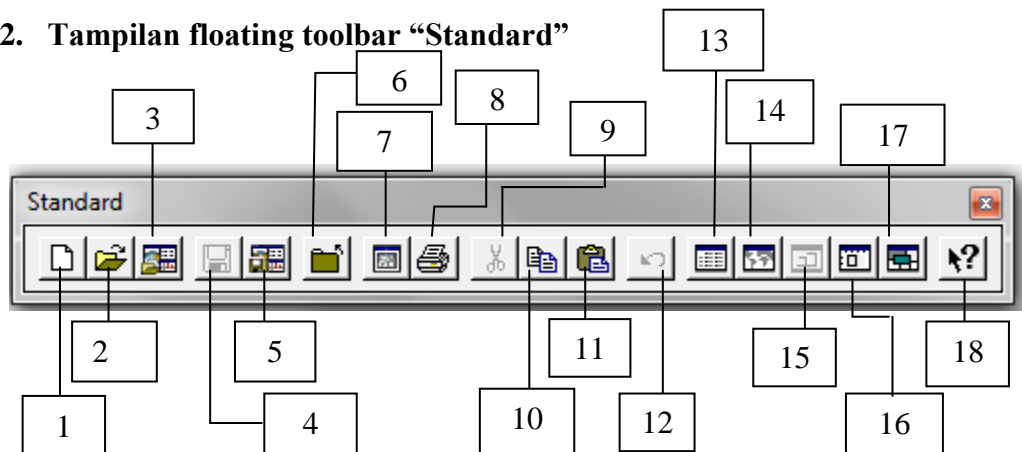
Sumber : (Ir. Eddy Prahasta MT, 2004, 09)

Adapun nama pada tabel dari Gambar II.3 diatas adalah sebagai berikut :

- a. Open DBMS (Nama dari tabel 1 pada gambar II.3)
- b. Refresh DBMS Table (Nama dari tabel 2 pada gambar II.3)

- c. Unlink DBMS Table (Nama dari tabel 3 pada gambar II.3)
- d. Make DBMS Table Mappable (Nama dari tabel 4 pada gambar II.3)
- e. Change the symbol for a mappable DBMS table (Nama dari tabel 5 pada gambar II.3)
- f. Disconnect DBMS (Nama dari tabel 6 pada gambar II.3).

2. Tampilan floating toolbar "Standard"



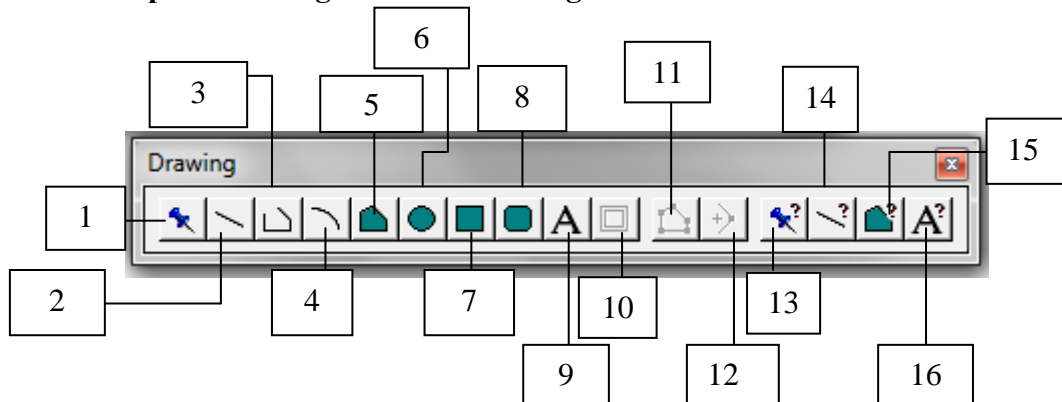
Gambar II.4. Tampilan Gambar Floating Toolbar Standard

Sumber : (Ir. Eddy Prahasta MT, 2004, 09).

Adapun nama pada tabel dari Gambar II.4 diatas adalah sebagai berikut :

- a. New Table (Pada tabel no.1)
- b. Open (Pada tabel no.2)
- c. Open Workspace (Pada tabel no.3)
- d. Save Table (Pada tabel no.4)
- e. Save Workspace (Pada tabel no.5)
- f. Close All (Pada tabel no.6)
- g. Save Window (Pada tabel no.7)
- h. Print (Pada tabel no.8)
- i. Cut (Pada tabel no.9)
- j. Copy (Pada tabel no.10)
- k. Paste (Pada tabel no.11)
- l. Undo (Pada tabel no.12)
- m. NewBrowser (Pada tabel no.13)
- n. New Mapper (Pada tabel no.14)
- o. New Grapher (Pada tabel no.15)
- p. New Layout (Pada tabel no.16)
- q. New Redistricter (Pada tabel no.17)
- r. Help (Pada tabel no.18).

3. Tampilan floating toolbar “Drawing”



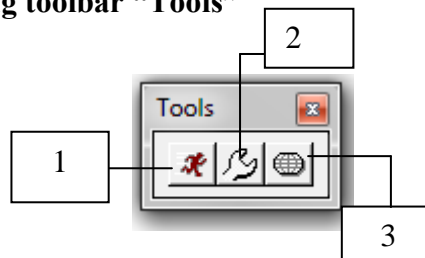
Gambar II.5. Tampilan Gambar Floating Toolbar Drawing

Sumber : (Ir. Eddy Prahasta MT, 2004, 09).

Adapun nama pada tabel dari Gambar II.5 diatas adalah sebagai berikut :

- | | |
|--|------------------------------------|
| a. Symbol (Pada tabel no.1) | g. Text (Pada tabel no.9) |
| b. Line (Pada tabel no.2) | j. Frame (Pada tabel no.10) |
| c. Polyline (Pada tabel no.3) | k. Reshape (Pada tabel no.11) |
| d. Arc (Pada tabel no.4) | l. Add Node (Pada tabel no.12) |
| e. Polygon (Pada tabel no.5) | m. Symbol (Pada tabel no.13) |
| f. Ellipse (Pada tabel no.6) | n. Line Style (Pada tabel no.14) |
| g. Rectangle (Pada tabel no.7) | o. Region Style (Pada tabel no.15) |
| h. Rounded Rectangle (Pada tabel no.8) | p. Text Style (Pada tabel no.16) |

4. Tampilan floating toolbar “Tools”



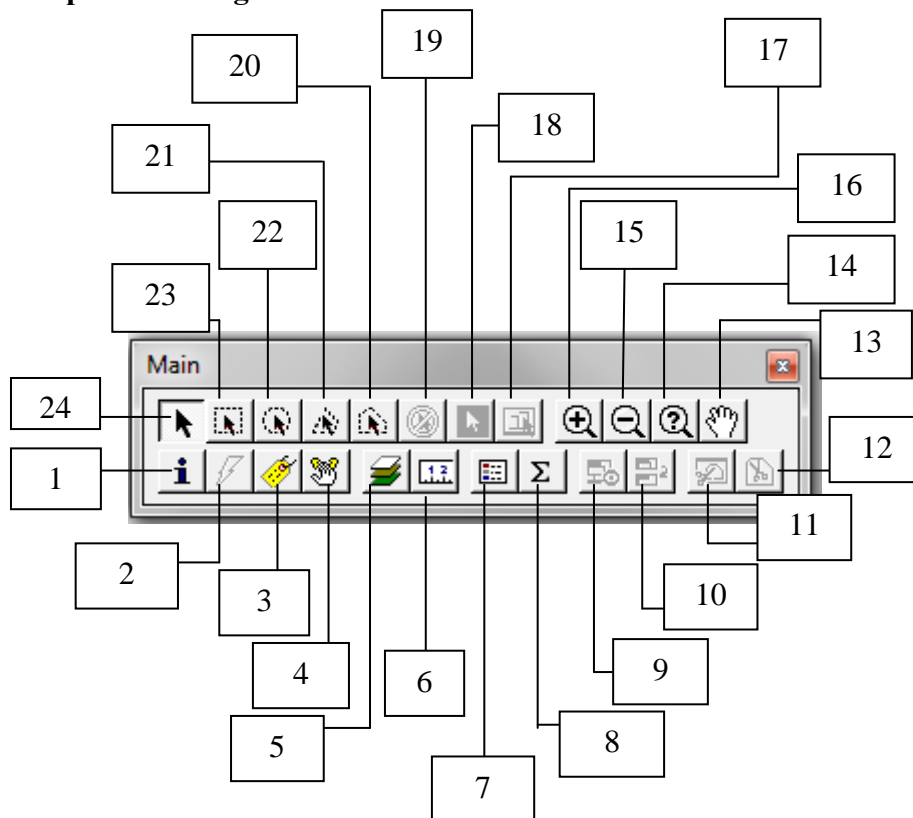
Gambar II.6. Tampilan Gambar Floating Toolbar Tools

Sumber : (Ir. Eddy Prahasta MT, 2004, 10).

Adapun nama pada tabel dari Gambar II.6 diatas adalah sebagai berikut :

- a. Run MapBasic Program (Pada tabel no.1)
- b. Show / Hide MapBasic Window (Pada tabel no.2)
- c. Show / Hide Map Global (Pada tabel no.3)

5. Tampilan floating toolbar “Main”



Gambar II.7. Tampilan Gambar Floating Toolbar Main

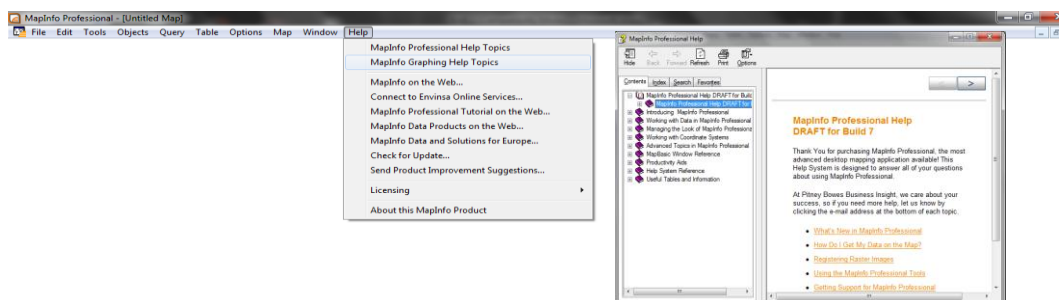
Sumber : (Ir. Eddy Prahasta MT, 2004, 10).

Adapun nama pada tabel dari Gambar II.7 diatas adalah sebagai berikut :

- | | |
|------------------------------------|--------------------------------------|
| a. Info (Pada tabel no.1) | m. Grabber (Pada tabel no.13) |
| b. HotLink (Pada tabel no.2) | n. Change View (Pada tabel no.14) |
| c. Label (Pada tabel no.3) | o. Zoom-out (Pada tabel no.15) |
| d. DragMap Window(Pada tabel no.4) | p. Zoom-in (Pada tabel no.16) |
| e. Layer Control (Pada tabel no.5) | q. Graph Select (Pada tabel no.17) |
| f. Ruler (Pada tabel no.6) | r. InvertSelection(Pada tabel no.18) |

- | | |
|--|---------------------------------------|
| g. Show/HideLegend(Pada tabel no.7) | s. Unselect All (Pada tabel no.19) |
| h. Show/Hide Statistics(Pada tabel no.8) | t. Boundary Select (Pada tabel no.20) |
| i. Set Target District(Pada tabel no.9) | u. Polygon Select(Pada tabel no.21) |
| j. Assign Selected Objects(Pada tabel no.10) | v. Radius Select(Pada tabel no.22) |
| k. Clip Region On/Off (Pada tabel no.11) | w. Marquee Select (Pada tabel no.23) |
| l. Set Clip Region (Pada tabel no.12) | x. Select (Pada tabel no.24) |

Selain diatas, masih banyak lagi implementasi interface yang terbilang cukup penting di dalam MapInfo Profesional. Beberapa diantaranya adalah kotak dialog “Help” yang dapat diaktifkan dengan menggunakan menu “Help” seperti berikut :



Gambar II.8. Tampilan Gambar Salah Satu Kotak Dialog “Help”

Sumber : (Ir. Eddy Prahasta MT, 2004, 10-11).

II.1.4.1. Data Spasial MapInfo

Data spasial utama yang digunakan oleh MapInfo yang menggunakan model vektor diimplementasikan sebagai sebuah table. Data spasial yang di implementasikan sebagai table ini terdiri dari beberapa komponen file sebagai berikut:

1. *.DAT , merupakan file yang digunakan untuk menyimpan data atribut atau tabel milik sebuah tabel.
2. *.TAB , merupakan file utama yang berisi informasi struktur tabel, urutan, nama field, dan tipe field yang terdapat di dalamnya.
3. *.MAP , merupakan file yang berisi informasi geografis yang mendeskripsikan objek-objek peta.
4. *.IND , Merupakan file yang berisi indeks data yang terdapat di dalam tabel atribut terkait (*.DAT).
5. *.ID , merupakan file yang berisi indeks data yang terdapat di dalam objek geografis terkait (*.MAP).

Dengan demikian, sebagai contoh untuk merepresentasikan data spasial (tabel) “Jalan”, MapInfo akan menggunakan file-file: “Jalan.Dat”, “Jalan.Tab”, “Jalan.Map”, “Jalan.Ind”, dan “Jalan.Id”. Sumber: (Ir. Eddy Prahasta MT, 2004, 11-12).

II.2. Macromedia Dreamweaver 8.0

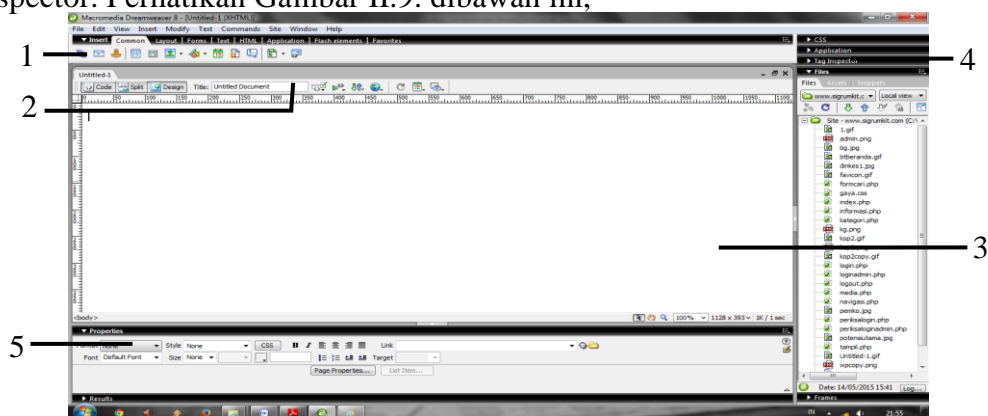
Merupakan salah satu dari bagian aplikasi perancangan web yang digunakan oleh web desainer ataupun web programmer untuk mengembangkan situs web. Ruang kerja, fasilitas dan kemampuan dreamweaver mampu meningkatkan produktivitas dan efektivitas dalam mendesain ataupun membangun situs web. Dreamweaver juga dilengkapi dengan fasilitas yang cukup lengkap. Maka dari itu aplikasi yang akan dibuat menggunakan Macromedia Dreamweaver 8.

II.2.1. Ruang Kerja Dreamweaver 8.0

Pada saat pertama kali memulai Macromedia Dreamweaver 8.0 ini, sebuah kotak dialog workspace akan muncul dan memberikan informasi penawaran untuk memilih tampilan ruang kerja yang di inginkan dan digunakan.

Dreamweaver workspace merupakan ruang kerja yang semua jendela dokumen dan panel integrasi pada satu jendela besar, dengan panel group yang berada disisi kanan. Pilihan ini sering dipakai oleh pengguna. Dan dreamweaver workspace adalah tampilan lama seperti pada dreamweaver versi-versi sebelumnya. Setiap dokumen akan berada pada jendela terpisah.

Ruang kerja pada dreamweaver memberikan fasilitas dan ruang untuk menuangkan kreasi dalam bekerja, seperti pada Gambar II.9. Komponen yang disediakan oleh Macromedia Dreamweaver 8.0 antara lain : Insert Bar, Document Bar, Document Window, Panel Groups, Site Panel, Property Inspector. Perhatikan Gambar II.9. dibawah ini,



Gambar II.9. Tampilan Gambar Jendela Kerja Macromedia Dreamweaver 8.0

Sumber : (Madcoms, 2011, 5).

Berikut bagian-bagian yang terdapat pada jendela kerja pada Macromedia Dreamweaver 8.0 diantaranya sebagai berikut :

a. Insert Bar (Keterangan gambar II.9 pada simbol 1)

Insert Bar merupakan kumpulan menu yang digunakan untuk memasukkan sebuah objek atau fungsi lainnya ke dalam jendela dokumen.

Contohnya : image, layer, tabel dan lain-lain.

Insert Bar memiliki tujuh menu di dalamnya yaitu : common, layout, forms, text, HTML, application dan flash elements. Untuk menampilkan dan menggunakan menu yang terdapat pada Insert Bar, dapat meng-klik tombol drop down yang berada di sebelah kiri pada Insert Bar. Perhatikan gambar di bawah berikut,

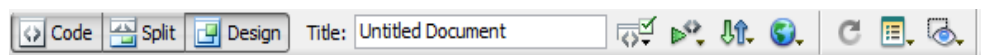


Gambar II.10. Tampilan Gambar Menu Insert Bar

Sumber : (Madcoms, 2011, 6).

b. Document Toolbar (Keterangan gambar II.9 pada simbol 2)

Document Toolbar digunakan sebagai penempatan file-file yang telah dibuka dan sekaligus untuk menampilkan nama dari file tersebut. Document Toolbar juga mempunyai tiga buah tab yang dapat membantu untuk mendesain web dengan mengubah tampilan dari jendela dokumen, diantaranya adalah code, split dan design seperti pada gambar di bawah berikut,

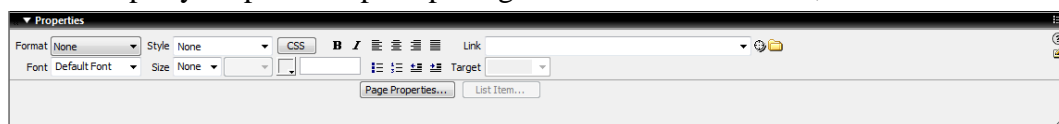


Gambar II.11. Tampilan Gambar Document Toolbar.

Sumber : (Madcoms, 2011, 7).

Adapun keterangan dari tampilan jendela document tersebut, diantaranya adalah :

- 1) Code : merupakan kumpulan kode program yang telah disusun, dan berfungsi untuk mengedit listing dari program.
 - 2) Split : digunakan untuk menampilkan tab dari code view dan design view secara bersamaan.
 - 3) Design : digunakan sebagai tempat untuk meletakkan objek-objek ke dalam jendela document, seperti tabel, form, dan lainnya.
- c. Document Window (Keterangan gambar II.9 pada simbol 3)
- Document Window atau biasa disebut dengan jendela dokumen merupakan tempat untuk menampilkan objek-objek atau kode program dari tab code, split dan design.
- d. Panel Groups (Keterangan gambar II.9 pada simbol 4)
- Panel Group berfungsi untuk mengatur halaman web yang telah dibuat. Contohnya : panel file berfungsi untuk mencari dan mengetahui letak halaman file-file yang telah dibuat.
- e. Property Inspector (Keterangan gambar II.9 pada simbol 5)
- Property Inspector berfungsi untuk mengetahui atau mengubah property dari sebuah objek. Contoh : untuk penulisan sebuah teks dalam perubahan font, colour, dan size dari teks tersebut dengan menggunakan Property Inspector seperti pada gambar dibawah berikut,



Gambar II.12. Tampilan Gambar Property Inspector

Sumber : (Madcoms, 2011, 10).

II.2.2. Komponen Ruang Kerja

Didalam ruang kerja Macromedia Dreamweaver 8.0, terdapat beberapa komponen utama yang membentuk suatu ruang kerja. Berikut keterangan singkat tentang menu yang terdapat pada Macromedia Dreamweaver 8.0. Menu berisi sub menu yang biasanya ditampilkan sebagai drop down menu yang akan muncul ketika menekan tombol menu utama.



Gambar II.13. Tampilan Gambar Menu Pada Macromedia Dreamweaver 8.0

Sumber : (Madcoms, 2011, 6).

Didalam sistem menu tersebut, terdapat beberapa komponen yang dapat digunakan yaitu:

1. File Menu

Berisikan menu standart seperti : new, open, close, save, dan sebagainya

2. Edit Menu

Berisikan perintah seleksi dan pencarian seperti : undo, redo, cut, copy, paste, parent tag, find and replace dan menyediakan akses ke keyboard shortcut editor dan tag library editor.

3. View Menu

Mengijinkan melihat tampilan dokumen dengan beberapa cara (seperti design view dan code view) dan untuk menampilkan serta menyembunyikan beberapa elemen pada dokumen dan peralatan yang dimiliki Macromedia Dreamweaver 8.0.

4. Insert Menu

Menyediakan alternatif untuk memasukkan objek-objek pada dokumen.

5. Modify Menu

Untuk mengubah properti elemen halaman web atau item lainnya yang dipilih. Menggunakan menu ini dapat mengedit tag, mengubah table dan elemen table serta menampilkan beberapa tindakan yang dapat dilakukan untuk item pada library.

6. Text Menu

Menyediakan perintah yang dibutuhkan untuk melakukan pemformatan pada teks.

7. Commands Menu

Menyediakan akses untuk beberapa macam perintah, termasuk untuk membuat album foto dengan cepat dan melakukan optimasi gambar dengan menggunakan program lain.

8. Site Menu

Menyediakan item menu untuk membuat, membuka dan mengedit site.

9. Window Menu

Menyediakan akses kesemua panel, inspectors dan jendela pada Macromedia Dreamweaver 8.0.

10. Help Menu

Menyediakan akses ke dokumentasi yang dimiliki oleh Macromedia Dreamweaver 8.0, termasuk cara penggunaan Dreamweaver, membuat ekstensi dan material lainnya yang dapat membantu menggunakan penggunaan aplikasi Dreamweaver 8.0 dengan lebih baik.

II.3. Database (MySQL)

MySQL adalah sebuah program aplikasi untuk membuat database yang umum digunakan pada web server. MySQL banyak digunakan karena kemampuannya dalam menampung data yang cukup besar dan proses loading yang cepat dalam pengambilan data. Selain itu, MySQL bersifat open source dikarenakan tidak perlu membayar untuk penggunaannya. Database MySQL merupakan sebuah fasilitas untuk menyimpan database dan dapat mempermudah pengguna dalam membuat password, login, guest book, contact dan lain-lain.

MySQL bersifat RDBMS (*Relational Database Management System*) RDBMS memungkinkan seorang admin dapat menyimpan banyak informasi ke dalam tabel-tabel, dimana tabel-tabel tersebut saling berkaitan satu sama lain. Keuntungan RDBMS sendiri adalah kita dapat memecah *database* kedalam tabel-tabel yang berbeda. Setiap tabel memiliki informasi yang berkaitan dengan dengan tabel yang lainnya.

Sumber : (Loka Dwiartara ,2012, 6)

II.4. UML (Unified Modelling Language)

UML adalah salah satu alat bantu yang sangat handal di dunia pengembangan system yang berorientasi obyek. Hal ini disebabkan karena UML menyediakan bahasa pemodelan visual yang memungkinkan bagi pengembang system untuk membuat cetak biru atas visi mereka dalam bentuk baku, mudah dimengerti serta dilengkapi dengan mekanisme yang efektif untuk berbagi (sharing) dan mengkomunikasikan rancangan mereka dengan yang lain.

UML merupakan kesatuan dari bahasa pemodelan yang dikembangkan oleh Booch, Object Modelling Technique (OMT) dan Object Oriented Software Engineering (OOSE).

Metode Booch dari Grady Booch sangat terkenal dengan nama metode Design Object Oriented. Metode ini menjadikan proses analisis dan design ke dalam empat tahapan yaitu : identifikasi kelas-kelas dan obyek-obyek, identifikasi semantik dari hubungan obyek dan kelas tersebut, perincian interface dan implementasi. Keunggulan metode Booch adalah pada detil dan kayanya dengan notasi dan elemen.

Pemodelan OMT yang dikembangkan oleh Rumbaugh didasarkan pada analisis terstruktur dan pemodelan entity relationship. Tahapan utama dalam metodologi ini adalah analisis, design system, design object dan implementasi. Keunggulan dari metode ini adalah dalam penotasian yang mendukung semua konsep Object Oriented.

Metode OOSE dari Jacobson lebih memberi penekanan pada use case. OOSE memiliki tiga tahapan, yaitu : membuat model requirement dan analisis, design dan implementasi, dan model pengujian (test model). Keunggulan dari metode ini adalah mudah dipelajari karena memiliki notasi yang sederhana namun mencakup seluruh tahapan dalam rekayasa perangkat lunak.

Dengan UML, metode Booch, OMT, dan OOSE digabungkan dengan membuang elemen-elemen yang tidak praktis ditambah dengan elemen-elemen dari metode lain yang lebih efektif dan elemen-elemen baru yang belum ada pada metode terdahulu sehingga UML lebih ekspresif dan seragam dari metode lainnya.

Maka dari itu, UML sangat penting dikarenakan UML merupakan hasil kerja dari konsorsium (Gabungan beberapa pengusaha/industriawan yang mengadakan perjanjian suatu usaha proyek bersama) berbagai organisasi yang berhasil dijadikan sebagai standar baku dalam OOAD(Object Oriented Analysis & Design). Kontribusi untuk UML telah dihasilkan dari banyak perusahaan-perusahaan ternama diantaranya Digital Equipment Corp, Hewlet-Packard Company, i-Logic, Intellicorp, IBM, Icon Computing, Electronic Data Service Corporation, MCI System House, Microsoft, Oracle, Rational Software, TI, Sterling Software, Taskon A/S, dan sebagainya.

Sumber : (Munawar, 2005, 17-19).

II.4.1. Use Case Diagram

Use Case Diagram adalah gambaran graphical dari beberapa atau semua actor, use-case, dan interaksi diantara komponen-komponen tersebut yang memperkenalkan suatu sistem yang akan dibangun. Use case diagram menjelaskan manfaat suatu sistem jika dilihat menurut pandangan orang yang berada di luar sistem. Diagram ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar.

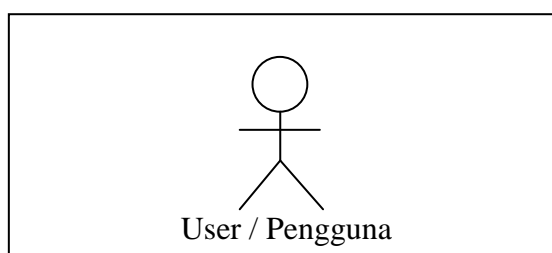
Use case diagram dapat digunakan selama proses analisis untuk menangkap requirement sistem dan untuk memahami bagaimana sistem seharusnya bekerja. Selama tahap desain, use case diagram berperan untuk menetapkan perilaku sistem saat di implementasikan. Dalam sebuah model mungkin terdapat satu atau beberapa use case diagram. Kebutuhan atau requirements system adalah fungsionalitas apa yang harus disediakan oleh sistem kemudian

didokumentasikan pada model use case yang menggambarkan fungsi sistem yang diharapkan (use case), dan yang mengelilinginya (actor), serta hubungan antara actor dengan use case (use case diagram) itu sendiri.

Berikut beberapa komponen pembentuk Use Case Diagram

1. Actor.

Pada dasarnya actor bukanlah dari use case diagram, namun unit dapat tercipta suatu use case diagram diperlukan beberapa actor. Actor tersebut merepresentasikan seseorang atau sesuatu (seperti perangkat, system lain) yang berinteraksi dengan system. Sebuah actor mungkin hanya memberikan informasi inputan pada system, hanya menerima informasi dari system atau keduanya menerima dan member informasi pada system. Actor hanya berinteraksi dengan use-case, tetapi tidak memiliki kontrol atas use-case. Actor digambarkan dengan stick man. Actor dapat digambarkan secara umum, dimana untuk membedakannya kita dapat menggunakan relationship seperti pada gambar berikut :



Gambar II.14. Tampilan Gambar Actor.

Sumber : (Fowler Martin, 2004)

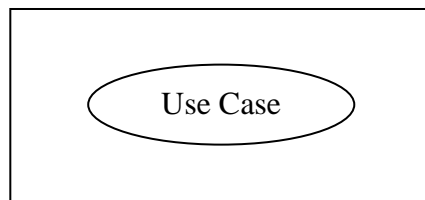
2. Use-case.

Use-case adalah gambaran fungsionalitas dari suatu system, pengguna system paham dan mengerti mengenai kegunaan system yang akan dibangun. Use-case

diagram adalah penggambaran system dari sudut pandang pengguna system tersebut atau yang sering disebut dengan user (pengguna) sehingga pembuatan use-case lebih dititik beratkan pada fungsionalitas yang ada pada system, bukan berdasarkan alur atau urutan kejadian.

Cara menentukan Use-case dalam suatu system, yaitu :

- a. Pola perilaku perangkat lunak aplikasi.
- b. Gambaran tugas dari sebuah actor.
- c. Sistem atau benda yang memberikan sesuatu yang bernilai kepada actor.
- d. Apa yang dikerjakan oleh suatu perangkat lunak (bukan bagaimana mengerjakannya).



Gambar II.15. Tampilan Gambar Use Case.

Sumber : (Fowler Martin, 2004)

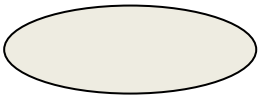
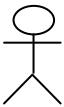



Ada beberapa relasi yang terdapat pada use-case diagram, diantaranya sebagai berikut :

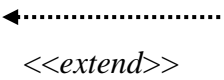
- a. Association, menghubungkan link antar elemen.
- b. Generalization, disebut juga inheritance (pewarisan). Sebuah elemen dapat merupakan spesialisasi dari elemen lainnya.
- c. Dependency, sebuah elemen bergantung dalam beberapa cara ke elemen lainnya.
- d. Aggregation, bentuk association dimana sebuah elemen berisi elemen lainnya.

Tipe relasi yang mungkin terjadi pada use-case diagram :

- a. <<include>>, yaitu kelakuan yang harus terpenuhi agar sebuah kejadian (event) dapat terjadi. Dimana kondisi ini sebuah use-case adalah bagian dari use-case lainnya.
- b. <<extends>>, merupakan kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.
- c. <<communicates>>, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah communicates association. Ini merupakan pilihan selama asosiasi hanya tipe relationship yang dibolehkan antara actor dan use-case.

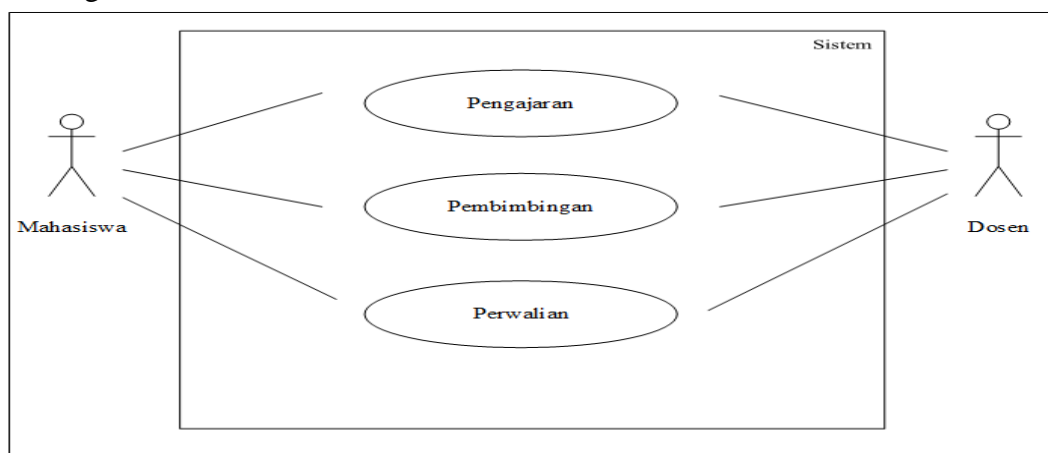
Tabel II.1. Komponen Use Case Diagram

Gambar	Keterangan
	<p><i>Use case</i> menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan actor, biasanya dinyatakan dengan menggunakan kata kerja di awal nama <i>Use case</i>.</p>
	<p>Actor juga dapat berkomunikasi dengan object, maka actor juga dapat diurutkan sebagai kolom. Simbol Actor sama dengan simbol Actor Use Case diagram.</p>
	<p>Asosiasi antara aktor dan <i>use case</i>, digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan aliran data.</p>
	<p>Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk mengindikasikan bila aktor berinteraksi secara pasif dengan sistem.</p>
 <<include>>	<p>Include merupakan kelakuan yang harus terpenuhi agar sebuah kejadian (event) dapat terjadi.</p>

	<p>merupakan kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.</p>
---	---

Sumber : (Grady Booch, 2013)

Kegunaan dari *use case* adalah untuk mendaftarkan *actor-actor* dan *use case-use case* dan memperlihatkan *actor-actor* mana yang berpartisipasi dalam masing-masing *use case*.



Gambar II.16. Tampilan Gambar Use Case

Sumber : (Adi Nugroho, 2010, 34)



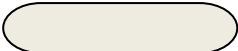
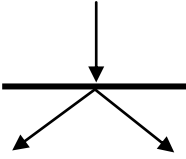
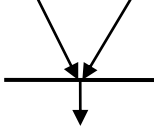

Gambar II.16. memperlihatkan contoh penggambaran diagram *use case*. Ikon-ikon berbentuk orang adalah *actor-actor*-nya sementara elips-elips yang ada menggambarkan fungsionalitas-fungsionalitas sistem. Sebuah (bukan seorang!) *actor* merupakan idealisasi dari orang yang ada di luar sistem, idealisasi proses-proses yang berinteraksi dengan sistem, atau idealisasi sesuatu yang berinteraksi dengan sistem, subsistem, atau kelas, pada sistem/perangkat lunak yang sedang kita kembangkan. Sebuah *actor* pada dasarnya menggambarkan interaksi pengguna-pengguna sistem dengan sistem/perangkat lunak yang sedang kita kembangkan. Saat sistem/perangkat lunak dijalankan, suatu pengguna secara fisik mungkin diikat menjadi *actor* yang ada dalam sistem merupakan kelakuan yang

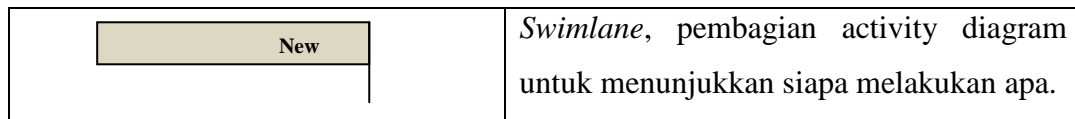
hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm m dan selanjutnya mempresentasikan *instance* majemuk dari definisi *actor* yang sama.

3. Activity Diagram.

Activity Diagram digambarkan sebagai rangkaian aliran dari aktifitas dan digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya. Diagram ini sangat mirip dengan flowchart karena memodelkan workflow dari suatu aktifitas ke aktifitas lainnya. Penggambaran activity diagram pada awal pemodelan proses dapat membantu memahami keseluruhan proses. Activity diagram juga digunakan untuk menggambarkan interaksi antara beberapa use case.

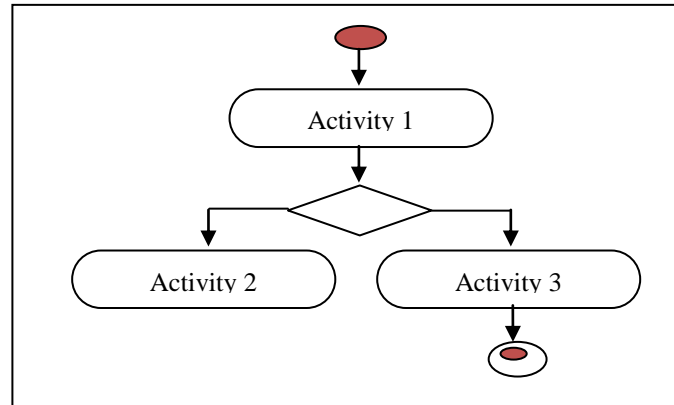
Tabel II.2. Simbol – simbol yang digunakan dalam Activity Diagram

Gambar	Keterangan
	<i>Start Point</i> , diletakkan pada pojok kiri atas dan merupakan awal aktifitas
	<i>End Point</i> , akhir aktifitas.
	<i>Activites</i> , menggambarkan suatu proses/kegiatan bisnis.
	<i>Fork</i> (percabangan), digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel menjadi satu.
	<i>Join</i> (penggabungan) atau <i>Rake</i> , digunakan untuk menunjukkan adanya dekomposisi.
	<i>Decision Points</i> , menggambarkan pilihan untuk pengambilan keputusan, <i>true</i> atau <i>false</i> .



Sumber : (Windu Gata, 2013, 6).

Berikut contoh bentuk *Activity diagram*:

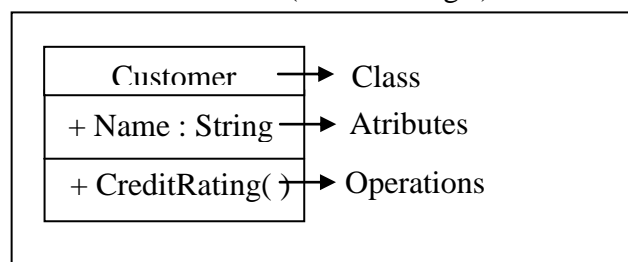


Gambar II.17. Tampilan Gambar Contoh Activity Diagram.

Sumber : (Fowler Martin, 2004)

4. Class Diagram.

Class adalah sebuah spesifikasi yang akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/property) suatu sistem dan menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi).



Gambar II.18. Tampilan Gambar Class.

Sumber : (Fowler Martin, 2004).

Class memiliki tiga area pokok, diantaranya sebagai berikut :

- a. Nama (Class Name)
- b. Atributes
- c. Metode (Operations).

Pada UML, class digambarkan dengan segi empat yang dibagi beberapa bagian. Bagian atas merupakan nama dari class. Bagian yang tengah merupakan struktur dari class (atribut). Dan yang bagian bawah merupakan sifat dari class (metode/operations).

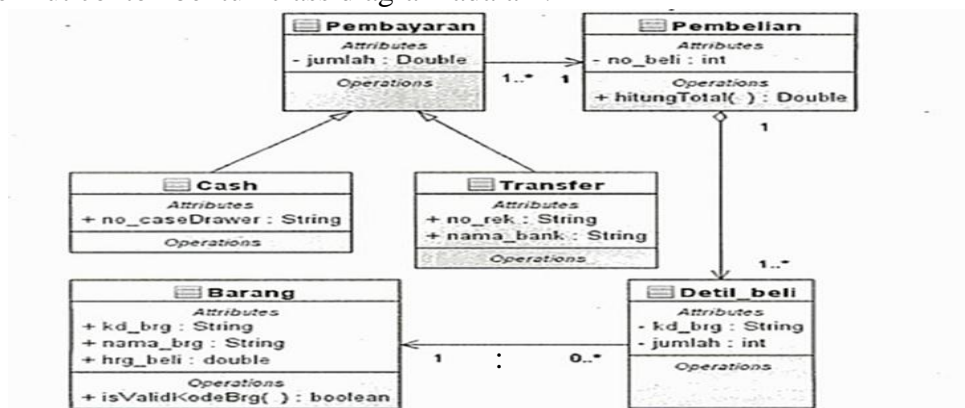
Dan beberapa hubungan antar class, diantaranya disimbolkan dengan sebagai berikut :

Tabel II.3. Hubungan antar class diagram.

Multiplicity	Penjelasan
1	Satu dan hanya satu
0..*	Boleh tidak ada atau 1 atau lebih
1..*	1 atau lebih
0..1	Boleh tidak ada, maksimal 1
n..n	Batasan antara. Contoh 2..4 mempunyai arti minimal 2 dan maksimal 4.

Sumber : (Windu Gata, 2013, 9).

Berikut contoh bentuk class diagram adalah :



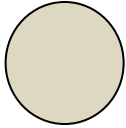
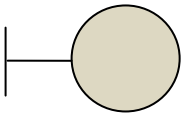
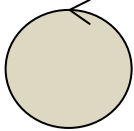


Gambar II.19. Contoh Class Diagram


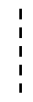
(Sumber : Miftkhul Huda & Bunafit Komputer, 2010, 143).

5. Sequence Diagram.

Menurut Adi Nugroho (2010 : 42-43) *Sequence Diagram* memperlihatkan interaksi sebagai diagram dua matra (dimensi). Matra vertikal adalah sumbu waktu; waktu bertambah dari atas ke bawah. Matra horizontal memperlihatkan peran pengklasifikasi yang merepresentasikan objek-objek mandiri yang terlibat dalam kolaborasi. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem.

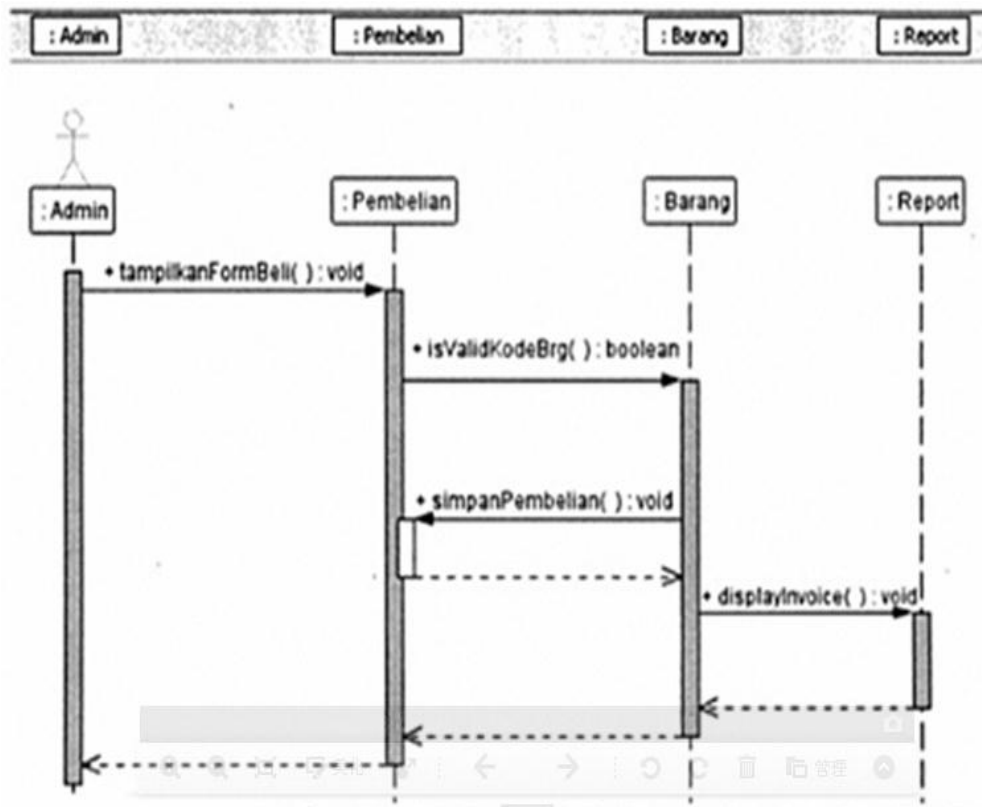
Tabel II.4. Simbol – simbol yang digunakan dalam *Sequence Diagram*

Gambar	Keterangan
	<i>Entity Class</i> , merupakan bagian dari sistem yang berisi kumpulan kelas berupa entitas – entitas yang membentuk gambaran awal sistem yang menjadi landasan untuk menyusun basis data.
	<i>Boundary Class</i> , berisi kumpulan kelas yang menjadi interface atau interaksi antara satu atau lebih aktor dengan sistem, seperti tampilan formentry dan form cetak.
	<i>Control Class</i> , suatu objek yang berisi logika aplikasi yang tidak memiliki tanggung jawab kepada entitas, contohnya adalah kalkulasi dan aturan bisnis yang melibatkan berbagai objek, <i>control objek</i> mengkoordinir pesan antara <i>boundary</i> dengan entitas
	<i>Message</i> , simbol mengirim pesan antara <i>class</i> .
	<i>Recursive</i> , menggambarkan pengiriman pesan yang dikirim untuk dirinya sendiri.

	<p><i>Activation</i>, <i>activation</i> mewakili sebuah eksekusi operasi dari objek, panjang kotak ini berbanding lurus dengan durasi aktivasi sebuah operasi.</p>
	<p><i>Lifeline</i>, garis titik – titik yang terhubung dengan objek, sepanjang <i>lifeline</i> terdapat <i>activation</i>.</p>

(Sumber : Windu Gata, 2013, 7).

Berikut contoh Sequence Diagram :



Gambar II.20. Contoh Sequence Diagram

(Sumber : Miftkhul Huda & Bunafit Komputer, 2010, 143).