

BAB II

TINJAUAN PUSTAKA

II.1. Sistem

Menurut (Anastasia Diana & Lilis Setiawati; 2011:3) Sistem merupakan serangkaian bagian yang saling tergantung dan bekerja sama untuk mencapai tujuan tertentu. Suatu sistem pasti tersusun dari sub-sub sistem yang lebih kecil yang saling tergantung dan bekerja sama untuk mencapai tujuan.

II.2. Sistem Informasi Geografis

Dengan melihat kata-kata penyusun nama Sistem Informasi Geografis, maka nama tersebut dapat dijabarkan sebagai berikut :

1. Sistem

Istilah ini digunakan untuk mewakili pendekatan sistem yang digunakan dalam SIG, dengan lingkungan yang kompleks dan komponen yang terpisah-pisah, sistem digunakan untuk mempermudah pemahaman dan penanganan yang terintegrasi. Teknologi komputer sangat dibutuhkan untuk pendekatan ini jadi hampir semua sistem informasinya berdasarkan pada komputer. (Eddy Prahasta;2007:5)

2. Informasi

Informasi berasal dari pengolahan sejumlah data. Dalam SIG informasi memiliki volume terbesar. Setiap objek geografi memiliki pengaturan data tersendiri karena tidak sepenuhnya data yang ada dapat terwakili dalam

peta. Jadi, semua data harus diasosiasikan dengan objek spasial yang dapat membuat peta menjadi *intelligent*. Ketika data tersebut diasosiasikan dengan permukaan geografi yang representatif, data tersebut mampu memberikan informasi dengan hanya mengklik mouse pada objek. (Eddy Prahasta;2007:5)

3. Geografis

Istilah ini digunakan karena Sistem Informasi Geografis dibangun secara berdasarkan pada geografi atau spasial. Objek ini mengarah pada spesifikasi lokasi dalam suatu daerah. Objek bisa berupa fisik, budaya atau ekonomi alamiah. Penampakan tersebut ditampilkan pada suatu peta untuk memberikan gambaran yang representative dari spasial suatu objek sesuai dengan kenyataannya di bumi. Simbol, warna dan gaya garis digunakan untuk mewakili setiap spasial yang berbeda pada peta dua dimensional. Saat ini, teknologi komputer telah mampu membantu proses pemetaan melalui pengembangan dari *automated cartography* (pembuatan peta) dan *Computer Aided Design (CAD)*. (Ruslan Nuryadin; 2005:19).

Sistem Informasi Geografis adalah sistem basis data dengan kemampuan-kemampuan khusus untuk data yang tereferensi secara spasial atau koordinat geografis berikut sekumpulan operasi-operasi yang mengelola data tersebut. Sistem Informasi Geografis dibutuhkan karena untuk data spasial penanganannya sangat sulit terutama karena peta dan data statistik cepat kadaluarsa hingga tidak ada

pelayanan penyediaan data dan informasi yang diberikan menjadi tidak akurat. (Eddy Prahasta,2007: 23).

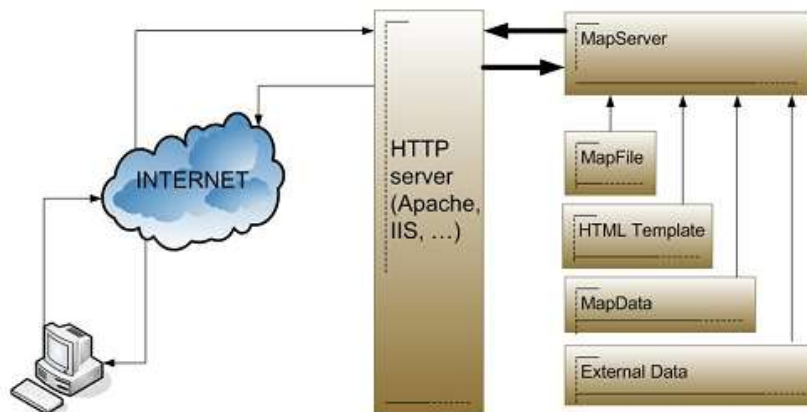
Dengan demikian, SIG diharapkan mampu memberikan kemudahan-kemudahan seperti :

- a. Penanganan data geospasial menjadi lebih baik dalam format baku.
- b. Revisi dan pemutakhiran data menjadi lebih mudah.
- c. Data geospasial dan informasi menjadi lebih mudah dicari, dianalisis dan direpresentasikan.
- d. Menjadi produk yang mempunyai nilai tambah.
- e. Kemampuan menukar data geospasial.
- f. Penghematan waktu dan biaya.
- g. Keputusan yang diambil menjadi lebih baik.

II.3. MapServer

MapServer merupakan aplikasi *freeware* dan *open source* yang memungkinkan kita menampilkan data spasial (peta) di web. Aplikasi ini pertama kali dikembangkan di Universitas *Minesotta*, Amerika Serikat untuk projek *ForNet* (sebuah projek manajemen sumber daya alam) yang disponsori oleh NASA. Pengembangan *MapServer* menggunakan berbagai aplikasi *open source* atau *freeware* seperti *Shapelib* (<http://shapelib.maptools.org>) untuk baca/tulis format data *Shapefile*, *FreeType* untuk membuat karakter, *GDAL/OGR* untuk baca/tulis berbagai format data vektor maupun raster, dan *Proj4* untuk menangani berbagai proyeksi peta. Pada dasarnya, *MapServer* adalah sebuah program CGI (*Common*

Gateway Interface). Program tersebut akan dieksekusi di web server, dan berdasarkan beberapa parameter tertentu (terutama konfigurasi dalam bentuk file *.MAP), akan menghasilkan data yang kemudian akan dikirim ke web browser, baik dalam bentuk gambar peta ataupun bentuk lain.



Gambar II.1 : Arsitektur aplikasi *web-based* GIS dengan *MapServer*

(Sumber : Eddy Prahasta ; 2006:46)

Berdasarkan arsitektur *MapServer* sebagai program CGI pada Gambar II.1, dapat kita lihat bahwa *browser (client)* akan mengirimkan *request* ke *web server* dalam bentuk permintaan yang berhubungan dengan spasial. Kemudian oleh *web server* permintaan ini dikirimkan ke *server* aplikasi (yang telah dibangun dengan menggunakan *script* yang tersedia) dan *MapServer* (program CGI). Setelah itu *MapServer* akan membaca *mapfile*, data peta, dan data eksternal untuk membentuk sebuah gambar yang sesuai dengan permintaan. Setelah gambar di-render, *file image* yang bersangkutan akan dikirim ke *web server* hingga akhirnya sampai ke *client* sesuai dengan format tampilan yang telah diatur dalam *template*.

Menurut (Ruslan Nuryadin ; 2005 : 25) dalam buku Panduan Menggunakan *MapServer* yang ditulisnya diberitahukan beberapa fitur *MapServer*, yaitu:

- a. Menampilkan data spasial dalam format vektor seperti *Shapefile*, Arc View, PHP *Mapscript* dan berbagai format data vektor lain dengan menggunakan *library* OGR.
- b. Menampilkan data spasial dalam format raster seperti TIFF/GeoTIFF, EPPL7 dan berbagai format data raster lain dengan menggunakan *library* GDAL.
- c. Menggunakan *quadtree* dalam *indexing* data spasial, sehingga operasi-operasi spasial dapat dilakukan dengan cepat.
- d. Dapat dikembangkan (*customizable*) dengan tampilan keluaran yang dapat diatur menggunakan file-file template.
- e. Dapat melakukan seleksi objek berdasar nilai, atau berdasar titik area, atau berdasar sebuah objek spasial tertentu.
- f. Mendukung rendering karakter berupa font TrueType.
- g. Mendukung penggunaan data raster maupun vektor yang dibagi-bagi (*tiled*) menjadi sub bagian yang lebih kecil sehingga proses untuk mengambil dan menampilkan gambar dapat dipercepat.
- h. Dapat menggambarkan elemen peta (skala grafis, peta indeks dan legenda peta) secara otomatis.
- i. Dapat menggambarkan peta tematik yang dibangun menggunakan ekspresi logika maupun ekspresi reguler.
- j. Konfigurasi dapat diatur secara langsung melalui parameter yang ditentukan pada URL.
- k. Dapat menangani berbagai sistem proyeksi secara langsung.

1. Saat ini, selain dapat mengakses *MapServer* sebagai program CGI, kita dapat mengakses *MapServer* sebagai modul *MapScript*, melalui berbagai bahasa skrip seperti *PHP*, *Perl*, *Python* atau *Java*. Akses fungsi-fungsi *MapServer* melalui skrip akan memudahkan dalam pengembangan karena pengembang dapat memilih bahasa yang familiar.

II.4. Website

Website atau *world wide web* sering disingkat sebagai *www* atau *web*, yaitu sebuah sistem dimana informasi dalam bentuk teks, gambar, suara, dan lain-lain direpresentasikan dalam bentuk *hypertext* dan dapat diakses oleh perangkat lunak yang disebut *browser*. (Angga Wibowo ; 2007 : 2)

II.4.1. Internet

Internet (*International Networking*) adalah suatu kumpulan jaringan komputer dari berbagai tipe, yang saling berkomunikasi dengan menggunakan suatu standar komunikasi. Saat ini ada jutaan sistem komputer dengan puluhan juta pengguna di seluruh dunia telah bergabung dengan *internet*. (Angga Wibowo ; 2007 : 3)

II.4.2. Web browser

Web browser adalah suatu perangkat lunak yang digunakan untuk menampilkan halaman-halaman *website* yang berada di *internet*. Terdapat beberapa macam *web browser* yang dapat kita pakai untuk menampilkan halaman-halaman *website*. Ada 3 jenis web browser yang sering dipakai, antara lain : *Internet Explorer*, *Mozilla* , *Netscape*. (Angga Wibowo ; 2007 : 6)

II.4.3. Web Server

Web Server merupakan sebuah perangkat lunak server yang berfungsi menerima permintaan *HTTP* atau *HTTPS* dari klien yang dikenal dengan *browser web* dan mengirimkan kembali hasilnya dalam bentuk halaman-halaman *web* yang umumnya berbentuk dokumen *HTML*. *Server web* yang terkenal diantaranya adalah *Apache* dan *IIS (Microsoft Internet Information Service)*. *Apache* merupakan *server web antar-platform*, sedangkan *IIS* hanya dapat beroperasi di sistem operasi Windows. (Angga Wibowo ; 2007 : 5)

II.5. PHP

PHP singkatan dari *hypertext preprocessor* yang digunakan sebagai bahasa *script server-side* dalam pengembangan *web* yang disisipkan pada dokumen HTML. Penggunaan PHP memungkinkan *web* dapat dibuat dinamis sehingga *maintanance* situs *web* tersebut lebih mudah dan efisien. PHP merupakan software *open-source* yang dis ebarakan dan dilisensikan secara gratis dan dapat didownload dari situs resminya yaitu : <http://www.php.net>. (Angga Wibowo ; 2007 : 2)

II.6. Unified Modeling Language (UML)

Menurut (Adi Nugroho ; 2009 : 10) UML Adalah perkakas untuk analisis dan perancangan yang sesungguhnya digunakan untuk peyederhanaan permasalahan. UML merupakan metodologi kolaborasi antara metoda-metoda *Booch*, *OMT (Object Modeling Technique)*,serta *OOSE (Object Oriented Software Engineering)* dan beberapa metoda lainnya, merupakan metodologi yang paling

sering di gunakan pada saat ini untuk mengadaptasi maraknya penggunaan bahasa pemrograman berorientasi objek.

Menurut (prabowo Pudjo Widodo & Herlawati;2011:6) UML diaplikasikan untuk maksud tertentu, biasanya antara lain :

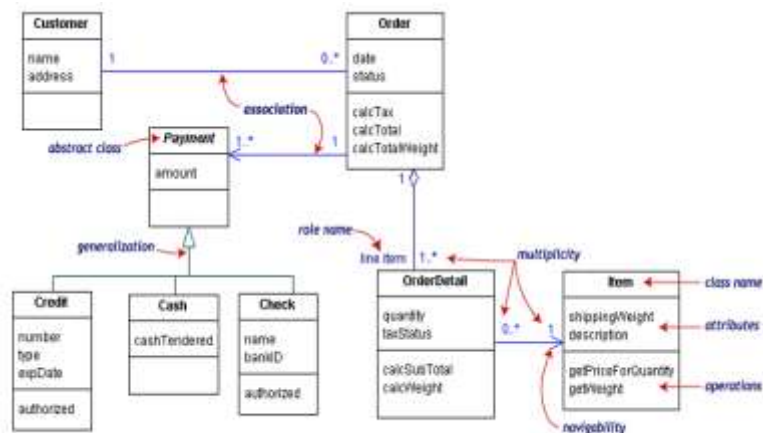
1. Merancang perangkat Lunak.
2. Sarana Komunikasi antaraperangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasikan sistem yang ada, proses-proses dan organisasinya.

Blok pembangunan utama UML adalah diagram. Beberapa diagram ada yang rinci (jenis *timing diagram*) dan lainnya ada yang bersifat umum (misalnya diagram kelas). Para pengembang sistem berorientasi objek menggunakan bahasa model untuk menggambarkan, membangun dan mendokumentasikan sistem yang mereka rancang. UML memungkinkan para anggota team untuk bekerja sama dengan bahasa model yang sama dengan mengaplikasikan beragam sistem. Intinya UML merupakan alat komunikasi yang konsisten dalam mendukung para pengembang sistem saat ini.

II.6.1 Diagram-Diagram UML

Beberapa literatur menyebutkan bahwa UML menyediakan sembilan jenis diagram, model-model ini dapat dikelompokan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain :

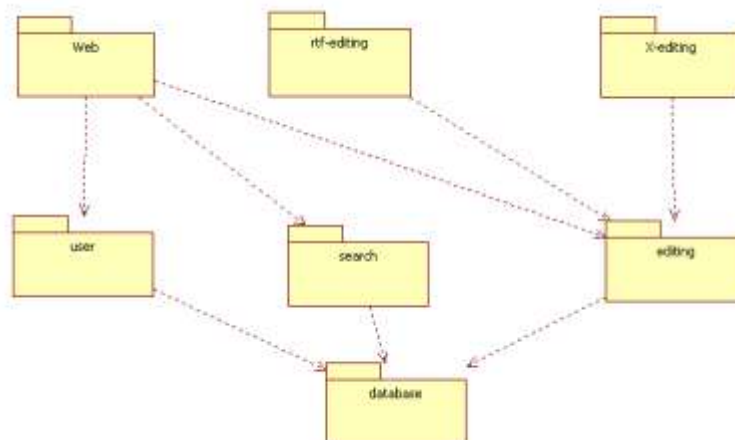
1. Diagram Kelas : Bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, sering pula diagram kelas memuat kelas-kelas aktif.



Gambar II.2 : Diagram Kelas

(Sumber : Prabowo Pudjo Widodo & Herlawati ; 2011:30)

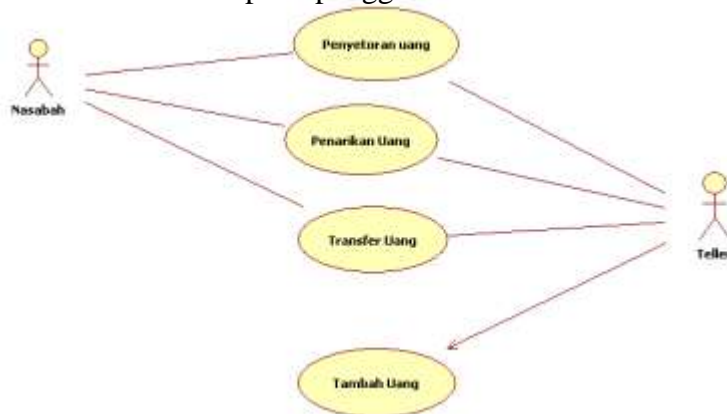
2. Diagram Paket (*Package Diagram*) : Bersifat statis. Diagram ini memperlihatkan kumpulan kelas-kelas, merupakan bagian dari diagram komponen.



Gambar II.3 : Diagram Paket

(Sumber : Prabowo Pudjo Widodo & Herlawati ; 2011:32)

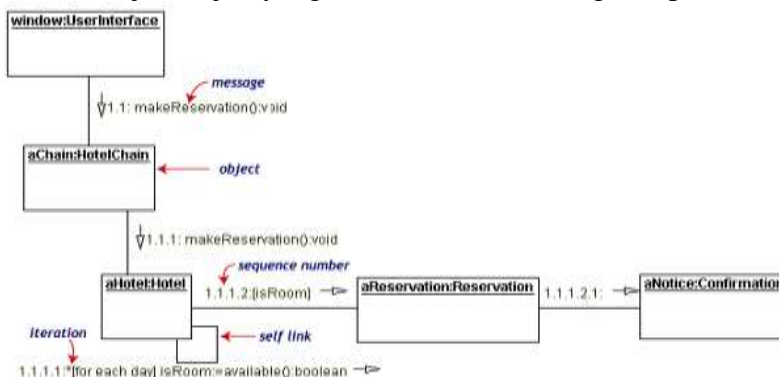
3. Diagram *Use-Case* : Bersifat statis. Diagram ini memperlihatkan himpunan *use-case* dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.



Gambar II.4 : Diagram Use Case

(Sumber : Prabowo Pudjo Widodo & Herlawati ; 2011:25)

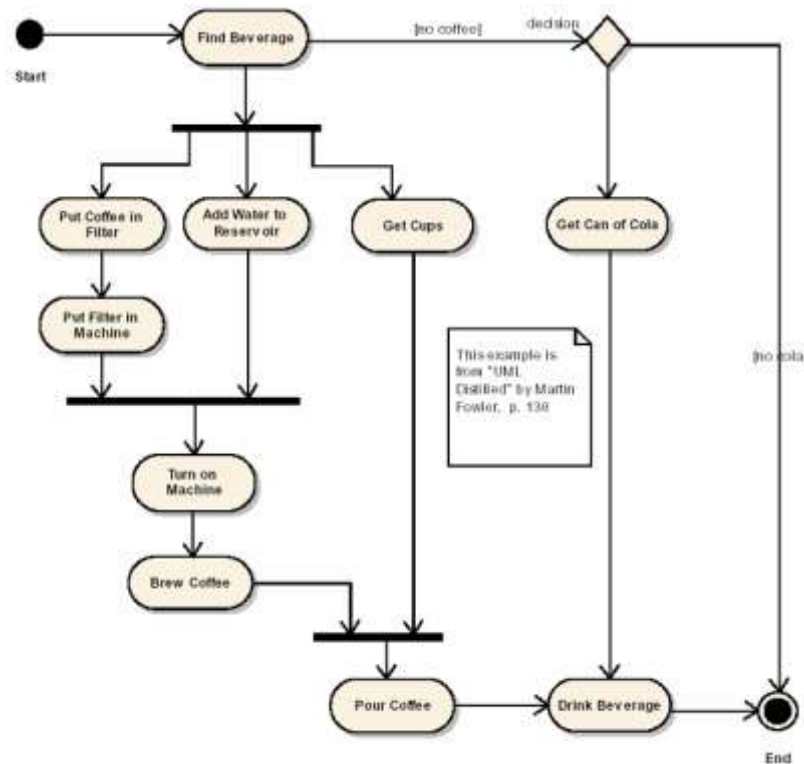
4. Diagram Interaksi dan *Sequence* (Urutan) : Bersifat dinamis. Diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan dalam suatu waktu tertentu.
5. Diagram Komunikasi (*Communication Diagram*) : Bersifat dinamis. Diagram sebagai pengganti diagram kolaborasi UML 1.4 yang menekankan organisasi struktural dari objek-objek yang menerima serta mengirim pesan.



Gambar II.5 : Diagram Komunikasi

(Sumber : Prabowo Pudjo Widodo & Herlawati ; 2011:41)

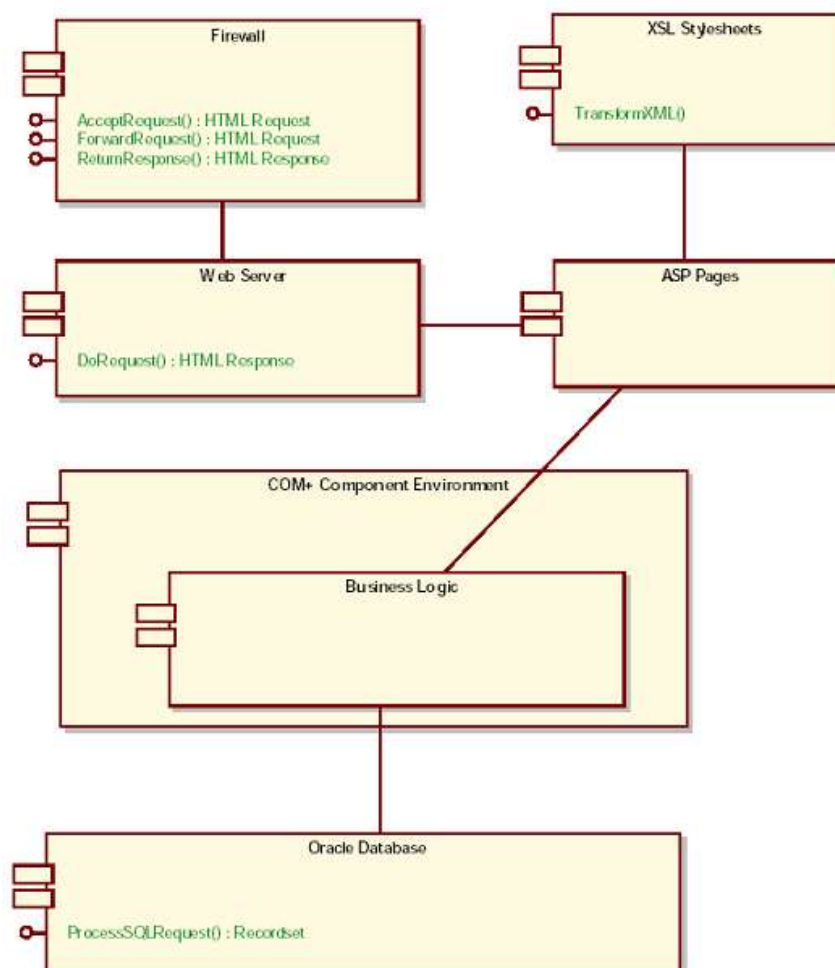
6. Diagram *Statechart* (*Statechart Diagram*) : Bersifat dinamis. Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (*state*), transisi, kejadian serta aktifitas. Diagram ini terutama penting untuk memperlihatkan sifat dinamis dari antarmuka (*interface*), kelas, kolaborasi dan terutama penting pada pemodelan sistem-sistem yang reaktif.
7. Diagram Aktivitas (*Activity Diagram*). Bersifat dinamis. Diagram aktivitas adalah tipe khusus dari diagram status yang memperlihatkan aliran dari suatu aktivitas ke aktivitas lainnya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi suatu sistem dan memberi tekanan pada aliran kendali antar objek.



Gambar II.6 : Diagram Aktivitas

(Sumber : Prabowo Pudjo Widodo & Herlawati ; 2011:67)

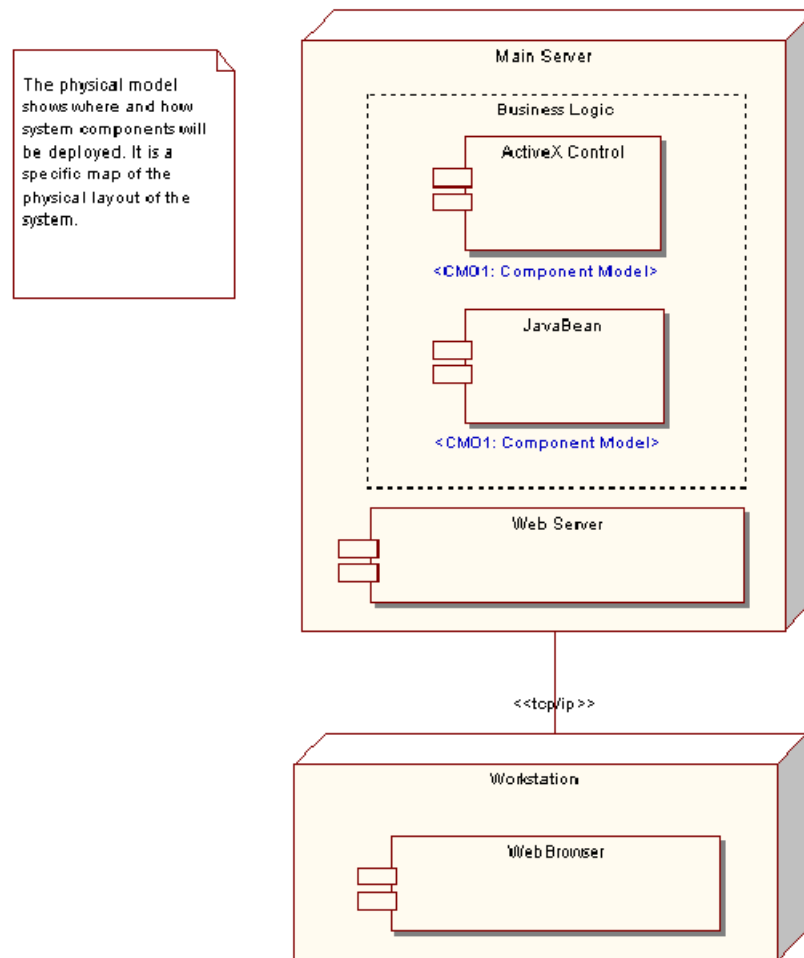
8. Diagram Komponen (*Component Diagram*) : Bersifat statis. Diagram komponen ini memperlihatkan organisasi serta ketergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya. Diagram ini berhubungan dengan diagram kelas dimana komponen secara tipikal dipetakan kedalam satu atau lebih kelas-kelas, antarmuka-antarmuka serta kolaborasi-kolaborasi.



Gambar II.7 : Diagram Komponen

(Sumber : Prabowo Pudjo Widodo & Herlawati ; 2011:88)

9. Diagram *Deployment (Deployment Diagram)* : Bersifat statis. Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*run-time*). Memuat simpul-simpul beserta komponen-komponen.



Gambar II.8 : Diagram *Deployment*

(Sumber : Prabowo Pudjo Widodo & Herlawati ; 2011:72)

Diagram ini sangat berguna saat aplikasi kita berlaku sebagai aplikasi yang dijalankan pada banyak mesin (*distributed computing*), (prabowo Pudjo Widodo & Herlawati ; 2011: 6).

II.7. Database

Menurut (Abdul Kadir ; 2009:10) terdapat butir-butir yang bisa diambil dari definisi database yaitu :

1. Sebuah database menghimpun data yang terkait atau data yang saling berhubungan.
2. Kumpulan data tersebut terorganisasi.
3. Bisa melibatkan lebih dari satu organisasi.

Butir ketiga mengisyaratkan bahwa sebuah database bisa digunakan oleh lebih dari satu organisasi. Sebuah database mencatat berbagai data yang diperlukan oleh suatu organisasi, rekaman-rekaman data tersebut pada suatu saat akan diambil dan melalui suatu pemrosesan akan diperoleh suatu informasi yang dikehendaki oleh pengguna.

Database berbeda dengan sistem pemrosesan berbasis berkas. Sistem pemrosesan berbasis berkas adalah suatu model penyimpanan data yang mendasarkan pada penyimpanan data dalam bentuk *file* (berkas), yang memiliki banyak kelemahan dibanding database.sistem seperti ini banyak dipakai dimasa lalu, salah satu perangkat lunak yang biasa dipakai untuk mengimplementasikannya adalah COBOL.

II.7.1. Pemodelan Data

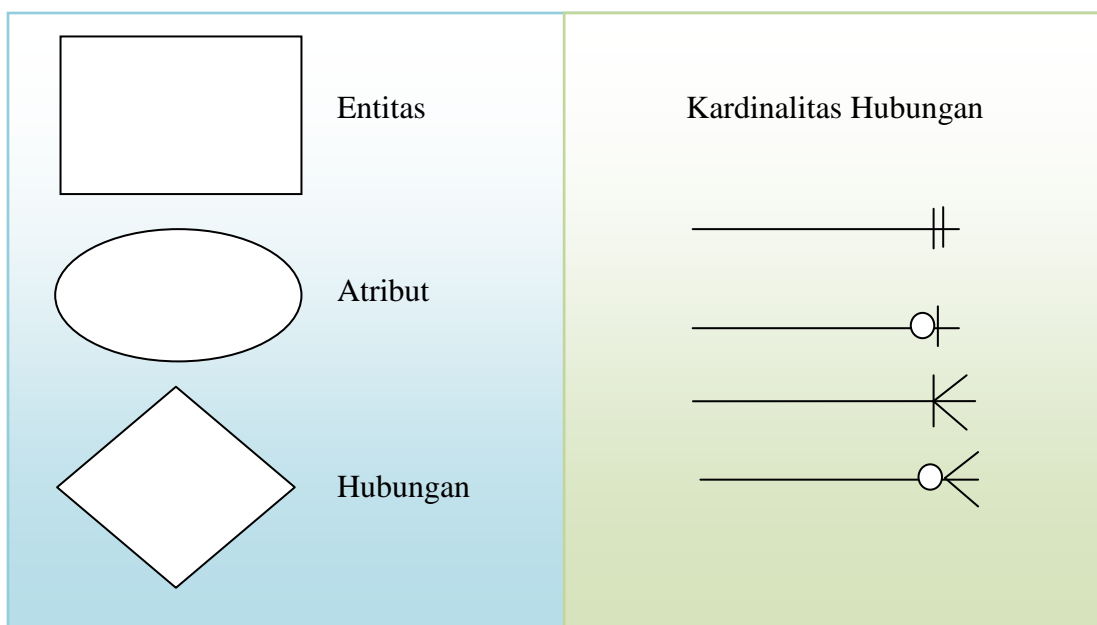
Pada Perancangan konseptual diperlukan suatu pendekatan yang digunakan untuk menggambarkan hubungan antar data. Hubungan tersebut dapat dinyatakan dalam bentuk E-R, karena model E-R adalah dasar penting dalam merancang

database maka akan dijelaskan tentang gambaran tentang model E-R, penjelasan mengenai komponen-komponen yang menyusun model E-R, hingga cara penyusunan model E-R.

II.7.2. Model E-R

Model E-R adalah suatu model yang digunakan untuk menggambarkan data dalam bentuk entitas, atribut dan hubungan antar entitas. Huruf E sendiri menyatakan entitas dan R menyatakan hubungan (dari kata *Relationship*). Model ini dinyatakan dalam bentuk diagram, itulah sebabnya model E-R sering disebut sebagai diagram E-R.

Model E-R melibatkan sejumlah notasi, beberapa notasi dasar dalam model E-R ditunjukkan pada gambar II.9, notasi-notasi tersebut diberikan hanya untuk memberikan suatu pengetahuan dasar.



Gambar II.9 : Sejumlah notasi pada model E-R

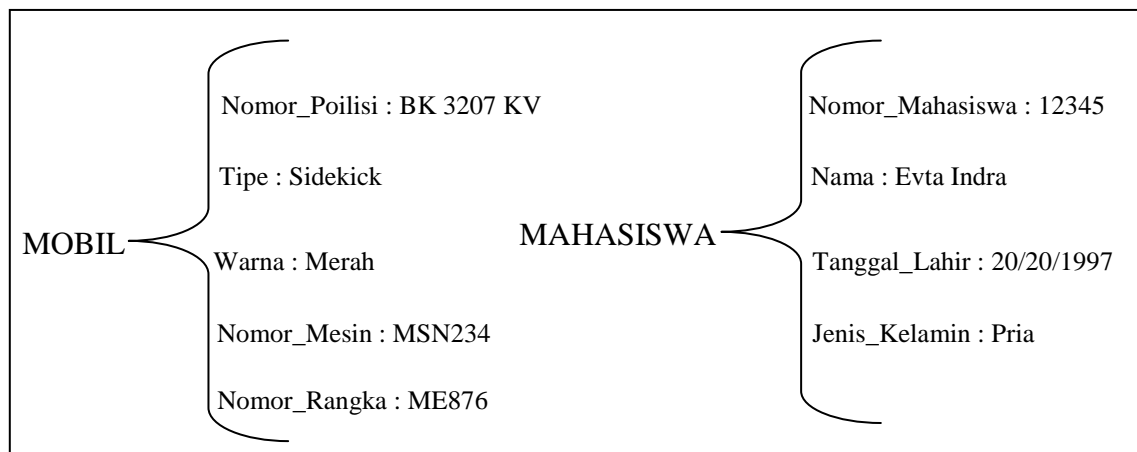
(Sumber : Abdul Kadir ; 2009:31)

1. Entitas

Yang dimaksud dengan entitas adalah sesuatu dalam dunia nyata yang keberadaannya tidak bergantung pada yang lain. Sebagai contoh, setiap pegawai dalam sebuah organisasi adalah sebuah entitas. Entitas dapat berupa suatu yang nyata ataupun abstrak (berupa suatu konsep). Secara lebih rinci dijelaskan bahwa entitas dapat berupa seseorang, sebuah tempat, sebuah objek, sebuah kejadian atau suatu konsep.

2. Atribut

Setiap entitas dinyatakan dalam sejumlah atribut. Atribut adalah properti atau karakteristik yang terdapat pada setiap entitas. Sebagai contoh, pada gambar II.10, terdapat entitas MOBIL yang mengandung atribut Nomor_Polisi, Tipe, Warna, Nomor_Rangka dan Nomor_Mesin. Selain itu terdapat entitas MAHASISWA yang mengandung atribut Nomor_Mahasiswa, Nama, Tanggal_Lahir, dan Jenis_Kelamin.

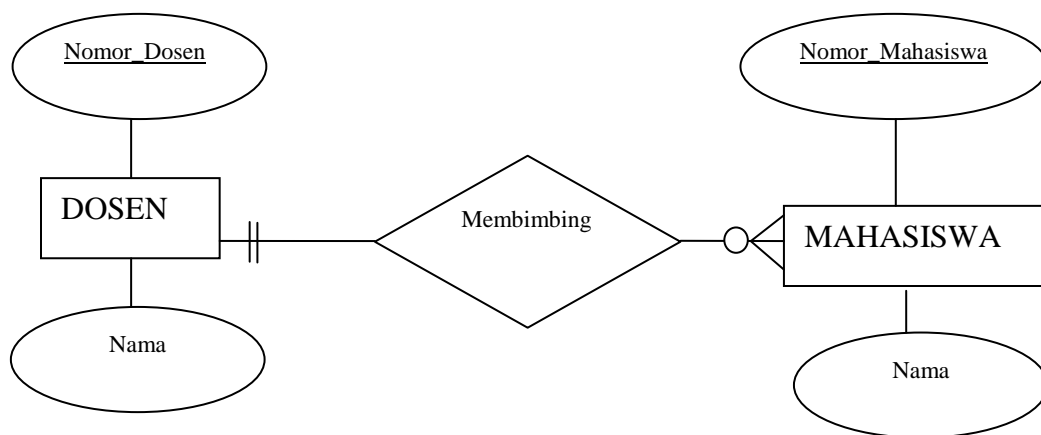


Gambar II.10 : Contoh Entitas dan Atribut

(Sumber : Abdul Kadir ; 2009:32)

3. Hubungan (*Relationship*)

Hubungan (*Relationship*) menyatakan ketertarikan antara beberapa tipe entitas. Sebagai contoh , tipe entitas MAHASISWA dan DOSEN mempunyai hubungan yang mencerminkan bahwa seorang mahasiswa memiliki dosen pembimbing akademis. Gambar II.11 menunjukkan hubungan tersebut.



Gambar II.11 : Contoh Hubungan antara tipe entitas

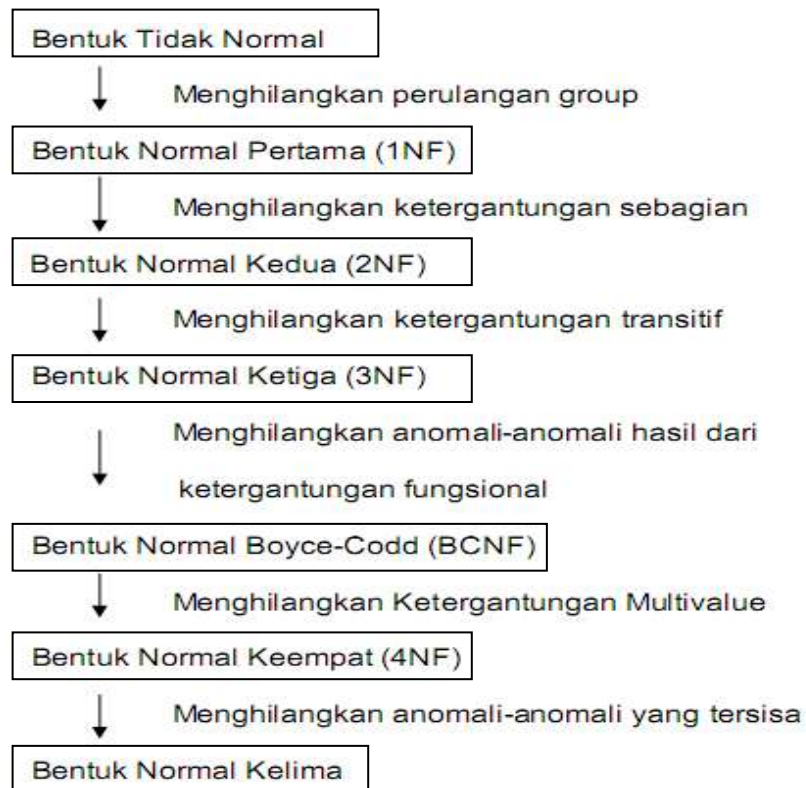
(Sumber : Abdul Kadir ; 2009:45)

II.7.3. Normalisasi

Normalisasi adalah proses yang digunakan untuk menentukan pengelompokan atribut-atribut dalam sebuah relasi sehingga diperoleh relasi yang berstruktur baik. Dalam hal ini yang dimaksud dengan relasi yang berstruktur baik adalah relasi yang memenuhi dua kondisi berikut

1. Mengandung redundansi sedikit mungkin, dan
2. Memungkinkan baris-baris dalam relasi disisipkan, dimodifikasi dan dihapus tanpa menimbulkan kesalahan atau ketidakkonsistenan.

Normalisasi sendiri dilakukan melalui sejumlah langkah. Setiap langkah berhubungan dengan bentuk normal (*normal form*) tertentu. Dalam hal ini yang disebut bentuk normal adalah suatu keadaan relasi yang dihasilkan oleh penerapan aturan-aturan sederhana yang berhubungan dengan dependensi fungsional terhadap relasi tersebut. Gambar II.12 berikut ini akan memperlihatkan hubungan keenam bentuk normal tersebut.



Gambar II.12 : Langkah-langkah dalam normalisasi

(Sumber : Abdul Kadir ; 2009:118)