

BAB II

TINJAUAN PUSTAKA

II.1. Pengertian Sistem dan Analisa Sistem

II.1.1. Sistem

Sistem adalah sekumpulan komponen yang saling berhubungan, bekerja sama untuk mencapai suatu tujuan dengan menerima *input* serta menghasilkan *output* dalam proses transformasi yang teratur (Budi Sutedjo Darma Oetomo:2006,168)

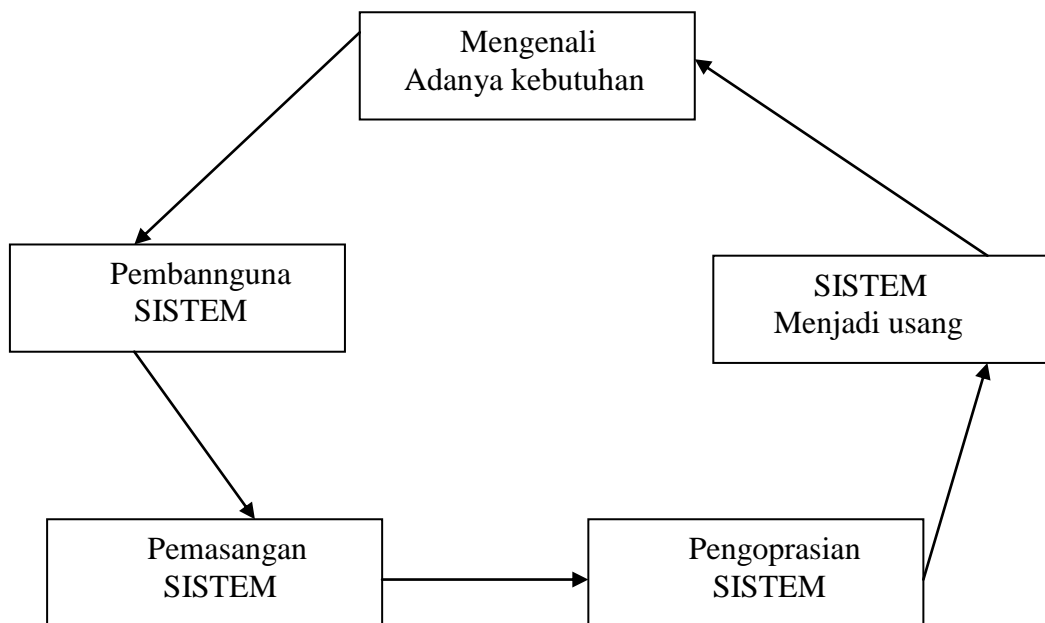
Suatu sistem dapat dirumuskan sebagai setiap kumpulan komponen atau atau subsistem untuk dirancang untuk mencapai suatu tujuan dengan pendekatan sistem, kita berhubungan dengan komponen perseorangan dan kita lebih menekankan perannya dalam sistem, dari pada perannya sebagai suatu keseluruhan individu. Dengan pendekatan sistem untuk menggambarkan kenyataan, dapat diberikan suatu keuntungan yang besar kepada pemakai. Keberhasilan komponen-komponen yang dipertimbangkan secara bersama sebagai suatu sistem mungkin lebih besar dari pada jumlah keberhasilan setiap komponen yang dipertimbangkan secara terpisah. Contohnya, seorang dokter ahli THT yang memahami dengan baik struktur sistem pernafasan dan proses sistem tersebut. Dokter tersebut akan mudah akan mengidentifikasi penyakit pasiennya yang mengalami sulit bernapas. Dia akan dapat dengan mudah mendeteksi apakah masalah kesulitan bernapas pasiennya tersebut disebabkan

oleh kesalahan srtuktur sistem pernafasannya ataukah pada proses sistem pernafasannya.(Tata Sutabri:2005,8-9).

II.1.2. Siklus Hidup Sistem

Siklus hidup sistem (*system life cycle*) adalah proses evolusioner yang diikuti dalam menerangkan sistem atau subsistem informasi berbasis komputer. Siklus hidup sistem terdiri dari serangkaian tugas yang erat mengikuti langkah-langkah pendekatan sistem karna tugas-tugas tersebut mengikuti pola yang teratur dan dilakukan secara *top down*. siklus hidup sistem sering disebut sebagai pendekatan air terjun (*waterfall approach*) bagi pembangunan dan pengembangan sistem.

Adapun gambar dari siklus hidup sistem seperti di bawah ini.



Gambar II.1 Daur Hidup Sistem

Sumber : Tata Sutabri ; 2005 : 15

Keterangan Gambar II.2

1. Mengenali adanya kebutuhan

Sebelum segala sesuatunya terjadi, timbul suatu kebutuhan atau problema yang harus dapat dikenali sebagai mana adanya. Kebutuhan dapat terjadi sebagai hasil perkembangan dari organisasi dan volume yang meningkat melebihi kapasitas dari sistem yang ada. Semua kebutuhan ini harus dapat didefinisikan dengan jelas. Tanpa adanya kejelasan dari kebutuhan yang ada, pembangunan sistem akan kehilangan arah dan efektifitasnya.

2. Pembangunan Sistem

Suatu proses atau seperangkat prosedur yang harus diikuti untuk menganalisis kebutuhan yang timbul dan pembangunan suatu sistem untuk dapat memenuhi kebutuhan tersebut.

3. Pemasangan Sistem

Setelah tahap pembangunan sistem selesai. Sistem kemudian akan dioperasikan. Pemasangan sistem merupakan tahap yang paling penting pula dalam daur hidup sistem. Peralihan dari tahap pembangunan menuju tahap operasional terjadi pemasangan sistem yang sebenarnya, yang merupakan langkah akhir dari suatu pembangunan sistem.

4. Pengoperasian Sistem

Program-program komputer dan prosedur-prosedur pengoperasian yang membentuk suatu sistem informasi semuanya bersifat statis, sedangkan organisasi ditunjang oleh sistem informasi tadi. Ia selalu mengalami perubahan-perubahan kegiatan bisnis, perubahan peraturan, dan

kebijaksanaan ataupun kemajuan teknologi. Untuk mengatasi perubahan-perubahan tersebut, sistem harus diperbaiki atau diperbarui.

5. Sistem Menjadi Usang

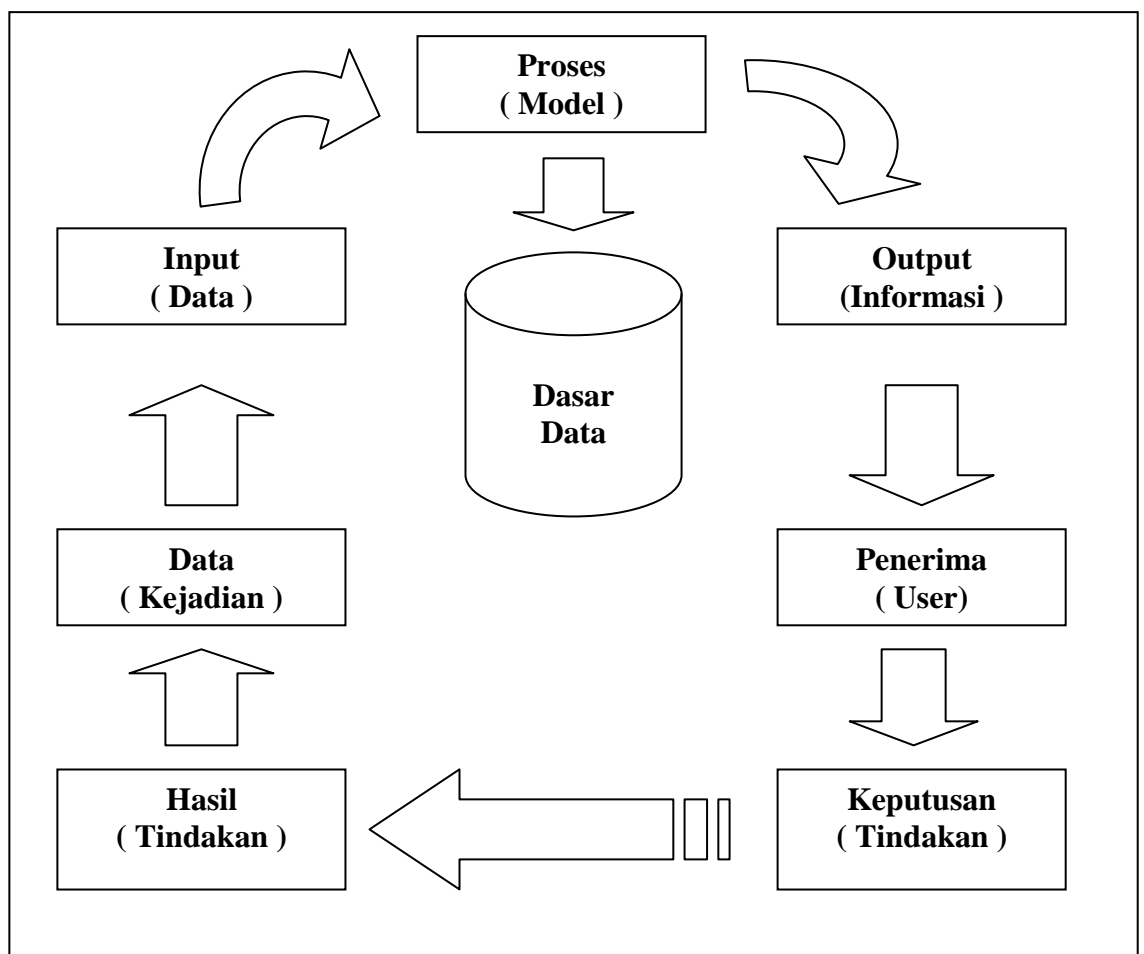
Kadang perubahan yang terjadi begitu drastis sehingga tidak dapat diatasi dengan melakukan perbaikan-perbaikan pada sistem yang berjalan. Tibalah saatnya secara ekonomis dan teknis sistem yang sudah tak layak lagi untuk dioperasikan dan sistem yang baru perlu dibangun untuk menggantikannya.

II.1.3. Pengertian Informasi

Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Informasi pada dasarnya merupakan fakta-fakta atau data-data yang telah mengalami proses yang disebut dengan proses transformasi data sehingga menjadi informasi. Data merupakan bahan baku yang belum diolah dan belum dapat dipergunakan. Untuk itu data harus diolah sedemikian rupa hingga menjadi informasi yang mempunyai arti tersendiri sesuai dengan kegunaan dari pada informasi tersebut.

Pada dasarnya informasi berguna untuk memberi tahu penerima informasi mengenai suatu permasalahan, agar si penerima informasi itu dapat mengetahui dan menguasai permasalahan yang dihadapi. Untuk memperoleh informasi yang bermanfaat bagi penerimanya, perlu untuk dijelaskan bagaimana siklus yang terjadi atau dibutuhkan dalam menghasilkan informasi.

Pertama – tama barang dimasukkan ke dalam model yang umumnya memiliki urutan proses tertentu dan pasti. Setelah diproses akan dihasilkan informasi tertentu yang bermanfaat bagi penerima (*Level Management*) sebagai dasar dalam membuat suatu keputusan atau melakukan tindakan tertentu. Dari keputusan atau tindakan tersebut akan menghasilkan atau diperoleh kejadian-kejadian tertentu yang akan digunakan kembali sebagai data yang nantinya akan dimasukkan ke dalam model (proses), begitu seterusnya. Dengan demikian akan membentuk suatu siklus informasi (*Information Cycles*) atau siklus pengolahan data (*Data Processing Cycles*), seperti gambar berikut:



Gambar II. 2 : Siklus Informasi

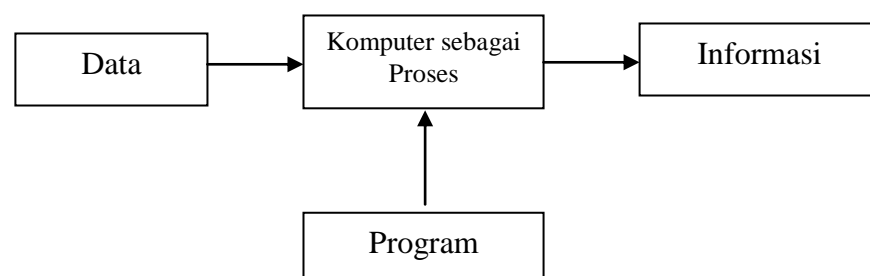
Sumber : Jogiyanto (2005 ; 9)

Hubungan data dengan informasi adalah Sangat erat sekali. Data diibaratkan sebagai bahan baku yang mengalami proses transformasi, sehingga keluarannya berubah menjadi barang jadi.

Agar informasi yang penulis sajikan lebih bermanfaat maka terlebih dahulu dibuat aliran informasi yang lebih jelas dan lengkap. Berkaitannya dengan penyedia informasi bagi manajemen dalam mengambil suatu keputusan, yang diperoleh harus berkualitas, maka kualitas dari informasi tergantung pada.

1. Akurat : akurat berarti bahwa informasi harus bebas dari kesalahan – kesalah dan tidak biasa (menyesatkan) dan jelas mencerminkan maksudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerimaan informasi kemungkinan banyak terjadi gangguan yang dapat merubah informasi atau merusak informasi tersebut.
 2. Relevansi : relevansi berarti bahwa informasi benar – benar berguna bagi suatu tindakan dan keputusan oleh seseorang
 3. Tepat waktu : tepat waktu berarti bahwa informasi yang datang pada penerimaan tidak boleh terlambat. Informasi yang sudah using tidak akan mempunyai nilai lagi. Karena informasi merupakan landasan dalam pengambilan keputusan. Dewasa ini mahalnya nilai informasi disebabkan harus cepatnya informasi itu di dapat,
- Untuk lebih jelasnya informasi merupakan hasil atau output dari proses informasi data.

Hal ini dapat dilihat seperti gambar II.3 dibawah ini:

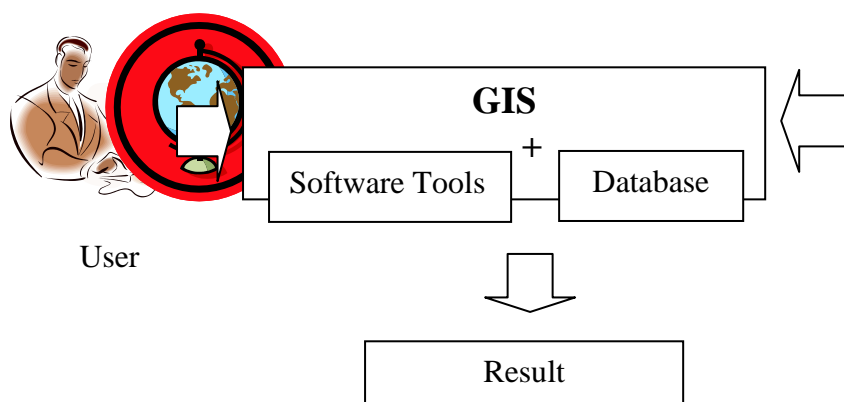


Gambar II. 3 : Transformasi Data Menjadi Informasi
Sumber : Abdul Kadir & Terra Ch. Triwahyuni (2005 ; 3)

II.2. Sistem Informasi Geografis

II.2.1. SIG dan Sistem Kerjanya

Penggunaan Sistem Informasi Geografis (SIG) meningkat tajam sejak tahun 1980-an. Peningkatan pemakaian sistem ini terjadi dikalangan pemerintah, militer, akademis, atau bisnis terutama di negara-negara maju. Perkembangan teknologi digital sangat besar peranannya dalam perkembangan penggunaan SIG dalam berbagai bidang. Hal ini dikarenakan teknologi SIG banyak mendasarkan pada teknologi digital ini sebagai alat analisis. Dapat kita lihat pada gambar II.5 dibawah ini:



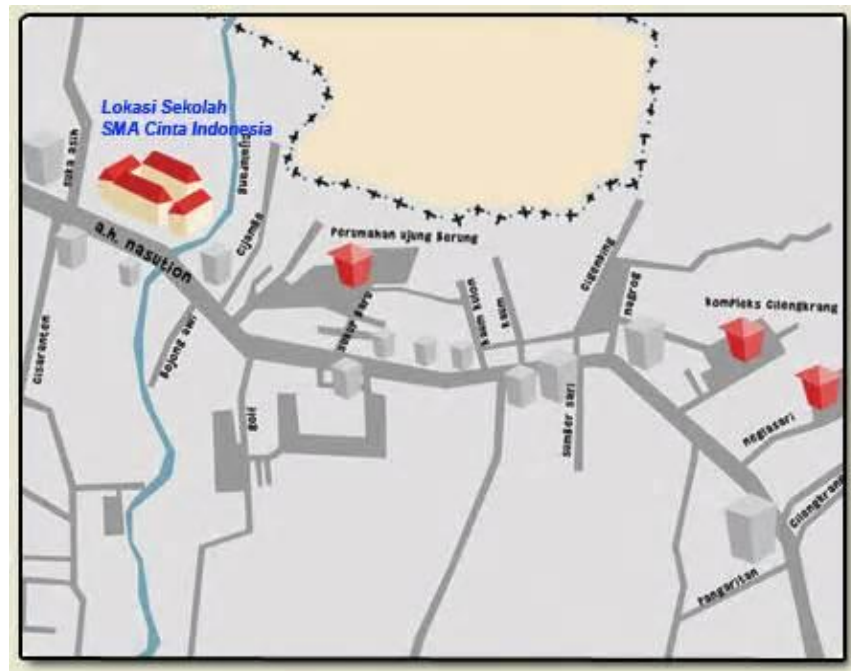
Gambar II. 4 : Pola Keterkaitan GIS
Sumber : Eko Budiyanto (2005 ; 1)

SIG merupakan sebuah sistem yang saling berangkaian satu dengan yang lain. BAKOSURTANAL menjabarkan SIG sebagai kumpulan yang terorganisir dari perangkat keras komputer, perangkat lunak, data geografi, dan personil yang didesain untuk memperoleh,

menyimpan, memperbaiki, memanipulasi, menganalisis dan menampilkan semua bentuk informasi yang bereferensi geografi.

Pengertian SIG dapat beragam tetapi mempunyai satu kesamaan, yaitu bahwa SIG adalah suatu sistem yang berkaitan dengan informasi geografis (Maguire, 1991 dalam Subaryono 2005). Dalam arti yang lebih sempit, Sig merupakan suatu sistem berbasis komputer yang digunakan untuk menyimpan dan menganalisis obyek-obyek dan fenomena-fenomena dengan lokasi geografis merupakan karakteristik yang penting untuk dianalisis.

Dengan demikian, basis analisis dari SIG adalah data spasial dalam bentuk digital yang diperoleh melalui data satelit atau data lain terdigitasi. Analisis SIG memerlukan tenaga ahli sebagai *interpreter*, perangkat keras komputer dan software pendukung.



Gambar II. 5 : Pola Keterkaitan Peta Pada GIS

Sumber : p3m.amikom.ac.id

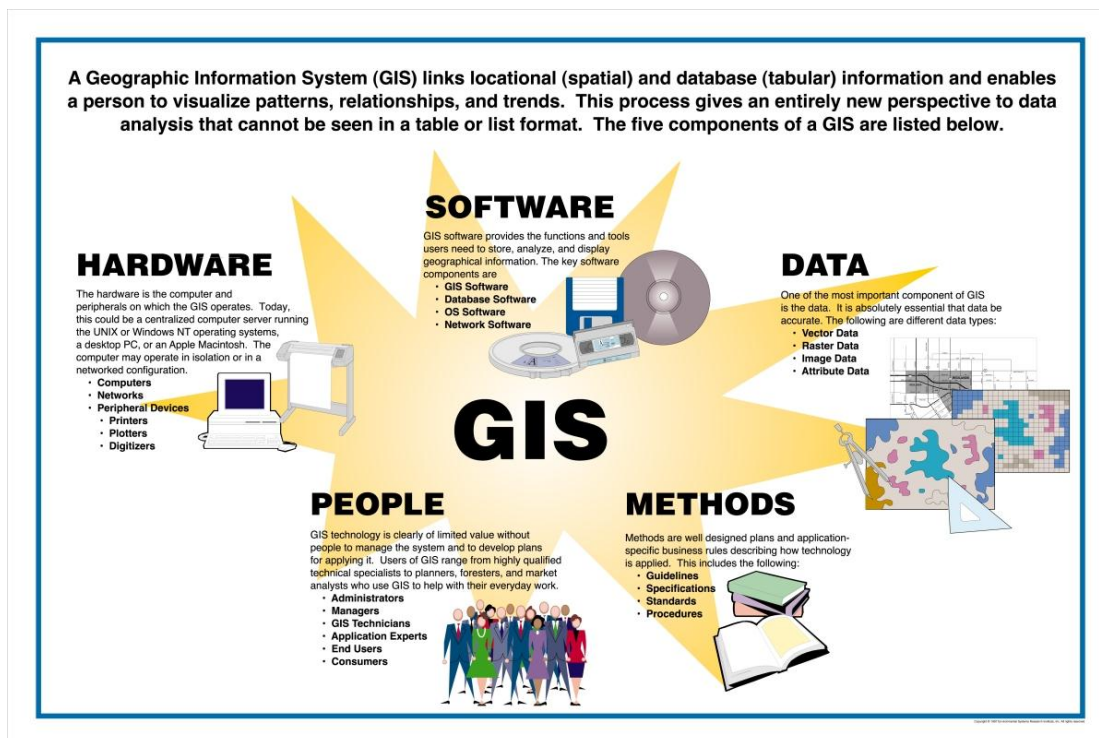
II.2.2. Teknis – Teknis Membangun GIS

Teknis dari pembanguna GIS sangat penting yang harus kita perhatikan dalam pembuatan suatu peta yang berkaitan dala gis yang pertama kita perhatikn adalah kita harus terutama memiliki Hardware yang membantu kita dalam pengerjaannya. Agar kita gampang seperti kita memiliki bahan-bahan dan komput dan loptop, baaru kiata mencari softwarena seperti menginstal software yang mendukung dalam pembuatan GIS. Data juga sangan penting dala pembuatan laporan yang akan dikerjakan. Data disini sangat penting buat bahan penulisan dan orang-orang yang mendukung dalam penulisan ini untuk mendapatkan suatu data dan buku –buku juga dal pendukung penulisan ini semua bahan – bahannya sangat penting dalam membangun GIS. Dalam survey pemetaan, yang khususnya mengelola data yang memiliki informasi spasial

(berreferensi keruangan), dalam arti yang lebih sempit adalah sistem komputer yang memiliki kemampuan yang membangun dan menyimpan, mengelola sistem, untuk mengumpulkan suatu titik koordinat suatu data yang digunakan dalam software yang telah dicustomize dimana suatu data berbasis open source seperti GRASS atau DIG atau secara khususnya suatu data yang membentuk dan memenuhi kebutuhan. Serta telah didefinisikan dengan sangat baik. Meskipun ada suatu perangkat yang gratis untuk melihat GIS dataset, akses public terhadap informasi geografis didominasi oleh sumber daya online seperti Google Earth dan pemetaan web interaktif.

Bagaimana GIS menggunakan informasi dalam penggunaan peta, jika data yang digunakan tidak dalam bentuk digital, yaitu dalam bentuk komputer dapat menggunakan dan mengenali, berbagai teknik dapat menangkap informasi. Peta dapat didigitalkan oleh tangan tracing dengan mouse komputer pada layar atau tablet digitalisasi untuk mengumpulkan koordinat fitur, Scanner elektronik juga dapat mengkonversi peta ke digital. Seperti gambar dibawah ini.

(p3m.amikom.ac.id)



Gambar II.6 : Teknis – Teknis Membangun GIS
Sumber : p3m.amikom.ac.id

II.3. Alat Bantu Pengembangan Sistem

Alat bantu pengembangan sistem yang digunakan dalam penulisan skripsi ini adalah sebagai berikut :

II.3.1. Menggunakan Unified Modeling Language(UML)

UML Adalah singkatan dari *Unified Modeling Language* yang berarti bahasa pemodelan standar. Berarti *UML* memiliki sintak dan simantik, ketika kita membuat medel menggunakan konsep *UML* dan aturan – aturan yang harus diikuti. Bagaimana elemen pada model – model yang kita buat berhubungan satu dengan lainnya harus mengikuti standar yang ada, *UML*

juga bukan sekedar diagram tetapi juga menceritakan konteksnya.(Prabowo Pudjo Widodo,Heriawati:2011,6).

UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :

1. Merancang perangkat lunak.
2. Sarana komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang yang diperlukan sistem.
4. Mendokumentasi sistem yang ada, proses – proses dan organisasinya.

II.3.2. Diagram – Diagram UML

Beberapa literature menyebutkan bahwa *UML* menyediakan Sembilan jenis diagram, yang lain menyebutkan delapan karena ada beberapa diagram yang digabung, misalnya diagram komunikasi, diagram urutan dan diagram pewaktuan digabung menjadi diagram interaksi. Namun demikian model-model itu dapat dikelompokkan berdasarkan sifatnya yaitu statis atau dinamis. Jenis diagram itu antara lain:

1. Diagram Kelas. Bersifat statis

Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek.

2. Diagram Paket (*Package Diagram*). Bersifat statis

Diagram ini memperlihatkan kumpulan kelas – kelas, merupakan bagian dari diagram komponen.

3. Diagram *use-case*. Bersifat statis

Diagram ini memperlihatkan himpunan *use-case* dan akto-aktor. Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku suatu sistem yang dibutuhkan serta diharapkan pengguna.

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 yang menekankan organisasi structural dari objek-objek yang menerima serta mengirim pesan.

6. Diagram *Statechar* (*Statechar Diagram*). Bersifat dinamis

Diagram status memperlihatkan keadaan-keadaan pada sistem, memuat status (*state*), transisi kejadian serta aktifitas.

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.

8. Diagram Komponen (*Component Diagram*). Bersifat dinamis

Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem atau perangkat lunak pada komponen-komponen yang telah ada sebelumnya.

9. Diagram *Deployment (Deployment Diagram)*. Bersifat statis

Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (*runtime*)

II.3.3. Diagram Use Case (*Use Case Diagram*)

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. (Prabowo, 2011 : 10).

Use case adalah deskripsi fungsi dari sebuah system dari perspektif pengguna. Use case bekerja dengan cara mendeskripsikan tipikal interaksi antara user (pengguna) sebuah system dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. (Munawar, 2005 : 63).

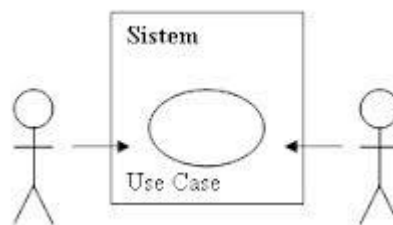
Komponen pembentuk diagram use case adalah :

1. Aktor (*actor*), menggambarkan pihak – pihak yang berperan dalam sistem.
2. *Use case*, aktivitas/sarana yang disiapkan oleh bisnis/sistem.
3. Hubungan (*link*), aktor mana saja yang terlibat dalam *use case* ini.

II.3.3.1. Aktor

Chonoles (2003 : bab 8) menyarankan sebelum membuat use case dan menentukan aktornya, agar mengidentifikasi siapa saja yang terlibat dalam sistem kita.

(Prabowo dan Herlawati, 2011 : 17).

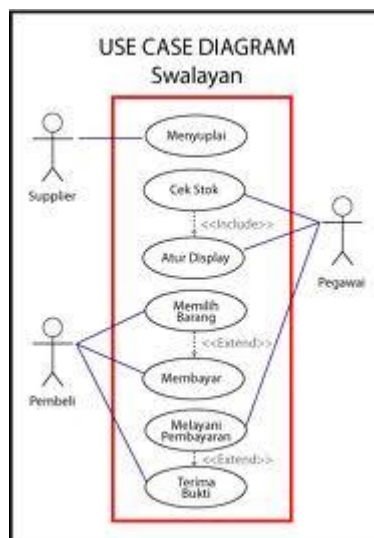


Gambar II.7. Contoh Aktor

(Prabowo Pudjo Widodo dan Herlawati, 2011 : 17)

II.3.3.2. Use Case

Menurut Pilone (2005 : bab 7.1), *use case* menggambarkan fungsi tertentu dalam suatu sistem berupa komponen, kejadian atau kelas. Sedangkan Whitten (2004 : 258) mengartikan *use case* sebagai urutan langkah – langkah yang secara tindakan saling terkait (skenario), baik terotomatisasi maupun secara manual, untuk tujuan melengkapi satu tugas bisnis tunggal. (Prabowo dan Herlawati, 2011 : 21).



Gambar II.8. Contoh Use case

(Prabowo Pudjo Widodo dan Herlawati, 2011 : 26)

II.3.3.3. Relasi antar use case / aktor

Pada diagram *use case*, relasi digambarkan sebagai sebuah garis antar dua simbol. Pemaknaan relasi berbeda – beda tergantung bagaimana garis tersebut digambar dan tipe simbol apa yang digunakan untuk menghubungkan garis tersebut.

(Prabowo dan Herlawati, 2011 : 24).

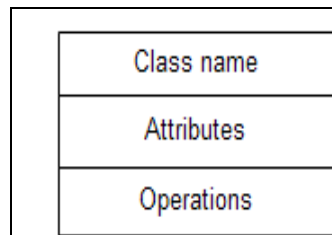
II.3.4. Diagram Kelas (*Class Diagram*)

Diagram kelas adalah inti dari proses pemodelan objek. Baik *forward engineering* maupun *reverse engineering* memanfaatkan diagram ini. *Forward engineering* adalah proses perubahan model menjadi kode program sedangkan *reverse engineering* sebaliknya merubah kode program menjadi model (Prabowo dan Herlawati, 2011 : 37).

Diagram kelas merupakan kumpulan kelas – kelas objek. Oleh karena itu pengertian kelas sangat penting sebelum merancang diagram kelas. Whitten (2004: 410), mengartikan kelas sebagai satu set objek yang memiliki atribut dan perilaku yang sama (Prabowo dan Herlawati, 2011 : 39).

Secara teknis, Pender (2003 : bab 5) mengartikan sebuah kelas sebagai suatu defenisi sumber daya yang termasuk di dalamnya informasi – informasi yang menggambarkan fitur suatu entitas yang sudah didefenisikan dalam kelasnya (Prabowo dan herlawati, 2011 : 39).

Kelas dinyatakan dalam kotak yang terbagi menjadi beberapa kompartemen. Kompartemen adalah area dalam kelas yang berisi informasi. Kompartemen pertama berisi nama kelas, berikutnya atribut dan yang terakhir operasi. Kompartemen – kompartemen itu bisa disembunyikan untuk memperjelas diagram yang kita buat.

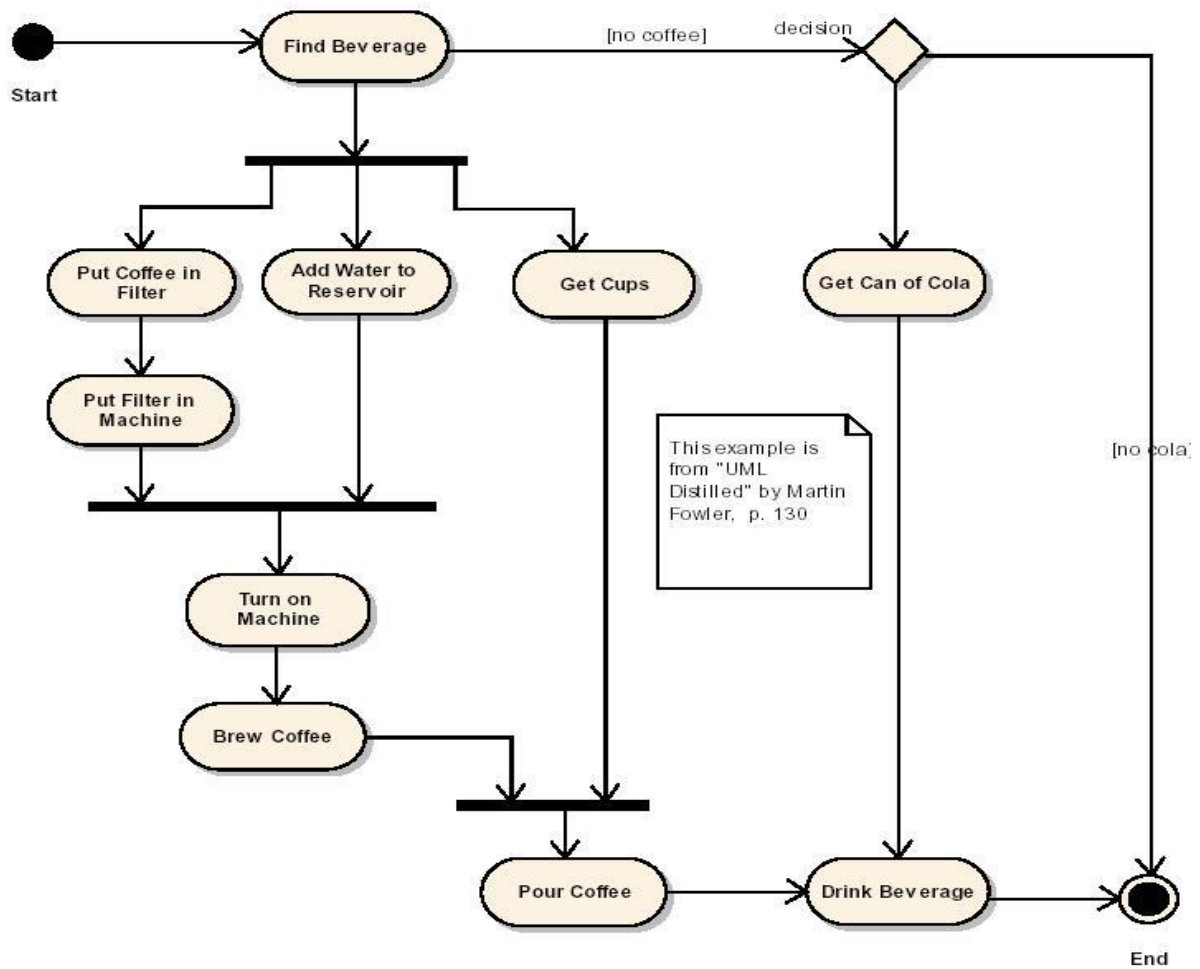


Gambar II.9. Notasi kelas sederhana
(Prabowo Pudjo Widodo dan Herlawati, 2011 : 41)

II.3.5. Diagram Aktifitas (*Activity Diagram*)

Diagram aktifitas (*Activity diagram*) adalah type 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 (Prabowo dan Herlawati, 2011 : 11).






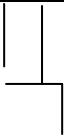
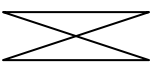
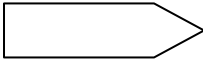
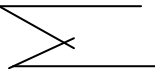

Diagram aktivitas lebih memfokuskan diri pada eksekusi dan alur sistem dari pada bagaimana sistem itu dirakit. Diagram ini tidak hanya memodelkan software melainkan memodelkan model bisnis juga. Diagram aktivitas menunjukkan aktivitas sistem dalam bentuk kumpulan aksi – aksi (Prabowo dan Herlawati, 2011 : 143).



Gambar II.10 Contoh Pembuatan Diagram Class

(Prabowo Pudjo Widodo dan Herlawati, 2011 : 175)

Tabel II.1 Simbol – simbol yang sering dipakai pada 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)

(Munawar, 2005 : 109-110)

II.3.6. Sequence Diagram

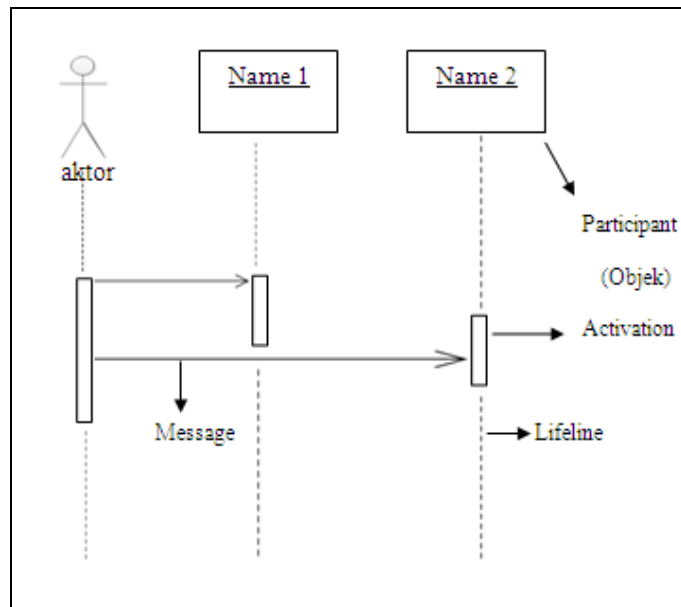
Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah scenario. Diagram ini menunjukkan sejumlah objek dan message (pesan) yang dilakukan diantara objek – objek ini di dalam usecase.

Komponen utama *sequence* diagram terdiri atas objek yang dituliskan dengan kotak segiempat bernama. Message diwakilkan oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan progress vertical. (Munawar, 2005 : 87).

Untuk membentuk diagram *sequence*, perlu diketahui notasi-notasi yang termasuk kedalam diagram *sequence*. Peraturan dalam membuat diagram *sequence* yaitu:

1. Objek atau partisipan diletakkan di dekat bagian atas diagram dengan urutan dari kiri ke kanan.
2. Setiap partisipan terhubung dengan garis titik-titik yang disebut *lifeline*. Pada *lifeline* terdapat kotak yang disebut *activation*. *Activation* mewakili sebuah eksekusi operasi dari partisipan. Panjang kotak ini berbanding lurus dengan durasi *activation*.
3. Sebuah message bergerak dari satu *participant* ke *participant* yang lain dan dari satu *lifeline* ke *lifeline* yang lain. Sebuah partisipan bisa mengirim sebuah pesan kepada dirinya sendiri.

4. *Time* adalah diagram yang mewakili waktu pada arah vertikal. Waktu dimulai dari atas ke bawah. Message yang lebih dekat dari atas akan dijalankan terlebih dahulu.



Gambar II.11. Simbol – simbol sequence diagram

(Munawar, 2005 : 89)

II.4. Website

Website atau situs juga dapat diartikan sebagai kumpulan halaman yang menampilkan informasi data teks, data gambar diam atau gerak, data animasi, suara, video dan atau gabungan dari semuanya, baik yang bersifat statis maupun dinamis yang membentuk satu rangkaian bangunan yang saling terkait dimana masing-masing dihubungkan dengan jaringan-jaringan halaman (*hyperlink*). Bersifat statis apabila isi informasi *website* tetap, jarang berubah, dan isi informasi search *web* dinamis, walaupun tidak tertutup kemungkinan digunakan untuk pemakaian lain.

II.5. Database

Istilah “Database” berawal dari ilmu computer. Meskipun kemudian artinya semakin luas, memasukan hal – hal yang diluar bidang elektronika, artikel mengenai database komputer. Catatan yang mirip dengan database sebenarnya sudah ada sebelum revolusi industry yaitu dalam bentuk buku besar, kuitansi dimana kumpulan data berhubung dengan bisnis.

Adapun konsep dasar dari Database adalah kumpulan dari catatan, atau potongan dari pengetahuan. Sebuah database memiliki penjelasan terstruktur dari jenis fakta yang tersimpan di dalamnya, penjelasan ini disebut dengan skema. Ada banyak cara untuk mengorganisasikan skema, atau model data. Model yang umum sekarang digunakan sekarang adalah model rasional, yang menurut istilah yaitu mewakili semua informasi dalam bentuk tabel yang saling berhubungan dimana setiap tabel terdiri dari kolom dan baris. Definisi sebenarnya adalah menggunakan terminologi matematika. Dalam model ini hubungan antara tabel diwakili dengan menggunakan nilai yang sama antara tabel.

(Abdul Kadir : 2008, 6)

II.5.1. Perangkat Untuk Membuat Database

Database dapat dibuat dan diolah dengan menggunakan suatu program komputer, yaitu yang biasa disebut dengan software (perangkat lunak). Software yang digunakan untuk mengelola dan memanggil query database disebut

Database Management System (DBMS) atau jika diterjemahkan kedalam bahasa Indonesia berarti “Sistem Manajemen Basis Data”.

DBMS terdiri dari dua komponen, yaitu Relational Database Management System (RDBMS) dan Overview of Database Management System (ODBMS). RDBMS meliputi Interface Drivers, SQL Engine, Transaction Engine, Relational Engine, dan Storage Engine. Sedangkan ODBMS meliputi Language Drivers, Query Engine, Transaction Engine, dan Storage Engine.

II.5.2 . MySQL

MySQL merupakan *software* RDBMS (atau *server database*) yang dapat mengolah *database* dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak *user* (*multi-user*), dan dapat melakukan suatu proses secara sinkron atau berbarengan (*multi-threaded*).

Saat ini, MySQL banyak digunakan di berbagai kalangan untuk melakukan penyimpanan dan pengolahan data, mulai dari kalangan akademis sampai ke industry, baik industry kecil menengah, maupun besar.

Beberapa contoh aplikasi yang menggunakan MySQL adalah: Joomla (www.jomla.org), Wordpress (www.wordpress.com) , MyBB (www.mybb.com), phpBB (www.phpbb.com), dan masih banyak sekali yang lainnya.(Budi Raharjo: 2011;21,22).

II.5.1. Alasan Menggunakan MySQL

Berikut ini beberapa alasan mengapa mereka memilih MySQL sebagai server database untuk aplikasi-aplikasi yang mereka kembangkan:

1. **Fleksibel**

MySQL dapat digunakan untuk mengembangkan aplikasi *desktop* maupun aplikasi web dengan menggunakan teknologi yang bervariasi. Ini berarti bahwa MySQL memiliki fleksibilitas terhadap teknologi yang akan digunakan sebagai pengembang aplikasi, apakah itu PHP, JSP, Java, Delphi, C++ maupun yang lainnya dengan cara menyediakan *plug-in* dan *driver* yang spesifik untuk masing-masing teknologi tersebut.

2. **Performa Tinggi**

MySQL memiliki mesin *query* dengan performa tinggi, dengan demikian proses transaksional dapat dilakukan dengan sangat cepat. Hal ini terbukti dengan digunakannya MySQL sebagai *database* dari beberapa aplikasi *web*.

3. **Lintas Platform**

MySQL dapat digunakan pada *platform* atau lingkungan yang beragam, bias Microsoft Windows, Linux, atau UNIX. Antar sistem operasi dapat dilakukan secara lebih mudah.

4. **Gratis**

MySQL dapat digunakan secara gratis. Meskipun demikian, ada juga *software* MySQL yang bersifat komersial. Biasanya yang sudah ditambahi dengan kemampuan spesifik dan mendapat pelayanan dari *technical support*.

5. Proteksi data yang Handal

Perlindungan terhadap keamanan data merupakan hal nomor satu yang dilakukan oleh para profesional yang dibidang *database*. MySQL menyediakan mekanisme yang *powerfull* untuk menangani hal tersebut, yaitu dengan menyediakan fasilitas manajemen *user*, enkripsi data dan lain sebagainya.

(www.google.com)

6. Komunitas Luas

Karena penggunaannya banyak maka MySQL memiliki komunitas yang luas. Hal ini berguna jika kita menemui suatu permasalahan dalam proses pengolahan data menggunakan MySQL.

II.5.2. Keistimewaan *MySQL*

MySQL memiliki beberapa keistimewaan, antara lain :

1. Portabilitas. *MySQL* dapat berjalan stabil pada berbagai sistem operasi seperti Windows, Linux, FreeBSD, Mac Os X Server, Solaris, Amiga, dan masih banyak lagi.

2. Perangkat lunak sumber terbuka. *MySQL* didistribusikan sebagai perangkat lunak sumber terbuka, dibawah lisensi GPL sehingga dapat digunakan secara gratis.
3. Multi-user. *MySQL* dapat digunakan oleh beberapa pengguna dalam waktu yang bersamaan tanpa mengalami masalah atau konflik.
4. '*Performance tuning*', *MySQL* memiliki kecepatan yang menakjubkan dalam menangani query sederhana, dengan kata lain dapat memproses lebih banyak SQL per satuan waktu.
5. Ragam tipe data. *MySQL* memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.
6. Perintah dan Fungsi. *MySQL* memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam perintah (*query*).
7. Keamanan. *MySQL* memiliki beberapa lapisan keamanan seperti level *subnetmask*, nama *host*, dan izin akses *user* dengan sistem perizinan yang mendetail serta sandi terenkripsi.
8. Skalabilitas dan Pembatasan. *MySQL* mampu menangani basis data dalam skala besar, dengan jumlah rekaman (*records*) lebih dari 50 juta dan 60 ribu tabel serta 5 milyar baris. Selain itu batas indeks yang dapat ditampung mencapai 32 indeks pada tiap tabelnya.
9. Konektivitas. *MySQL* dapat melakukan koneksi dengan klien menggunakan protokol TCP/IP, *Unix socket* (UNIX), atau *Named Pipes* (NT).

10. Lokalisasi *MySQL* dapat mendeteksi pesan kesalahan pada klien dengan menggunakan lebih dari dua puluh bahasa. Meski pun demikian, bahasa Indonesia belum termasuk di dalamnya.
11. Antar Muka. *MySQL* memiliki antar muka (*interface*) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).
12. Klien dan Peralatan. *MySQL* dilengkapi dengan berbagai peralatan (*tool*) yang dapat digunakan untuk administrasi basis data, dan pada setiap peralatan yang ada disertakan petunjuk online.
13. Struktur tabel. *MySQL* memiliki struktur tabel yang lebih fleksibel dalam menangani `ALTER TABLE`, dibandingkan basis data lainnya semacam *PostgreSQL* ataupun *Oracle*.

Berikut beberapa fungsi *MySQL* yang sering digunakan beserta deskripsi singkat dari fungsi-fungsi tersebut:

a. *MySQL_connect* ()

Sebelum melakukan operasi apapun pada *MySQL*, hal pertama yang harus dilakukan adalah membuka koneksi dengan *MySQL*. Untuk menjalankan fungsi tersebut, maka kita harus menjalankan fungsi *MySQL_connect*() dengan mengetikkan *hostname*, *username* dan *password*.

b. *MySQL_create_db* ()

Untuk membuat *database* yang hanya dapat diakses oleh *username* dan *password*, digunakan fungsi *MySQL_create_db()* setelah koneksi ke *MySQL* dibuka.

c. *MySQL_query ()*

Fungsi *MySQL_query()* digunakan untuk mengirim *query* kepada *database* aktif yang sudah dipilih.

d. *MySQL_fetch_row ()*

Digunakan untuk mengambil hasil *query* dari *database* sebagai data numerik *array*. Fungsi ini mengambil data baris per baris (*record*) dari *database* secara berulang-ulang sampai kondisi *query* menjadi *false*.

e. *MySQL_fetch_array ()*

Digunakan untuk mengambil hasil *query* dari *database* sebagai data *array* asosiatif.

f. *MySQL_num_rows ()*

Digunakan untuk menghitung jumlah baris hasil *query*.

II.6. PHP

PHP merupakan singkatan dari *PHP Hypertext Preprocessor*. Ia merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan proses deserver, hasilnya yang dikirim ke klien, tempat pemakai pengguna *browser*.

Contoh terkenal dari aplikasi *PHP* adalah *PHPBB* dan *MediaWiki* (software di belakang *Wikipedia*). *PHP* juga dapat dilihat sebagai pilihan lain dari

ASP.NET/C#/VB.NET Microsoft, ColdFusion Macromedia, JSP/Java Sun Microsystems, dan CGI/Perl. Contoh aplikasi lain yang lebih kompleks berupa CMS yang dibangun menggunakan *PHP* adalah Mambo, Joomla!, Postnuke, Xaraya, dan lain-lain.

II.6.1. Sejarah *PHP*

Pada awalnya *PHP* merupakan kependekan dari *Personal Home Page* (Situs Personal). *PHP* pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu *PHP* masih bernama FI (*Form Interpreted*), yang wujudnya berupa sekumpulan script yang digunakan untuk mengolah data *form* dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya *PHP/FI*. Dengan perilisannya kode sumber ini menjadi *open source*, maka banyak *programmer* yang tertarik untuk ikut mengembangkan *PHP*.

Pada November 1997, dirilis *PHP/FI 2.0*. Pada rilis ini *interpreter PHP* sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan *PHP/FI* secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang *interpreter PHP* menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian

pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk *PHP* dan meresmikan rilis tersebut sebagai *PHP 3.0* dan singkatan *PHP* dirubah menjadi akronim berulang *PHP: Hypertext Preprocessing*.

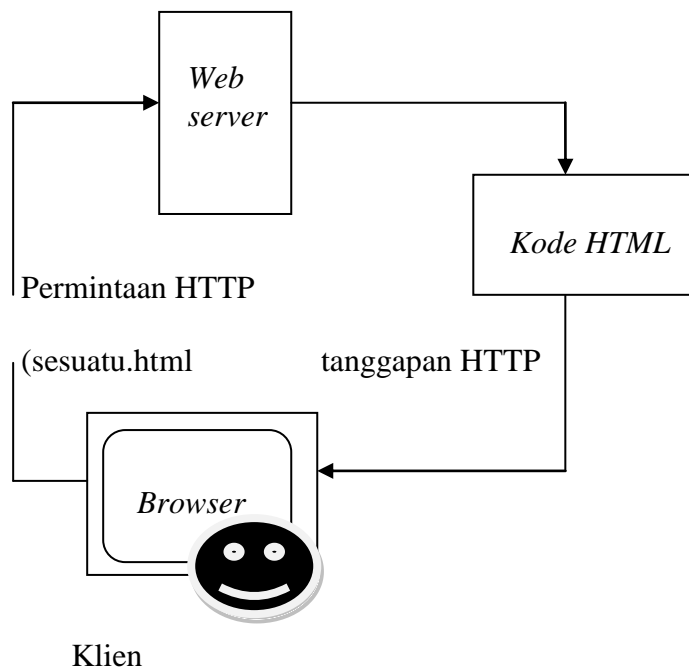
Pada pertengahan tahun 1999, Zend merilis interpreter *PHP* baru dan rilis tersebut dikenal dengan *PHP 4.0*. *PHP 4.0* adalah versi *PHP* yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis *PHP 5.0*. Dalam versi ini, inti dari interpreter *PHP* mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam *PHP* untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek.

II.6.2. Konsep kerja PHP

Model kerja HTML diawali dengan permintaan suatu alamat web oleh *browser* berdasarkan URL (*Uniform Resource Locator*) atau dikenal dengan sebutan alamat internet browser mendapatka alamat dari *web server*, mengidentifikasi halaman yang dikehendaki, dan menyampaikan segala informasi yang dibutuhkan oleh *web server*.

Selanjutnya *web server* akan mencari *file* yang diminta dan diberikan isinya ke *web server* atau *browser*. *Browser* yang mendapatkan isinya segera melakukan proses penerjemahan kode HTML dan menampilkannya kelayar pemakai.

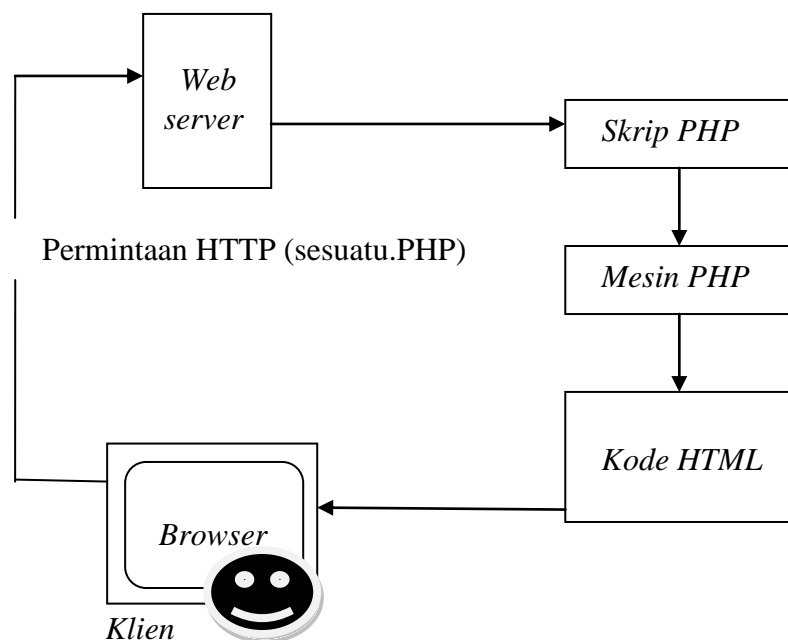


Gambar II.12 Skema HTML

Sumber ; Abdul Kadir ; 2008, 5

Bagaimana halnya kalau yang diminta adalah sebuah halaman PHP, prinsipnya serupa dengan kode HTML. Hanya saja, ketika berkas PHP diminta didapatkan oleh *web server*, isinya segera dikirim kemesin PHP dan mesin

ini yang memproses dan memberikan hasilnya berupa HTML ke *web server*. Selanjutnya, *web server* menyampaikan ke klien.



Gambar II.13. Skema PHP

Sumber ; (Abdul Kadir ; 2008, 6)

II.7. Google Map

Google Maps menurut pengertian Wikipedia adalah layanan aplikasi pemetaan berbasis web dan teknologi yang telah disediakan oleh Google, bersifat free (tidak untuk penggunaan commercial), yang memiliki kemampuan terhadap banyak layanan pemetaan berbasis web. Intinya

bahwa Google Maps menurut saya adalah layanan sistem pemetaan berbasis web service yang disediakan oleh Google. Sifatnya adalah server side atau peta digenerate oleh user yang sebelumnya sudah tersimpan di server google.

(widyawarta.wordpress.com)