BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

2.1.1 Pengertian sistem

Terdapat 2 kelompok pendekatan dalam mendefenisikan sistem yaitu menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya. Pendekatan sistem yang di tekankan pada prosedur menurut Jogiyanto HM(2001:1) mendefenisikan sistem sebagai berikut :

“sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu”.

Pendekatan sistem yang merupakan jaringan kerja dari prosedur lebih menekankan pada elemen atau komponen nya menurut Jogiyanto HM(2001:2)mendefenisikan sistem sebagai berikut: ”sistem merupakan kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu”. Sistem dapat di artikan sebagai kumpulan atau grup dari subsistem/bagian/komponen apapun baik fisik maupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai suatu tujuan tertentu. Kumpulan elemen tersebut terdiri dari hardware,software,prosedur,sumber daya dan data-data yang saling berhubungan dan saling melengkapi satu sama lain untuk mencapai suatu tujuan tertentu.

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu yaitu :

1. Komponen sistem(component)

   Suatu system terdiri dari sejumlah komponen-komponen yang saling berinteraksi ,yang artinya saling bekerja sama membentuk suatu kesatuan.komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem bagian-bagian dari sistem.setiap subsistem mempunyai sifat-sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batas sistem(boundary)

   Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya.Batas sistem ini memungkinkan suatu sistem di pandang sebagai suatu kesatuan.Batas suatu sistem menunjukan ruang lingkup dari sistem tertentu.

3. Lingkungan luar sistem (enviroment)

   Lingkungan luar sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut.

4. Penghubung system (interface)

   Penghubung merupakan media penghubung antara suatu subsistem dengan subsistem yang lainnya.melalui media penghubung ini memungkinkan sumber-sumber daya mengalir dari suatu subsistem ke subsistem yang lain nya.Dengan penghubung satu subsistem dapat berinteraksi dengan subsistem yang lainnya membentuk satu kesatuan.

5. Masukan sistem (input)

   Energy yang di masukkan kedalam suatu sistem,masukan ini dapat berupa masukan perawatan(maintanance) dan memasukkan signal.

   Masukan perawatan yaitu energy yang dimasukkan sistem supaya sistem tersebut beroperasi,sedangkan masukkan signal yaitu energy yang diproses untuk mendapatkan keluaran.

6. Keluaran sistem(output)

   Keluaran adalah hasil dari energy yang di olah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

7. Pengolah sistem(proses)

   Suatu sistem dapat mempunyai suatu bagian pengolahan atau sistem itu sendiri sebagai pengolahnya,pengolah akan merubah masukan menjadi keluaran.

8. Sasaran sistem(goal)

   Suatu sistem pasti mempunyai tujuan atau sasaran. Sasaran dari suatu sistem sangat menentukan sekali masukan yang di butuhkan dan keluaran yang akan dihasilkan sistem.suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan nya.

2.2 Konsep Dasar Informasi

2.2.1 Pengertian Informasi

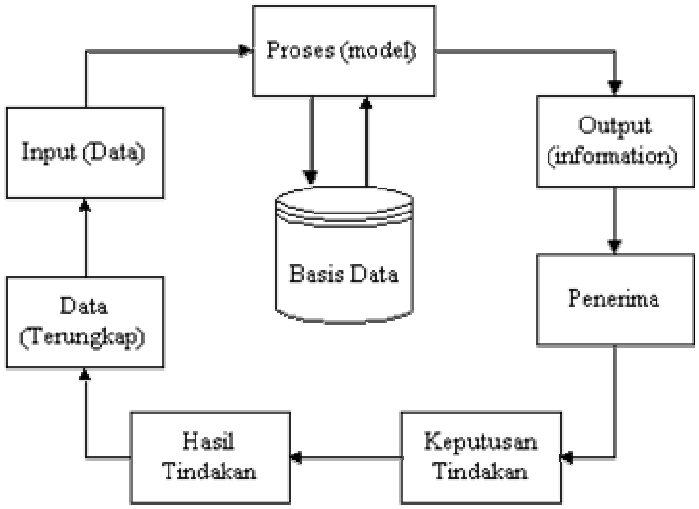
Menurut Jogiyanto HM (2001:8) Informasi dapat di defenisikan sebagai berikut *: "informasi adalah data yang di olah menjadi bentuk yang berguna dan lebih berarti bagi yang menerimanya"* . Sumber dari informasi adalah data.

2.2.1.1 Pengertian Data

Menurut Azhar Susanto (2000:37) Mendefenisikan data sebagai berikut : "data adalah fakta atau apapun yang dapat digunakan sebagai input untuk menghasilkan informasi". Maka data adalah suatu karangan tertulis mengenai fakta atau kenyataan yang masih berdiri sendiri ,belum mempunyai pengertian sebagai kelompok,belum terkoordinasi satu sama lain dan belum di olah sesuai dengan keperluan tertentu menjadi suatu bentuk yang sesuia dengan keperluan pengguna informasi yang bersangkutan.

2.2.2 Siklus Informasi

Informasi merupakan suatu proses perubahan dunia menjadi informasi. Data yang di olah untuk menghasilkan informasi menggunakan suatu model proses tertentu. Data yang diolah melalui suatu model menjadi informasi,penerima kemudian menerima informasi tersebut ,membuat dan suatu keputusan dan melakukan tindakan yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus,adapun siklus informasi sebagai berikut :



Gambar 2.1 Siklus Informasi

Sumber : http://referensisisteminformasi.blogspot.com/2008/07/siklus-informasi.html

2.2.3 Kualitas Informasi

Kualitas suatu informasi dilihat dari 3 hal yaitu :

a. Akurat artinya suatu informasi harus bebas dari kesalahan2 dan tidak menyesatkan bagi si pemakai.
b. Tepat waktu artinya suatu informasi harus sesuai dengan keadaan saat itu,keterlambatan suatu informasi bisa berakibat fatal bagi organisasi pemakainya dalam hal ini dikarenakan informasi merupakan landasan dalam pengambilan sebuah keputusan.
c. Relevan artinya suatu informasi harus bermanfaat bagi pengguna nya dalam waktu pendek ataupun dalam waktu panjang.

2.3 Pengertian Sistem Informasi

Sistem informasi adalah kumpulan dari sub sub sistem baik fisik maupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai suatu tujuan yaitu mengolah data menjadi informasi yang berguna .komponen dasar sistem informasi meliputi:

1. Blok masukan (input blok)

   Input mewakili data yang masuk ke dalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan di masukkan ,dapat berupa dokumen –dokumen dasar.

2. Blok model (model block)

   Blok ini terdiri dari kombinasi prosedur,logika dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara tertentu untuk menghasilkan keluaran yang di inginkan.

3. Blok teknologi(technology block)

   Teknologi digunakan untuk menerima input,menjalankan model,menyimpan dan mengakses data,menghasilkan dan mengirimkan keluaran dan membantu pengendalian sistem secara keseluruhan.

4. Blok keluaran(blok output)

   Keluaran merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

5. Blok basis data(database block)

   Merupakan kumpulan data yang saling berhubungan satu sama lain,tersimpan di perangkat keras computer dan digunakan perangkat lunak untuk memanipulasi nya.

   Data perlu disimpan dalam basis data untuk keperluaan penyediaan informasi lebih lanjut.data di dalam basis data perlu di organisasikan sedemikian rupa,supaya informasi yang di hasilkan berkualitas.

6. Blok kendali(controls block)

   Beberapa pengendalian perlu di rancang dan di terapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat di cegah ataupun bila terlanjur terjadi kesalahan langsung cepat diatasi.

   Tujuan dilakukan nya tahap ini sebagai berikut:

   a) Membuat keputusan apabila sistem saat ini mempunyai masalah atau sudah tidak berfungsi secara baik dan hasil analisisnya digunakan sebagai dasar untuk memperbaiki sistem
   b) Mengetahui ruang lingkup pekerjaan yang akan ditangani.
   c) Memahami sistem yang sedang berjalan saat ini,mengidentifikasikan dan mencari solusi.

2.4 Sistem basis data

Menurut Fathansyah,Ir (2002:9) Basis data adalah :

Sistem basis data merupakan sistem yang terdiri dari atas sekumpulan tabel yang saling berhuungan (dalam sebuah basis data di sebuah sistem komputer) dan

sekumpulan program(DBMS)yang menungkinkan beberapa pemakai lain untuk mengakses dan memanipulasi table-tabel tersebut.

Sistem Basis data pada dasarnya di buat untuk tujuan Sebagai berikut: (fatansyah 2002:5)

1. Kecepatan dan kemudahan

   Dengan sistem basis data diharapkan pengolahan data dapat lebih cepat dan lebih mudah di bandingkan dengan sistem manual.

2. Efisiensi ruang penyimpanan

   Dengan basis data,efisiensi /optimalisasi penggunaan ruang penyimpan dapat dilakukan,karena kita dapat melakukan penekanan jumlah redudansi data.

3. Keakuratan

   Pemanfaatan pengkodean atau pembentukan relasi antar data bersama dengan penerapan aturan atau batasan tipe,data,domain data ,keunikan data dan sebagainya ,yg secara ketat dapat diterapkan dalam sebuah basis data ,sangat berguna untuk menekan ketidak akuratan masukan ataupun penyimpanan data.

4. Ketersediaan

   Dengan pertumbuhan data yang semakin banyak,maka di perlukan tempat penyimpanan yang sangat besar,karena itu kita perlu memilah data dalam bentuk kategori-kategori tertentu sehingga data yang tidak terlalu penting dapat di hapus

5. Kelengkapan

   Untuk mengakomodasi kebutuhan kelengkapan data yang semaki berkembang maka kita tidak hanya dapat menambah record-record data tetapi juga melakukan

   perubahan struktur dalam basis data,baik dalam bentuk penambahan objek baru (tabel)atau dengan penambahan field-field baru pada suatu tabel.

6. Keamanan

   Dengan sistem basis data kita dapat memproteksi semua objek yang ada dengan berbagai tingkatan keamanan.

7. Kebersamaan Pemakaian

   Pemakai basis data biasanya terdiri dari banyak pemakai,sehingga hendaknya basis data harus bisa di akses oleh banyak pemakai dalam waktu yang bersamaan.

Dalam sebuah Basis Data secara lengkap akan terdapat komponen-komponen utama yaitu

1. Hardware
2. Operating system
3. Database
4. Sistem(aplikasi/perangkat lunak)pengelola basis data(DBMS)
5. User

Keuntungan Database adalah :

a. Mereduksi redudansi
b. Data dapat di share antar aplikasi

c. Standarisasi data dapat di lakukan

d. Batasan security dapat diterapkan

e. Mengelola integritas(keterjaminan akurasi)data

f. Menyeimbangkan kebutuhan yang saling konflik

g. Independesi data (obyektif DBS)

Struktur sistem Database :

a. File manager : mengelola space dan struktur data.

b. Database manager : menyediakan antar muka dengan data fisik

c. Query processor: menterjemahkan query ke instruksi yang dimengerti database manager

d. DML precompiler : mengkonversi perintah DML menjadi calls normal.

e. DDL compiler : menkonversi perintah DDL menjadi meta data.

2.5 Alat-Alat (Tools) Yang Digunakan

2.5.1 UML (Unified Modeling Language)

Menurut Bambang Hariyanto. Ir.,MT. (2004:17)

UML adalah bahasa grafis untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. UML berorientasi objek menerapkan banyak level abstraksi, tidak bergantung proses pengembangan, tidak bergantung bahasa dan teknologi. Pemaduan beberapa notasi di beragam metodologi usaha bersama dari banyak pihak, di dukung oleh kakas- kakas yang

di integrasikan lewat XML . Standar UML di kelola oleh OMG(Object Management Group)

2.5.1.1 Pengenalan UML

UML adalah bahasa pemodelan untuk menspesifikasikan, memvisualisasikan, membangun dan mendokumentasikan artifak-artifak dari sistem.

1. Dari dalam sistem intensive process metode diterapkan sebagai proses untuk menurunkan atau mengevolusikan sistem.
2. Sebagai bahasa UML digunakan untuk komunikasi yaitu alat untuk menangkap pengetahuan mengenal suatu subyek dan mengekspresikan pengetahuan (sintaks) yang mempedulikan subyek. Untuk maksud berkomunikasi subyek adalah sistem yang di bahas.
3. Begitu diterapkan untuk memvisualisasikan sistem UML digunakan untuk menjelaskan sistem secara visual. Sebelum direalisasikan.
4. Begitu di terapkan untuk membangun sistem UML dapat digunakan untuk memandu realisasi sistem serupa dengan “”blue print”.
5. Begitu diterapkan untuk mendokumentasikan sistem, UML dapat di gunakan untuk menangkap pengetahuan sistem pada seluruh siklus hidup.

UML Bukanlah

1. Bahasa pemprograman visual.
2. Spesifikasi kakas, tapi spesifikasi pemodelan
3. Proses, tapi memungkinkan proses-proses.

Tujuan UML

1. Menyediakan bahasa pemodelan visual yang ekspresif dan siap pakai untuk mengembangkan pertukaran model-model yang berarti.
2. Menyediakan mekanisme perluasan dan spesialisasi untuk memperluas konsep-konsep inti
3. Mendukung spesifikasi independent bahasa pemprograman dan proses pengembangan tertentu.
4. Menyediakan basis formal untuk pemahaman bahasa pemodelan.
5. Mendorong pertumbuhan pasar kakas berorientasi objek.
6. mendukung konsep-konsep pengembangan level lebih tinggi seperti komponen kolaborasi framework dan pattern.

2.5.1.2 Bagian-Bagian Dari UML

Bagian-bagian dari UML dibagi menjadi 2 kategori, yakni view dan diagram.

View digunakan untuk melihat sistem yang di modelkan dari beberapa aspek yang berbeda. View bukan melihat grafik, tapi merupakan suatu abstraksi yang berisi sejumlah diagram. Beberapa jenis view dalam UML antara lain usecase,component view,deployment view.

View

A. Usecase View

Mendeskripsikan fngsionalitas sistem yang seharusnya dilakukan sesuai yang di inginkan external actors. Actor yang berinteraksi dengan sistem

berupa user atau sistem lain nya.View ini di gambarkan dalam usecase diagram dan kadang-kadang dengan activity diagram. View ini digunakan terutama untuk pelanggan perancangan (designer), pengembang (developer), dan penguji sistem(tester).

B. Logical View

Mendeskripsikan bagaimana fungsionalitas dari sistem, struktur statis (class,object dan relationship) dan kolaborasi dinamis yang terjadi ketika object mengirim pesan ke object lain dalam suatu fungsi tertentu. View ini di gambarkan dalam class diagram untuk struktur statis dan dalam state, sequence, collaboration, dan activity diagram untuk model dinamisnya view ini digunakan untuk perancangan (designer) dan pengembang (developer).

C. Component View

Mendeskripsikan implementasi dan ketergantungan modul komponen yang merupakan tipe lainnya dari code module, diperlihatkan dengan struktur dan ketergantungan juga alokasi sumber daya komponen dan informasiadministrative lain nya. View ini di gambar kan dalam component view dan digunakan untuk pengembang.

D. Concurrency View

Membagi sistem kedalam proses dan processor. View ini digambarkan dalam diagram dinamis (state, sequence, collaboration, dan activity diagram) dan diagram implementasi (Component Development

Diagram) serta digunakan untuk pengembang (developer), pengintegrasi (Integrator), dan penguji (tester).

E. Deployment View

Mendeskripsikan fisik dari sistem seperti computer dan perangkat (nodes) dan bagaimana hubungan nya dengan yang lain nya.

Diagram

Diagram bentuk grafik yang menunjukan symbol elemen model yang di susun untuk mengilustrasikan bagian atau aspek tertentu dan ketika digambarkan biasanya dialokasikan untuk view tertentu, adapun jenis diagram antara lain :

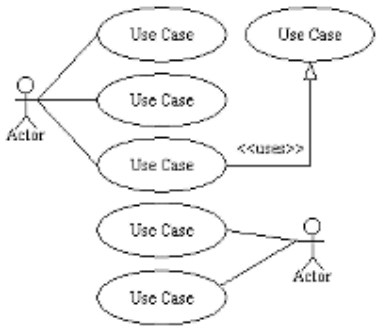
A. Usecase Diagram.

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah usecase merepresentasikan sebuah interaksi antara aktor dengan sistem. Use case merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-create sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

Use case diagram dapat sangat membantu bila kita sedang menyusun requirement sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang test case untuk semua feature yang ada pada sistem. Sebuah use case dapat meng-include fungsionalitas use case lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa use

case yang di-include akan dipanggil setiap kali use case yang meng-include dieksekusi secara normal. Sebuah use case dapat di-include oleh lebih dari satu use case lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang common. Sebuah use case juga dapat meng-extend use case lain dengan behaviour-nya sendiri. Sementara hubungan generalisasi antar use case menunjukkan bahwa use case yang satu merupakan spesialisasi dari yang lain.

Contoh use case diagram :



Gambar 2.2 Usecase Diagram

Sumber : http://ikc.cbn.net.id/umum/yanti-uml.php

B. Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). Class diagram

menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain.

Class memiliki tiga area pokok :

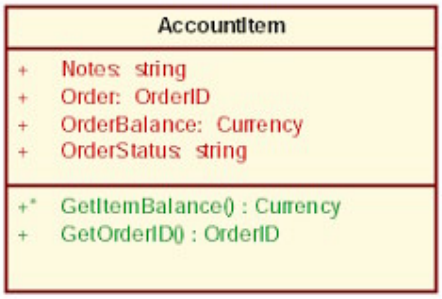
1. Nama (dan stereotype)
2. Atribut
3. Metoda

Atribut dan metoda dapat memiliki salah satu sifat berikut :

a) Private, tidak dapat dipanggil dari luar class yang bersangkutan

b) Protected, hanya dapat dipanggil oleh class yang bersangkutan dan anak-anak yang mewarisinya

c) Public, dapat dipanggil oleh siapa saja

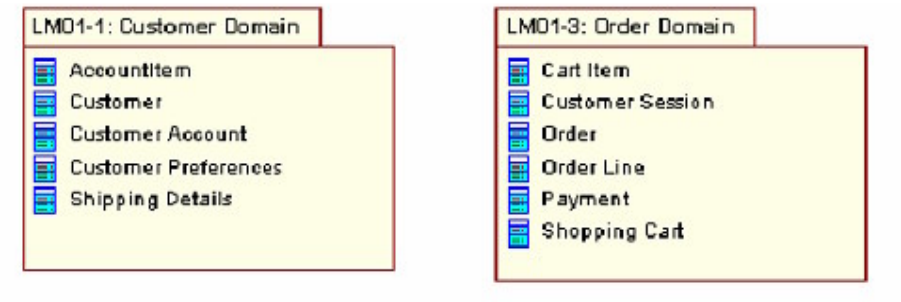


Gambar 2.3 Class Ikon diagram

Sumber : http://ikc.cbn.net.id/umum/yanti-uml.php

Class dapat merupakan implementasi dari sebuah interface, yaitu class abstrak yang hanya memiliki metoda. Interface tidak dapat langsung diinstansiasikan, tetapi harus diimplementasikan dahulu menjadi sebuah class. Dengan demikian interface mendukung resolusi metoda pada saat run-time.

Sesuai dengan perkembangan class model, class dapat dikelompokkan menjadi package. Kita juga dapat membuat diagram yang terdiri atas package.



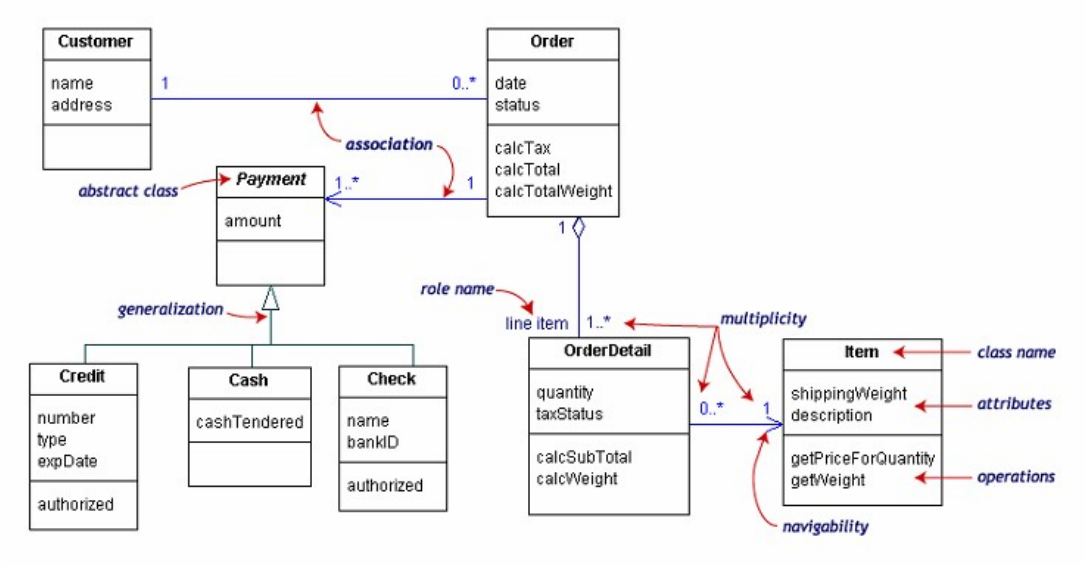
Gambar 2.4 Relation Class Diagram

Sumber : http://ikc.cbn.net.id/umum/yanti-uml.php

Hubungan Antar Class

1. Asosiasi, yaitu hubungan statis antar class. Umumnya menggambarkan class yang memiliki atribut berupa class lain, atau class yang harus mengetahui eksistensi class lain. Panah navigability menunjukkan arah query antar class.
2. Agregasi, yaitu hubungan yang menyatakan bagian ("terdiri atas..").
3. Pewarisan, yaitu hubungan hirarkis antar class. Class dapat diturunkan dari class lain dan mewarisi semua atribut dan metoda class asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari class yang diwarisinya. Kebalikan dari pewarisan adalah generalisasi.
4. Hubungan dinamis, yaitu rangkaian pesan (message) yang di-passing dari satu class kepada class lain. Hubungan dinamis dapat digambarkan dengan menggunakan sequence diagram yang akan dijelaskan kemudian.

Contoh class diagram:



Gambar 2.5 Class Diagram

Sumber : http://ikc.cbn.net.id/umum/yanti-uml.php

C. Statechart Diagram

Statechart diagram menggambarkan transisi dan perubahan keadaan (dari satu state ke state lainnya suatu objek pada sistem sebagai akibat dari stimuli yang diterima. Pada umumnya statech diagram menggambarkan class tertentu (satu class dapat memiliki lebih dari satu statech diagram). Dalam UML, state digambarkan berbentuk segiempat dengan sudut membulat dan memiliki nama sesuai kondisinya saat itu. Transisi antar state umumnya memiliki kondisi guard yang merupakan syarat terjadinya transisi yang bersangkutan, dituliskan dalam kurung siku. Action yang dilakukan sebagai akibat dari event tertentu dituliskan dengan

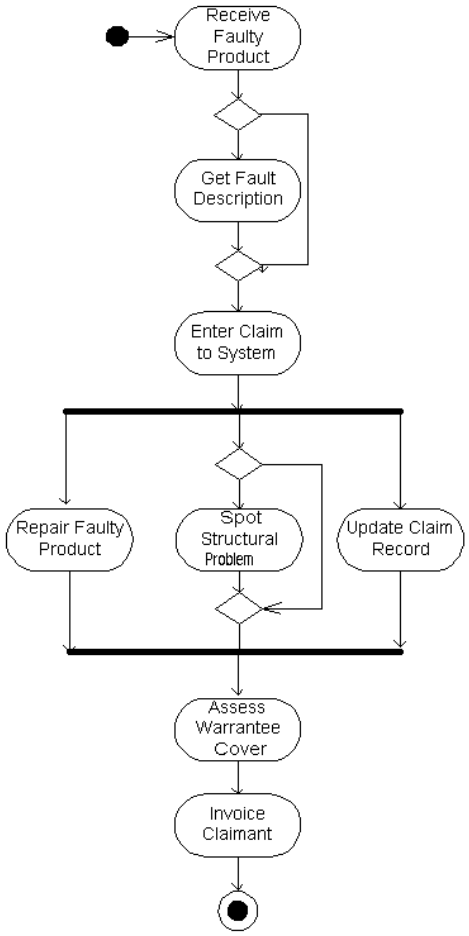
diawali garis miring. Titik awal dan akhir digambarkan berbentuk lingkaran berwarna penuh dan berwarna setengah.

D. Activity Diagram

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Activity diagram merupakan state diagram khusus, di mana sebagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (internal processing). Oleh karena itu activity diagram tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Sebuah aktivitas dapat direalisasikan oleh satu use case atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara use case menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.Sama seperti state, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. Decision digunakan untuk menggambarkan behaviour pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (fork dan join) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal. Activity diagram dapat dibagi menjadi beberapa object swimlane untuk

menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

Contoh activity diagram tanpa swimlane:



Gambar 2.6 Diagram Activity

Sumber :

http://www.idemployee.id.tue.nl/p.markopoulos/downloadablePapers/TUPIS2000/P_MarkopoulosTupis00.html

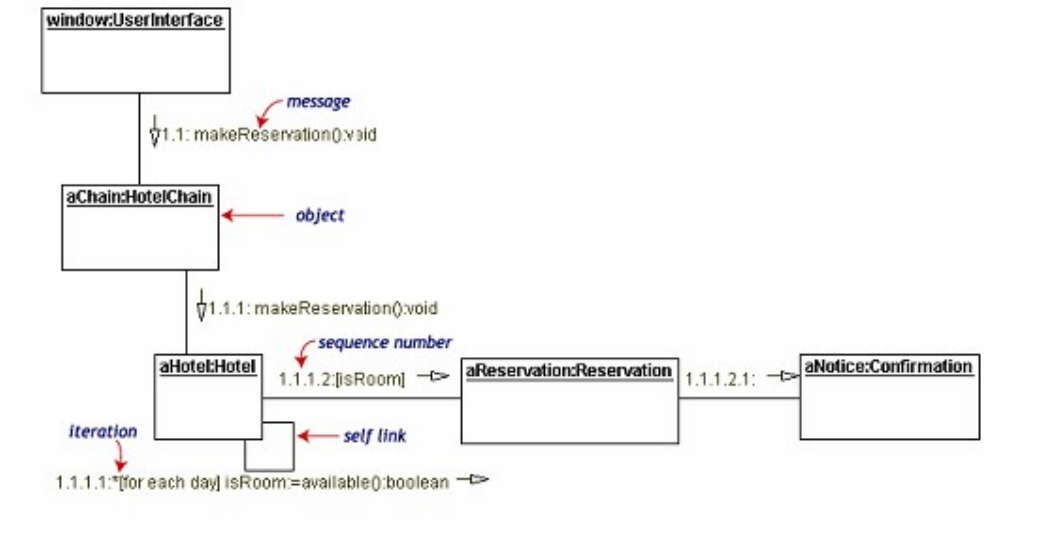
E. Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa message yang digambarkan terhadap waktu. Sequence diagram terdiri atar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan output tertentu. Diawali dari apa yang men-trigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki lifeline vertikal. Message digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, message akan dipetakan menjadi operasi/metoda dari class. Activation bar menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah message.Untuk objek-objek yang memiliki sifat khusus, standar UML mendefinisikan icon khusus untukobjek boundary, controller dan persistent entity.

F.Collaboration Diagram

Collaboration diagram juga menggambarkan interaksi antar objek seperti sequence diagram, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian message. Setiap message memiliki sequence number, di mana message dari level tertinggi memiliki nomor 1.

Messages dari level yang sama memiliki prefiks yang sama.



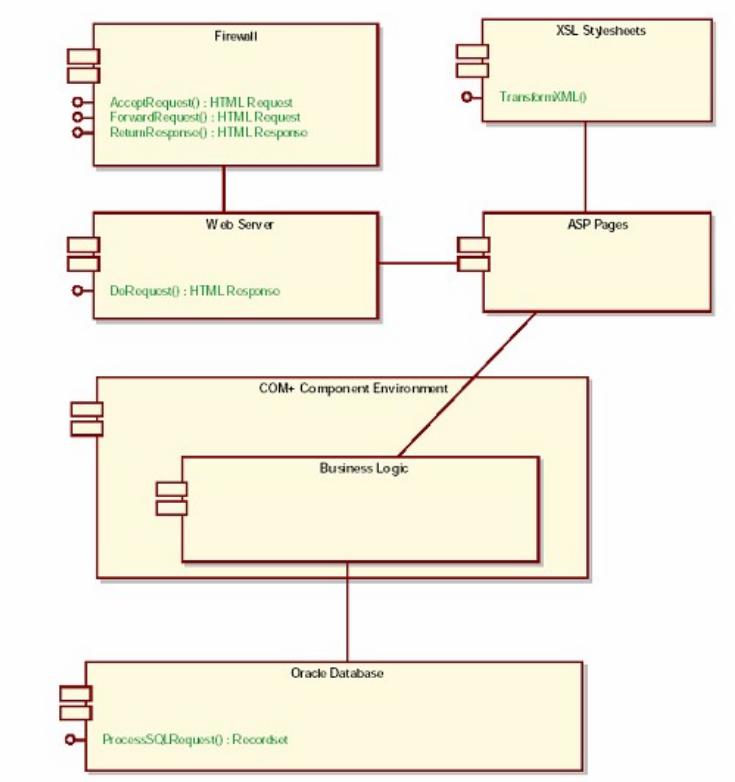
Gambar 2.7 Collaboration Diagram

Sumber : http://ikc.cbn.net.id/umum/yanti-uml.php

F. Component Diagram

Component diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (dependency) di antaranya. Komponen piranti lunak adalah modul berisi code, baik berisi source code maupun binary code, baik library maupun executable, baik yang muncul pada compile time, link time, maupun run time. Umumnya komponen terbentuk dari beberapa class dan/atau package, tapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa interface, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain.

Contoh component diagram:



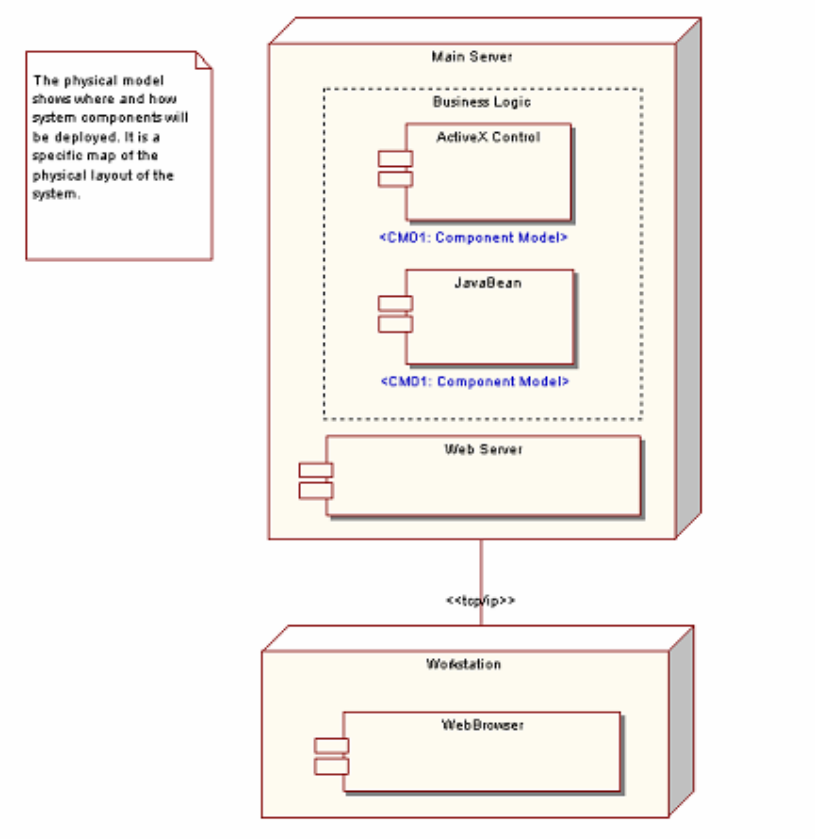
Gambar 2.8 Component Diagram

Sumber : http://ikc.cbn.net.id/umum/yanti-uml.php

H. Deployment Diagram

Deployment/physical diagram menggambarkan detail bagaimana komponen di-deploy dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisikal . Sebuah node adalah server, workstation, atau piranti keras lain yang digunakan untuk men-deploy komponen dalam lingkungan sebenarnya. Hubungan antar node (misalnya TCP/IP) dan requirement dapat juga didefinisikan dalam diagram ini.

Contoh deployment diagram :



Gambar 2.9 Deployment Diagram

Sumber : http://ikc.cbn.net.id/umum/yanti-uml.php

Tool Yang Mendukung UML

Saat ini banyak sekali tool pendesainan yang mendukung UML, baik itu tool komersial maupun opensource. Beberapa diantaranya adalah:

1. Rational Rose (www.rational.com)
2. Together (www.togethersoft.com)
3. Object Domain (www.objectdomain.com)
4. Jvision (www.object-insight.com)

5. Objecteering (www.objecteering.com)
6. MagicDraw (www.nomagic.com/magicdrawuml)
7. Visual Object Modeller (www.visualobject.com)

2.6 Bahasa Pemprograman Yang di Gunakan

Bahasa pemprograman yang digunakan dalam penulisan skripsi ini adalah PHP, HTML, Mysql.

2.6.1 PHP (personal home page)

PHP merupakan bahasa pemprograman server side yang di desain khusus untuk aplikasi web yang di tambahkan kedalam HTML . Sifat server side berarti pengerjaan skrip akan dilakukan di server ,baru kemudian hasilnya dikirim ke browser.

Kelebihan PHP dari bahasa pemrograman lain

(http://id.wikipedia.org/wiki/*php*/ 29 maret 2009**)**

a) Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaanya.
b) Web Server yang mendukung PHP dapat ditemukan dimana - mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan configurasi yang relatif mudah.
c) Dalam sisi pengembangan lebih mudah, karena banyaknya milis - milis dan developer yang siap membantu dalam pengembangan.
d) Dalam sisi pemahamanan, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.

e) PHP adalah bahasa open source yang dapat digunakan di berbagai mesin (Linux, Unix, Macintosh, Windows) dan dapat dijalankan secara runtime melalui console serta juga dapat menjalankan perintah-perintah sistem.

f) Skrip tidak dapat di intip dengan menggunakan fasilitas view HTML source.

PHP dirancang sedemikian khusus untuk membentuk web dinamis artinya dapat membentuk suatu tampilan berdasarkan permintaan terkini.

Pada saat ini PHP cukup popular sebagai peranti pemprograman web karena dapat berfungsi pada server-server yang berbasis unix,linux,windows dan mancitosh dan untuk mencoba kita tidak perlu menggunakan computer berkelas server ,cukup dengan menggunakan computer biasa kita bisa mempelajari dan mempraktekkan PHP.

9 Tipe data yang di miliki PHP :

1. Integer
2. Double
3. Boolean
4. String
5. Object
6. Array
7. Null
8. Nill
9. Resource

Program HELLO WORLD yang dituliskan PHP adalah sebagai berikut :

```
<?php
echo "<b>Hello World</b>" ;
echo "<b>Halo Dunia</b>" ;
?>
```

2.6.2 HTML(Hypertext Markup Languange) Dan CSS (Cascading Style Sheet)

(http://id.wikipedia.org/wiki/*HTML* / 29 maret 2009)

HyperText Markup Language (HTML) adalah sebuah *bahasa markup* yang digunakan untuk membuat sebuah halaman web dan menampilkan berbagai informasi di dalam sebuah browser Internet. Bermula dari sebuah bahasa yang sebelumnya banyak digunakan di dunia penerbitan dan percetakan yang disebut dengan SGML *(Standard Generalized Markup Language)*, HTML adalah sebuah standar yang digunakan secara luas untuk menampilkan halaman web. HTML saat ini merupakan standar Internet yang didefinisikan dan dikendalikan penggunaannya oleh World Wide Web Consortium (W3C).

HTML berupa kode-kode tag yang menginstruksikan browser untuk menghasilkan tampilan sesuai dengan yang diinginkan. Sebuah file yang merupakan file HTML dapat dibuka dengan menggunakan browser web seperti Mozilla Firefox atau Microsoft Internet Explorer. HTML juga dapat dikenali oleh aplikasi pembuka email ataupun dari PDA dan program lain yang memiliki kemampuan browser.

HTML dokumen tersebut mirip dengan dokumen teks biasa, hanya dalam dokumen ini sebuah teks bisa memuat instruksi yang ditandai dengan kode atau lebih dikenal dengan TAG tertentu. Sebagai contoh jika ingin membuat teks ditampilkan menjadi tebal seperti: **TAMPIL TEBAL**, maka penulisannya dilakukan dengan cara: TAMPIL TEBAL. Tanda digunakan untuk mengaktifkan instruksi cetak tebal, diikuti oleh teks yang ingin ditebalkan, dan diakhiri dengan tanda untuk menonaktifkan cetak tebal tersebut.

Secara garis besar, terdapat 4 jenis elemen dari HTML:

1. *Structural*. tanda yang menentukan level atau tingkatan dari sebuah teks (contoh, <h1>Golf</h1> akan memerintahkan browser untuk menampilkan "Golf" sebagai teks tebal besar yang menunjukkan sebagai Heading 1.
2. *Presentational*. tanda yang menentukan tampilan dari sebuah teks tidak peduli dengan level dari teks tersebut (contoh, boldface akan menampilkan **bold**. Tanda presentational saat ini sudah mulai digantikan oleh CSS dan tidak direkomendasikan untuk mengatur tampilan teks.
3. *Hypertext*. tanda yang menunjukkan pranala ke bagian dari dokumen tersebut atau pranala ke dokumen lain (contoh, Wikipedia akan menampilkan Wikipedia sebagai sebuah hyperlink ke URL tertentu).
4. Elemen *widget* yang membuat objek-objek lain seperti tombol (<button>), list (<li>), dan garis horizontal (<hr>).

Selain markup *presentational* , markup yang lin tidak menentukan bagaimana tampilan dari sebuah teks. Namun untuk saat ini, penggunaan tag HTML untuk menentukan tampilan telah dianjurkan untuk mulai ditinggalkan dan sebagai gantinya digunakan *Cascading Style Sheets.*

Contoh dokumen HTML sederhana

```
<!DOCTYPE html>
<html>
<head>
<title>Halo HTML</title>
</head>
<body>
<p>Halo Dunia!</p>
</body>
</html>
```

Yang bisa dilakukan dengan HTML yaitu :

1. Mengontrol tampilan dari web dan content nya
2. Mempublikasikan dokumen secara online sehingga bisa di akses dari seluruh dunia.
3. Membuat online form yang bisa digunakan untuk menangani pendaftaran,transaksi secara online.
4. Menambahkan objek2 seperti image,audi,video dan juga java applet dalam dokumen dan HTML.

Sedangkan CSS merupakan feature yang sangat penting dalam dynamic HTML. Meskipun bukan merupakan suatu keharusan dalam membuat web,akan tetapi penggunaan style sheet merupakan kelebihan tersendiri.

Suatu style sheet merupakan tempat dimana dapat mengontrol dan memanage style-style yang ada. Style sheet mendeskripsikan bagaimana tampilan document HTML yang menggunakannya,juga bisa membuat efek-efek spesial di web anda dengan menggunakan style sheet. Sebagai contoh anda bisa membuat style sheet yang mendefenisikan style untuk <H1> dengan style bold dan italic dan berwarna biru. Atau pada tag <P> yang akan di tampilkan dengan warna biru dan menggunakan font gothic dan masih banyak lagi yang bisa dilakukan dengan style sheet. Secara teoritis bisa menggunakan style sheet technologi denga HTML. Akan tetapi pada prakteknya hanya Cascading Style Sheet (CSS) tehnology yang support pada hampir semua web browser .karena css telah di standarakan oleh World Wide Web consortium(w3c) untuk digunakan di web browser.

2.6.3 Java Script

(http://id.wikipedia.org/wiki/*JavaScript* / 29 maret 2009**)**

JavaScript adalah nama implementasi Netscape Communications Corporation untuk ECMAScript standar, suatu bahasa skrip yang didasarkan pada konsep pemrograman berbasis prototipe. Bahasa ini terutama terkenal karena penggunaannya di situs web (sebagai JavaScript sisi klien) dan juga digunakan untuk menyediakan akses skrip untuk objek yang dibenamkan (*embedded*) di aplikasi lain.

Walaupun memiliki nama serupa, JavaScript hanya sedikit berhubungan dengan bahasa pemrograman Java, dengan kesamaan utamanya adalah penggunaan sintaks C. Secara semantik, JavaScript memiliki lebih banyak kesamaan dengan bahasa pemrograman Self.

Skrip JavaScript yang dimasukkan di dalam berkas HTML ataupun XHTML harus dimasukkan di antara tag <script>...</script>. Berikut ini adalah contohnya (yang akan menampilkan sebuah *dialog box* berisi Halo Dunia! ketika sebuah tombol diklik oleh pengguna) :

```
<input type="button" value="Tekan di sini" onclick="halo();">
<script type="text/javascript">
 function halo() {
 alert( "Halo Dunia!" );
  }
</script>
```

2.6.4 Ajax

Asynchronous JavaScript And XML, atau disingkat AJAX, adalah suatu teknik pemrograman berbasis web untuk menciptakan aplikasi web interaktif. Tujuannya adalah untuk memindahkan sebagian besar interaksi pada komputer web surfer, melakukan pertukaran data dengan server di belakang layar, sehingga halaman web tidak harus dibaca ulang secara keseluruhan setiap kali seorang pengguna melakukan perubahan. Hal ini akan meningkatkan interaktivitas, kecepatan, dan usability. Ajax merupakan kombinasi dari:

1. XHTML (atau HTML) dan CSS untuk bahasa mark up dan tampilan.
2. DOM yang diakses dengan client side scripting language, khususnya implementasi ECMAScript JavaScript dan JScript, untuk menampilkan secara dinamis dan berinteraksi dengan informasi yang ditampilkan seperti
3. Objek XMLHttpRequest untuk melakukan pertukaran data asinkronus dengan web server. Pada beberapa framework dan kasus Ajax, objek IFrame lebih dipilih daripada XMLHttpRequest untuk melakukan pertukaran data dengan web server.
4. XML umumnya digunakan sebagai format untuk pengiriman data, walaupun format lain juga memungkinkan, seperti HTML, plain text, JSON dan EBML. Seperti halnya DHTML, LAMP, atau SPA, Ajax bukanlah teknologi spesifik, melainkan merupakan gabungan dari teknologi yang dipakai bersamaan. Bahkan, teknologi turunan/komposit yang berdasarkan Ajax, seperti AFLAX sudah mulai bermunculan

2.7 Perangkat Lunak Yang Digunakan

Database MySQL

http://id.wikipedia.org/wiki/*MySQL*/ 29 maret 2009

MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (bahasa Inggris: *database management system*) atau DBMS yang

multithread, multi-user, dengan sekitar 6 juta instalasi di seluruh dunia. MySQL AB membuat MySQL tersedia sebagai perangkat lunak gratis dibawah lisensi GNU *General Public License* (GPL), tetapi mereka juga menjual dibawah lisensi komersial untuk kasus-kasus dimana penggunaannya tidak cocok dengan penggunaan GPL Tidak sama dengan proyek-proyek seperti Apache, dimana perangkat lunak dikembangkan oleh komunitas umum, dan hak cipta untuk kode sumber dimiliki oleh penulisnya masing-masing, MySQL dimiliki dan disponsori oleh sebuah perusahaan komersial Swedia MySQL AB, dimana memegang hak cipta hampir atas semua kode sumbernya. Kedua orang Swedia dan satu orang Finlandia yang mendirikan MySQL AB adalah: David Axmark, Allan Larsson, dan Michael "Monty" Widenius.

MySQL adalah Relational Database Management System (RDBMS) yang didistribusikan secara gratis dibawah lisensi GPL (General Public License). Dimana setiap orang bebas untuk menggunakan MySQL, namun tidak boleh dijadikan produk turunan yang bersifat closed source atau komersial. MySQL sebenarnya merupakan turunan salah satu konsep utama dalam database sejak lama, yaitu SQL (Structured Query Language). SQL adalah sebuah konsep pengoperasian database, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis. Keandalan suatu sistem database (DBMS) dapat diketahui dari cara kerja optimizer-nya dalam melakukan proses perintah-perintah SQL, yang dibuat oleh user maupun program-program aplikasinya.

Sebagai database server, MySQL dapat dikatakan lebih unggul dibandingkan database server lainnya dalam query data. Hal ini terbukti untuk query yang dilakukan oleh single user, kecepatan query MySQL bisa sepuluh kali lebih cepat dari PostgreSQL dan lima kali lebih cepat dibandingkan Interbase. Selain itu MySQL juga memiliki beberapa keistimewaan, antara lain :

1. Portability

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

2. Open Source

   MySQL didistribusikan secara open source (gratis), dibawah lisensi GPL sehingga dapat digunakan secara cuma-cuma.

3. MultiuserMySQL

   Dapat digunakan oleh beberapa user 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. Column types

   MySQL memiliki tipe kolom yang sangat kompleks, seperti signed / unsigned integer, float, double, char, text, date, timestamp, dan lain-lain.

6. Command dan functions

MySQL memiliki operator dan fungsi secara penuh yang mendukung perintah Select dan Where dalam query.

7. Security

MySQL memiliki beberapa lapisan sekuritas seperti level subnetmask, nama host, dan izin akses user dengan sistem perizinan yang mendetail serta password terenkripsi.

8. Scalability dan limits

MySQL mampu menangani database dalam skala besar, dengan jumlah 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. Connectivity

MySQL dapat melakukan koneksi dengan client menggunakan protokol TCP/IP, Unix soket (UNIX), atau Named Pipes (NT).

10. Localisation

MySQL dapat mendeteksi pesan kesalahan pada client dengan menggunakan lebih dari dua puluh bahasa. Meskipun demikian, bahasa Indonesia belum termasuk didalamnya.

11. Interface

MySQL memiliki interface (antar muka) terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).

12. Clients dan tools

    MySQL dilengkapi dengan berbagai tool yang dapat digunakan untuk administrasi database, dan pada setiap tool yang ada disertakan petunjuk online.

13. Struktur tabel

    MySQL memiliki struktur tabel yang lebih fleksibel dalam menangani ALTER TABLE, dibandingkan database lainnya semacam PostgreSQL ataupun Oracle.

2.8 Internet

http://www.total.or.id/info.php?kk=*internet/* 29maret 2009

Singkatan dari Interconnection Networking. The network of the networks. Diartikan sebagai a global network of computer networks atau sebuah jaringan komputer dalam skala global/mendunia. Jaringan komputer ini berskala internasional yang dapat membuat masing-masing komputer saling berkomunikasi. Network ini membentuk jaringan inter-koneksi (Inter-connected network) yang terhubung melalui protokol TCP/IP. Dikembangkan dan diuji coba pertama kali pada tahun 1969 oleh US Department of Defense dalam proyek ARPAnet. dalam arti akan membentuk suatu.

2.9 WWW(World Wide Web)

http://ibliz-kecil.blogspot.com/2008/08/*pengertian-www*.html/ 29 maret 2009

WWW (World Wide Web) adalah sebuah bagian dari internet yang sangat dikenal dalam dunia internet, dengan adanya WWW seorang dapat menampilkan sebuah halaman virtual yang disebut website.

Jika dilihat dari proses kerjanya WWW dapat dibagi menjadi beberapa komponen yaitu :

1. Protocol
2. Address
3. HTML (Hypertext Markup Language)
4. HTTP (Hypertext Transfer Protocol)
5. FTP (File Transfer Protocol)
7. Mailto
8. News
9. Telnet (Tele Networking)

Secara teknis, web adalah sebuah sistem dimana informasi dalam bentuk teks, gambar, suara, dan lain-lain yang tersimpan dalam sebuah *internet webserver* dipresentasikan dalam bentuk *hypertext*. Informasi di web dalam bentuk teks umumnya ditulis dalam format HTML (*Hypertext Markup Language*). Informasi lainnya disajikan dalam bentuk *grafis* (dalam format GIF, JPG, PNG), *suara* (dalam format AU, WAV), dan objek multimedia lainnya (seperti MIDI, Shockwave, Quicktime Movie, 3D World). Web dapat diakses oleh perangkat lunak *web client* yang secara populer disebut sebagai *browser*. *Browser* membaca halaman-halaman

web yang tersimpan dalam *webserver* melalui protokol yang disebut HTTP (*Hypertext Transfer Protocol*). Dewasa ini, tersedia beragam perangkat lunak *browser*. Beberapa diantaranya cukup populer dan digunakan secara meluas, contohnya seperti Microsoft Internet Explorer, Netscape Navigator, maupun Opera, namun ada juga beberapa produk browser yang kurang dikenal dan hanya digunakan di lingkungan yang terbatas. Sebagai dokumen *hypertext*, dokumen-dokumen di web dapat memiliki *link (sambungan)* dengan dokumen lain, baik yang tersimpan dalam *webserver* yang sama maupun di *webserver* lainnya. Link memudahkan para pengakses web berpindah dari satu halaman ke halaman lainnya, dan "berkelana" dari satu server ke server lain. Kegiatan penelusuran halaman web ini biasa diistilahkan sebagai *browsing*, ada juga yang menyebutnya sebagai *surfing* (berselancar).

http://this-live.blogspot.com/2008/11/*pengertian-www*.html/ 29 maret 2009

This document was created with Win2PDF available at http://www.win2pdf.com.
The unregistered version of Win2PDF is for evaluation or non-commercial use only.
This page will not be added after purchasing Win2PDF.