BAB II

LANDASAN TEORI

2.1 Konsep Dasar Sistem

Pada konsep dasar sistem akan membahasa mengenai pengertian dari sistem, yaitu;

2.1.1 Definisi sistem

Suatu sistem terdiri dari sejumlah komponen yang berinteraksi artinya saling bekerja sama membentuk suatu kesatuan. Beberapa para ahli mengemukakan pegertian sistem seperti dibawah ini :

Menurut Edhy Sutanta (2003:4) sistem merupakan sekumpulan hal atau kegiatan atau elemen atau subsistem yang saling berkejasama atau yang di hubungkan dengan cara-cara tertentu sehingga membentuk satu kesatuan untuk melaksanakan suatu fungsi guna mencapai suatu tujuan.

Menurut Raymond (2001:11) sistem merupakan sekelompok elemen yang terintegrasi dengan maksud yang sama untuk mencapaitujuan tertentu.

Berdasarkan beberapa pendapat yang telah di kemukakan diatas dapat ditarik kesimpulan bahwa “Sistem adalah kumpulan elemen atau bagian-bagian atau komponen-komponen atau subsistem-subsistem yang terintegrasi satu sama lain dan bekerja secara harmonis untuk mencapai tujuan tertentu atau maksud tertentu”.

2.2 Konsep Dasar Informasi

Dalam konsep dasar informasi akan membahas menganai pengertian dari data dan informasi, yaitu;

2.2.1 Definisi Data

Sumber informasi adalah data. Data merupakan bentuk jamak dari bentuk tunggal datum atau data item.

Menurut Humadiana dan Evi (2006:20) data merupakan fakta-fakta, simbol/karakter, data mentah, atau observasi yang menggambarkan suatu fenomena tertentu.

Sedangkan menurut Edhy Sutanta (2003:9) data merupakan bahan keterangan tentang kejadian-kejadian nyata atau fakta-fakta yang dirumuskan dalam sekelompok lambing tertentu yang tidak acak yang menunjukan jumlah, tindakan, atau hal.

Dari beberapa pandangan yang telah dipaparkan diatas dapat disimpulkan bahwa “Data merupakan kejadian nyata atau fakta atau bilangan atau karakter yang belum memiliki makna”.

2.2.2 Definisi Informasi

Informasi ibarat darah yang mangalir di dalam tubuh suatu organisasi, sehingga informasi ini sangat penting di dalam suatu organisasi. Informasi (*information*) dapat didefinisikan sebagai berikut:

Menurut Humadiana dan Evi (2006:19) informasi merupakan data yang telah diproses sedemikian rupa atau disampaikan dalam model yang memiliki lebih banyak makna.

Menurut Edhy Sutanta (2003:10) informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang.

Menurut Raymond (2001:15) informasi merupakan data yang telah diproses atau yang memiliki arti.

Berdasarkan beberapa pendapat yang dikemukan diatas dapat ditarik kesimpulan bahwa "Informasi adalah data yang sudah diolah, dibentuk, atau dimanipulasi sesuai dengan keperluantertentu dan memiliki makna yang berguna untuk peramalan atau dalam pengambilan keputusan".

2.2.3 Kualitas Informasi

Kualitas Informasi (*quality of information*) Menurut Prabu (2006:20) diantaranya ditentukan oleh beberapa hal, yaitu:

1. Relevan artinya suatu informasi harus bermanfaat bagi pengguna nya dalam waktu pendek ataupun dalam waktu panjang.
2. Akurat artinya suatu informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan bagi si pemakai serta harus dapat mencerminkan dengan jelas maksud dari informasi tersebut. Ketidak akuratan data terjadi karena sumber dari informasi tersebut mengalami gangguan dalam penyampaiannya baik hal itu dilakukan secara sengaja maupun tidak sehingga menyebabkan data asli tersebut berubah atau rusak.

Menurut Kusrini S.Kom dan Andri Koniyo (2007 : 8), Informasi yang berkualitas memiliki 3 kriteria, yaitu :

1. Akurat (*Accurate*)

   Informasi harus bebas dari kesalahan, tidak bias ataupun menyesatkan. Akurat juga berarti bahwa informasi itu harus dapt dengan jelas mencerminkan maksudnya.

2. Tepat pada waktunya (*Timeliness*)

   Informasi yang datang pada penerima tidak boleh terlambat. Didalam pengambilan keputusan, informasi yang sudah using tidak lagi bernilai. Bila informasi datang terlambat sehingga pengambilan keputusan terlambat dilakukan, hal itu dapat berakibat fatal bagi perusahaan.

3. Relevan (*Relevance*)

   Informasi yang disampaikan harus mempunyai keterkaitan dengan masalah yang akan dibahas dengan informasi tersebut. Informasi harus bermanfaat bagi pemakainya. Disamping karakteristik, nilai informasi juga ikut menentukan kualitasnya. Nilai informasi (*value of information*) ditentukan oleh dua hal, yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih besar disbanding biaya untuk mendapatkannya.

Berdasarkan beberapa pendapat yang dikemukan diatas dapat ditarik kesimpulan bahwa “informasi yang berkualitas harus memiliki kriteria akurat, tepat waktu dan relevan”.

2.2.4 Nilai Informasi

Menurut Tata (2005:30), nilai informasi ditentukan oleh 2 (dua) hal, yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai bila manfaat lebih efektif dengan biaya mendapatkannya. Pengukuran nilai informasi biasanya dihubungkan dengan analisis *cost effectiveness* atau *cost benefit*.

Menurut Jogiyanto (2005:11) Nilai dari informasi (*Value of Information*) ditentukan dari dua hal, yaitu manfaat dan biaya mendapatkannya.

Berdasarkan beberapa pendapat yang dikemukan diatas dapat ditarik kesimpulan bahwa ”Nilai suatu informasi ditentukan oleh mafaat dan biaya untuk mendapatkannya serta sempuna apabila dapat dinyatakan dengan jelas”.

2.3 Konsep Dasar Sistem Informasi

Dalam konsep dasar sistem akan membahsa menganai pengertian dari informasi dan data serta siklus informasi, yaitu;

2.3.1 Pengertian Informasi

Menurut Azhar Susanto (2004: 21) Informasi adalah hasil pengolahan data yang memberikan arti dan manfaat. Dari pengertian informasi ini, ada tiga hal yang harus diperhatikan yaitu:

1. Informasi merupakan hasil pengolahan data.
2. Memberikan makna atau arti.
3. Berguna dan bermanfaat.

Sedangkan menurut Jogiyanto HM (2005:8) Informasi dapat didefinisikan sebagai berikut: "Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya".

Berdasarkan beberapa pendapat yang dikemukan diatas dapat ditarik kesimpulan bahwa "Informasi adalah hasil dari pengelolaan data yang dapat memberi arti dan manfaat bagi yang menerimanya".

2.3.2 Pengertian Data

Menurut Azhar Susanto (2004:37) mendefinisikan data sebagai berikut: "Data adalah fakta atau apapun yang dapat didigunakan sebagai input untuk menghasilkan informasi". Maka data adalah suatu keterangan tertulis mengenai fakta atau kenyataan yang masih berdiri sendiri, belum mempunyai pengertian sebagai kelompok, belum terkoordinasi satu sama lain dan belum diolah sesuai dengan keperluan tertentu menjadi suatu bentuk yang sesuai dengan keperluan penggguna informasi yang bersangkutan.

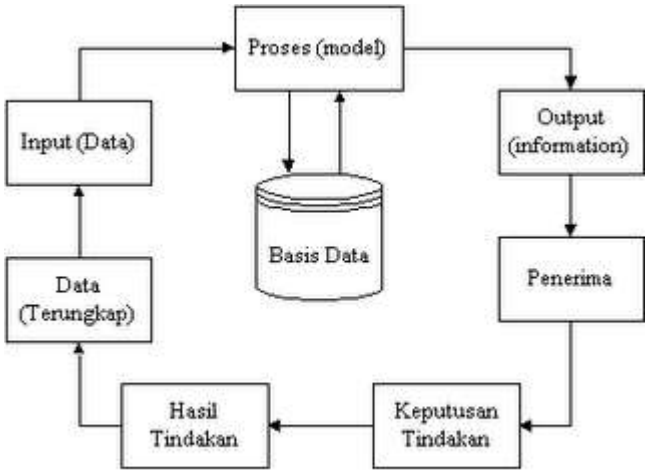
Sedangkan menurut Kristanto (2004:4) data adalah "Sesuatu yang nyata, fakta mengenai objek yang dapat mengurangi derajat ketidakpastian tentang suatu keadaan atau kejadian".

Berdasarkan beberapa pendapat yang dikemukan diatas dapat ditarik kesimpulan bahwa "Data adalah suatu kejadian atau fakta yang dapat menghasilkan suatu informasi".

2.3.3 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 II.1 Siklus Informasi

Sumber: Edhy Sutanta "*Sistem Informasi Manajemen".* Th 2003 hal : 32

2.4 Pengertian Sistem Informasi

Dalam sistem infomasi akan membahsa menganai pengertian dari sistem informasi, yaitu sebagai berikut;

2.4.1 Definisi Sistem Informasi

Terdapat berbagai macam pengertian Sistem Informasi menurut beberapa hal, diantaranya:

Menurut Dr. Azhar Susanto (2004: 55) "Sistem informasi adalah kumpulan dari sub-sub sistem baik phisik maupun non phisik yang saling

berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai tujuan, yaitu mengolah data menjadi informasi yang berguna".

Menurut Jogiyanto (2005 : 697) Sistem informasi dapat didefinisikan sebagai suatu sistem di dalam suatu organisasi yang merupakan kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur dan pengendalian yang di tunjukan untuk mendapatkan jalur komunikasi penting, memperoses tipe transaksi rutin tertentu, member sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar informasi untuk pengambil keputusan cerdik.

Dari definisi – definisi di atas dapat disimpulkan bahwa 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 satu tujuan yaitu mengolah data menjadi informasi yang berguna dalam pengambilan keputusan.

2.4.2 Komponen Sistem Informasi

Sistem informasi mempunyai enam buah komponen atau disebut juga dengan blok bangunan. Blok tersebut masing-masing saling berinteraksi untuk membentuk satu kesatuan untukmencapai sasaran. Komponen/blok tersebut yaitu:

1. Komponen Masukan

   Input mewakili data yang masuk kedalam sistem Informasi.

2. Komponen Model

Terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Komponen *Output* (komponen Keluaran)

   Produk atau hasil akhir dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Komponen Teknologi

   Teknologi merupakan alat dalam sistem informasi, teknologi digunakan untuk menerima input, menjalankan model, menyimpan,dan mengakses data, menghasilkan dan mengirimkan output dan membantu pengendalian sistem.

5. Komponen Basis Bata

   Basis data merupakan kumpulan data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya.

6. Komponen Kontrol

   Komponen kontrol merupakan pengendalian yang dirancang untuk menanggulangi gangguan terhadap sistem informasi.

Dari keenam komponen diatas harus ada bersama-sama dan membentuk satu kesatuan. Jika satu atau lebih komponen tersebut tidak ada maka sistem

informasi tidak bisa menjalankan fungsinya, yaitu pengolahan data yang menghasilkan informasi yang akurat tepat waktu dan relevan.

2.5 Sistem Basis Data

Menurut A.S & Shalahuddin (2011: 44) Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat. Kebutuhan basis data dalam sistem informasi meliputi:

1. Memasukkan, menyimpan, dan mengambil data.
2. Membuat laporan berdasarkan data yang telah disimpan.

Sedangkan menurut Fathansyah, Ir (2002:9) Basis data adalah sistem basis data merupakan sistem yang terdiri dari atas sekumpulan tabel yang saling berhubungan (dalam sebuah basis data di sebuah sistem komputer) dan sekumpulan program (DBMS) yang menungkinkan beberapa pemakai lain untuk mengakses dan memanipulasi table-tabel tersebut.

Menurut Fathansyah, Ir (2002:9), Sistem Basis data pada dasarnya di buat untuk tujuan sebagai berikut:

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, yang 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 semakin 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
8. 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:

1. Mereduksi redudansi
2. Data dapat di share antar aplikasi
3. Standarisasi data dapat di lakukan
4. Batasan security dapat diterapkan
5. Mengelola integritas (keterjaminan akurasi) data
6. Menyeimbangkan kebutuhan yang saling konflik
7. Independesi data (obyektif DBS)

Struktur Sistem Database:

1. File manager: mengelola spacedan struktur data.
2. Database manager: menyediakan antar muka dengan data fisik
3. Query processor: menterjemahkan query ke instruksi yang dimengerti database manager
4. DML precompiler: mengkonversi perintah DML menjadicalls normal.

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

Berdasarkan beberapa pendapat yang dikemukan diatas dapat ditarik kesimpulan bahwa "Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan yang saling berhubungan (dalam sebuah basis data di sebuah sistem komputer) dan sekumpulan program (DBMS) yang menungkinkan beberapa pemakai lain untuk mengakses dan memanipulasi table-tabel tersebut".

2.6 Penjadwalan

Menurut oleh Roger G Schroeder (2000:20) penjadwalan diartikan sebagai petunjuk atau indikasi apa saja yang harus dilakukan, dengan siapa, dan dengan peralatan apa yang di gunakan untuk menyelesaikan suatu pekerjaan pada waktu tertentu.

2.7 Skripsi

Pengertian skripsi menurut KBBI (2003:820) skripsi merupakan laporan pekerjaan lapangan dan membaca buku-buku untuk membentuk konsep baru yang meliputi fakta serta mengembangkan hipotesis antara variabel-variabel yang dijabarkan konsep tersebut.

2.8 Seminar

Menurut KBBI (2003:810) Seminar adalah pertemuan atau persidangan untuk membahas suatu masalah dibawah pimpinan ketua sidang yang biasanya guru besar, ahli dan sebagainya.

2.9 Alat Bantu dalam Perancangan Sistem

Alat bantu dalam perancangan sistem merupakan pemodelan data, mendeskripsikan data yang terlibat dalam perangkat lunak. Adapun alat bantu dalam perancangan sistem berupa *UML, Usecase Diagram, Class Diagram, Relation Class Diagram, Statechart Diagram, Diagram Activity Statechart, dan Class Ikon Diagram.*

2.9.1 Definisi Unified Modeling Language (UML)

Menurut Adi Nugroho (2005:3) "*Unified Modeling Language* (UML) adalah alat bantu analis serta perancangan perangkat lunak berbasis objek".

Menurut Martin Fowler (2005:1) "*Unified Modeling Language* (UML) adalah keluarga notasi grafis yang didukunng oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek".

Berdasarkan beberapa pendapat yang dikemukakan dapat ditarik kesimpulan bahwa "*Unified Modeling Language* (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*)".

2.9.2 Penggunaan Unified Modeling Language (UML)

Menurut Afif Amrulla (2002:20) “Langkah-langkah penggunaan *Unified Modeling Language* (UML), sebagai berikut:

1. Buatlah daftar business process dari level tertinggi untuk mendefinisikan aktivitas dan proses yang mungkin muncul.
2. Pendekatan usecase untuk setiap business process untuk mendefinisikan dengan tepat fungsional yang harus disediakan oleh system, kemudian perhalus usecase diagram dan dilengkapi dengan requirement, constraintsdan catatan lain.
3. Buatlah deployment diagram secara kasar untuk mendefinisikan arsitektur fisik sistem.
4. Definisikan requirementlain non fungsonal, securitydan sebagai yang juga harus disediakan oleh system.
5. Berdasarkan usecase diagram, mulailah membuat activity diagram.
6. Defisinikan objek-objek level atas package atau domain dan buatlah sequencedan/atau collaborationuntuk tiap alir pekerjaan, jika sebuah use casememungkinkan alir normal dan error, buat lagi satu diagram untuk masing-masing alir.
7. Buatlah rancangan user interface modelyang menyediakan antarmuka bagi pengguna untuk menjalankan skenario use case.
8. Berdasarkan model-model yang sudah ada, buatlah class diagram. Setiap package atau domain dipecah menjadi hirarki class lengkap dengan atributdan metodenya. Akan lebih baik jika untuk setiap class

dibuat unit test untuk menguji fungsionalitas class dan interaksi dengan classlain.

9. Setelah class diagram itu dibuat, kita dapat melihat kemungkinan pengelompokkan class menjadi komponen-komponen karena itu buatlah component diagram pada tahap ini. Juga, definisikan test integrasiuntuk setiap komponen meyakinkan ia bereaksi dengan baik.
10. Perhalus deployment diagram yang sudah dibuat. Detailkan kemampuan dan requirement piranti lunak, sistem operasi, jaringan dan sebagainya. Petakan komponen ke dalam node.
11. Memulailah membangun system. Ada dua pendekatan yang tepat digunakan:
    a. Pendekatan use case dengan mengassignsetiap usecase kepada tim pengembang tertentu untuk mengembangkan unit kode yang lengkap dengan test.
    b. Pendekatan komponenyaitu mengassign setiap komponen kepada tim pengembang tertentu.
12. Lakukan uji modul dan uj integrasi serta perbaiki model beserta codenya. Model harus selalu sesuai dengan codeyang aktual.
13. Perangkat lunak siap dirilis.

2.9.3 Fokus Unified Modeling Language (UML)

Menurut Adi Nugroho (2005:25) “Dalam kerangka spesifikasi, Unified Modeling Languag (UML) menyediakan model-model yang tepat, tidak mendua arti (ambigu) serta lengkap. Secara khusus *Unified Modeling*

Language (UML) menspesifikasikan langkah-langkah penting dalam pengambilan keputusan analisis, perancangan serta implementasi dalam sistem yang sangat bernuansa perangkat lunak(software intensive system). Dalam hal ini, *Unified Modeling Language* (UML) bukanlah merupakan bahasa pemrograman tetapi model-model yang tercipta berhubungan langsung dengan berbagai macam bahasa pemrograman, sehingga adalah mungkin melakukan pemetaan (mapping) langsung dari model-model yang dibuat dengan *Unified Modeling Language* (UML) dengan bahasa-bahasa pemrograman berorientasi objek, sepert *Java, Borland Delphi, Visual Basic, C++, PHP* yang menggunakan *framework*, dan lain-lain.

Pemetaan (mapping) *Unified Modeling Language* (UML) bersifat dua arah, yaitu:

a. Generasi kode bahasa pemrograman tertentu dari *Unified Modeling Language*(UML) forward engineering.
b. Generasi kode belum sesuai dengan kebutuhandan harapan pengguna, pengembang dapat melakukan langkah balik bersifat iterative dari implementasi ke *Unified Modeling Language*(UML) hingga didapat system/peranti lunak yang sesuai dengan harapan pengguna dan pengembang".

2.9.4 Bangunan Dasar Metodologi *Unified Modeling Language*(UML)

Menurut Adi Nugroho (2005:28). "Bangunan dasar metodologi *Unified Modeling Language* (UML) menggunakan tiga banguna dasar untuk mendeskripsikan sistem/perangkat lunak yang akan dikembangkan yaitu:

1. Sesuatu (things) Ada 4 (empat) things dalam *Unified Modeling Language* (UML), yaitu:
   a. Struktur things

      Merupakan bagian yang relatif statis dalam model *Unified Modeling Language* (UML). Bagian yang relatif statis dapat berupa elemen-elemen yang bersifat fisik maupun konseptual.

   b. Behavioral things

      Merupakan bagian yang dinamis pada model *Unified Modeling Language* (UML), biasanya merupakan kata kerja dari model *Unified Modeling Language* (UML), yang mencerminkan perilaku sepanjang ruang dan waktu.

   c. Grouping things

      Merupakan bagian pengorganisasian dalam *Unified Modeling Language* (UML). Dalam penggambaran model yang rumit kadang diperlukan penggambaran paket yang menyederhanakan model. Paket-paket ini kemudian dapat didekomposisi lebih lanjut. Paket berguna bagi pengelompokan sesuatu, misalnya model-model dan subsistem-subsistem.

   d. Annotational things

      Merupakan bagian yang memperjelaskan model *Unified Modeling Language* (UML) dan dapat berupa komentar-komentar yang menjelaskan fungsi serta cirri-ciri setiap element dalam model *Unified Modeling Language* (UML).

2. Relasi (*Relationship*)

   Ada 4 (empat) macam relationship dalam *Unified Modeling Language* (UML), yaitu:

   a. Ketergantungan

      Merupakan hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*indenpendent*) akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri (*indenpendent*).

   b. Asosiasi

      Merupakan apa yang menghubungkan antara objek satu dengan objek yang lainnya, bagaimana hubungan suatu objek dengan objek lainnya. Suatu bentuk asosiasi adalah agregasi yang menampilkan hubungan suatu objek dengan bagian-bagiannya.

   c. Generalisasi

      Merupakan hubungan dimana objek anak (*descendent*) berbagi perilaku dan struktur data dari objek yang ada diatasnya objek induk (ancestor). Arah dari atas kebawah dari objek induk ke objek anak dinamakan spesialisasi, sedangkan arah berlawanan sebaliknya dari arah bawah keatas dinamakan generalisasi.

   d. Realisasi

      Merupakan operasi yang benar-benar dilakukan oleh suatu objek.

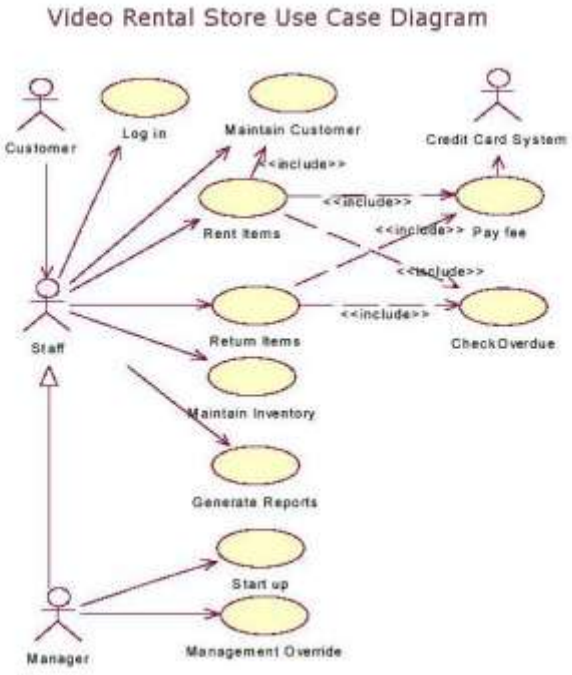
3. Diagram

   Berikut adalah macam diagram dalam *Unified Modeling Language* (UML), yaitu

a. Use Case Diagram

Diagram ini memperlihatkan himpunan use case dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta di harapkan pengguna.

Contoh Usecase diagram:



Gambar II.2 Use Case diagram

Sumber: Sholiq "*Pemodelan Sistem Informasi Berorientasi Objek dengan UML*" thn 2006. Hal 8

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

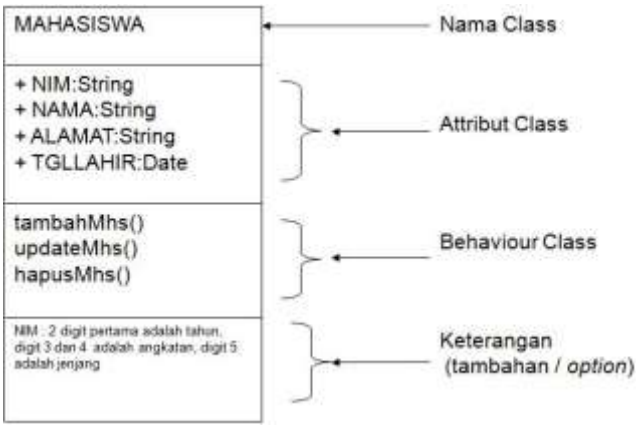
memanipulasi keadaan tersebut (metoda/fungsi). Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain.

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 classyang bersangkutan
b) Protected, hanya dapat dipanggil oleh classyang bersangkutan dan anak-anak yang mewarisinya
c) Public, dapat dipanggil oleh siapa saja.

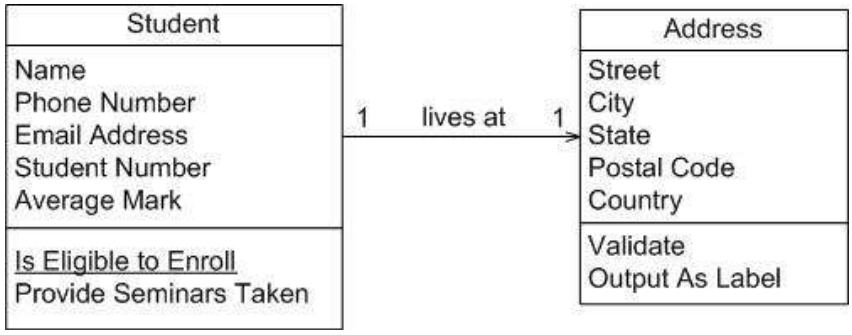


Gambar II.3 Class Ikon diagram

Sumber: Sholiq "*Pemodelan Sistem Informasi Berorientasi Objek dengan UML*" thn 2006. Hal 13

Class dapat merupakan implementasi dari sebuah interface, yaitu classabstrak yang hanya memiliki metoda. Interfacetidak 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 II.4 Relation Class Diagram

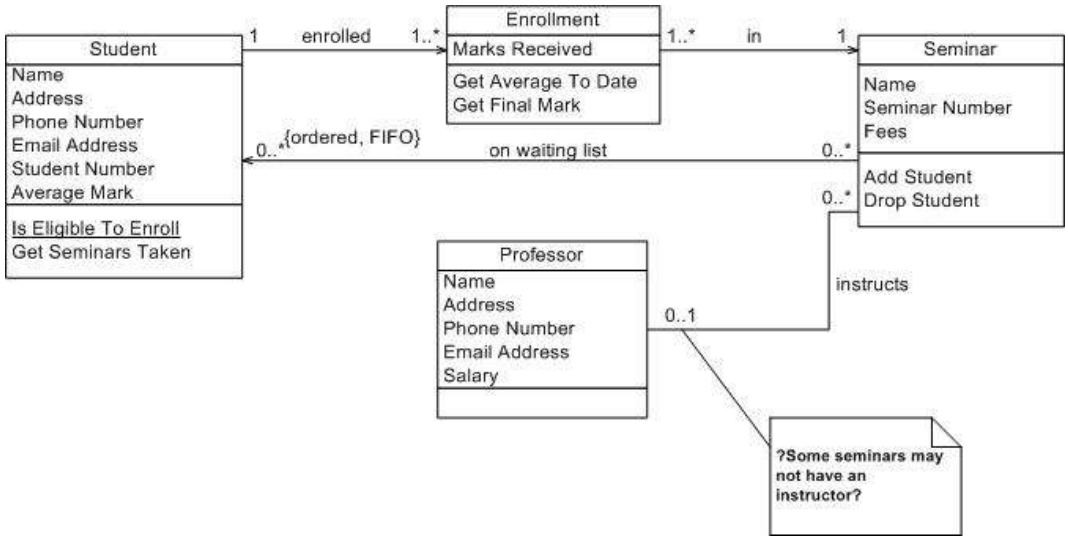
Sumber: Sholiq "*Pemodelan Sistem Informasi Berorientasi Objek dengan UML*" thn 2006. Hal 13

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 queryantar 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 diagramyang akan dijelaskan kemudian.

Contoh class diagram:



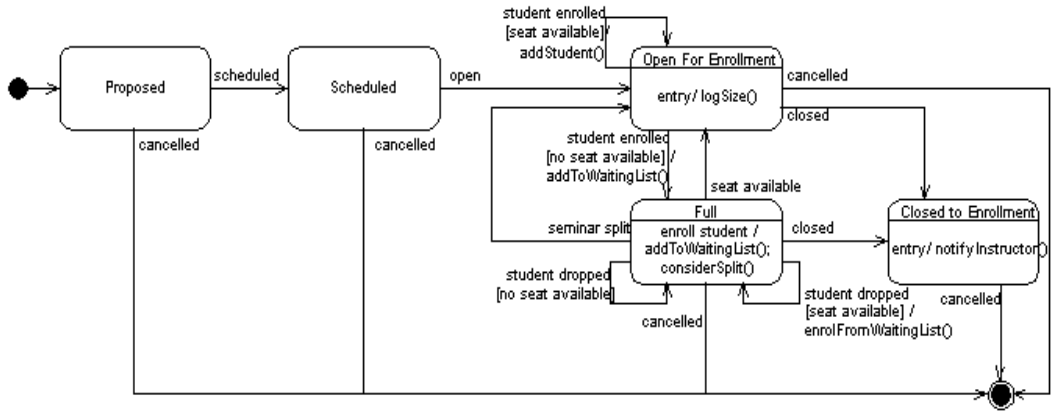
Gambar II.5 Class Diagram

Sumber: Sholiq "*Pemodelan Sistem Informasi Berorientasi Objek dengan UML*" thn 2006. Hal 8

c. Statechart Diagram

Statechart diagram menggambarkan transisi dan perubahan keadaan (dari satu state ke state lainnya suatu objek pada sistemsebagai 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.



Gambar II.6 Statechart Diagram

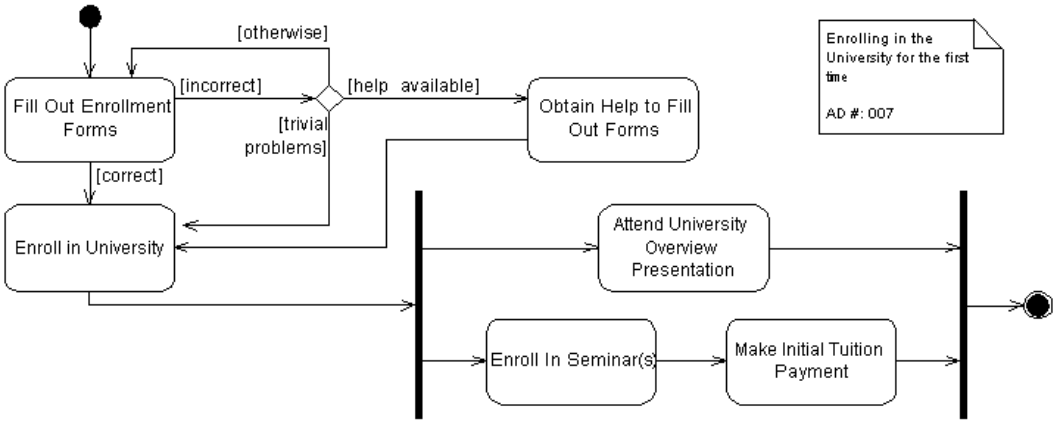
Sumber: Sholiq "*Pemodelan Sistem Informasi Berorientasi Objek dengan UML*" thn 2006. Hal 15

d. Activity Diagram

Activity diagrams menggambarkan berbagai alir aktivitas dalamsistem yang sedang dirancang, bagaimana masing-masing alur 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 actor 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:



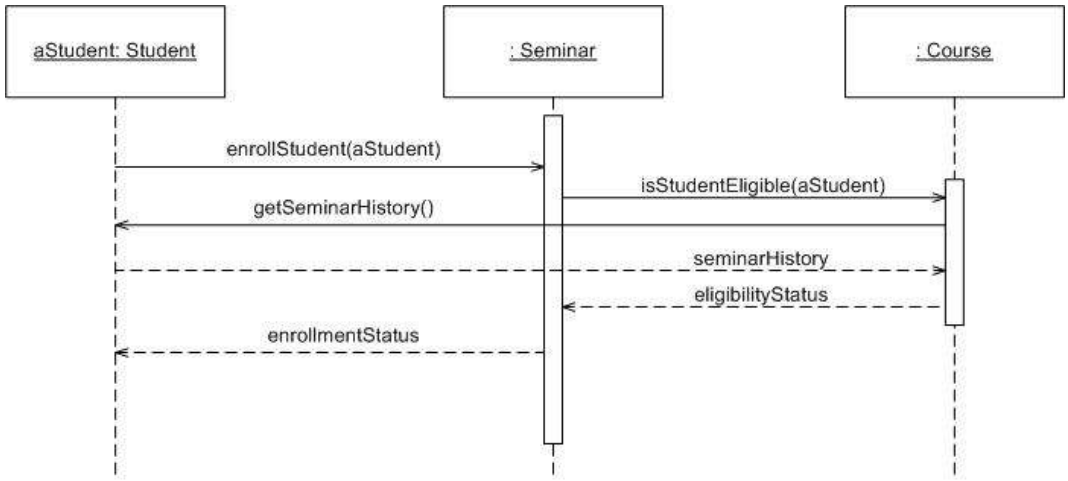
Gambar II.7 Diagram Activity

Sumber: Sholiq "*Pemodelan Sistem Informasi Berorientasi Objek dengan UML*" thn 2006. Hal 9

e. Sequence Diagram

Sequence diagrammenggambarkan 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 darisebuah event untuk menghasilkan output tertentu. Diawali dari apa yang mentrigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan. Masingmasing 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 untuk objek boundary, controllerdan persistent entity.



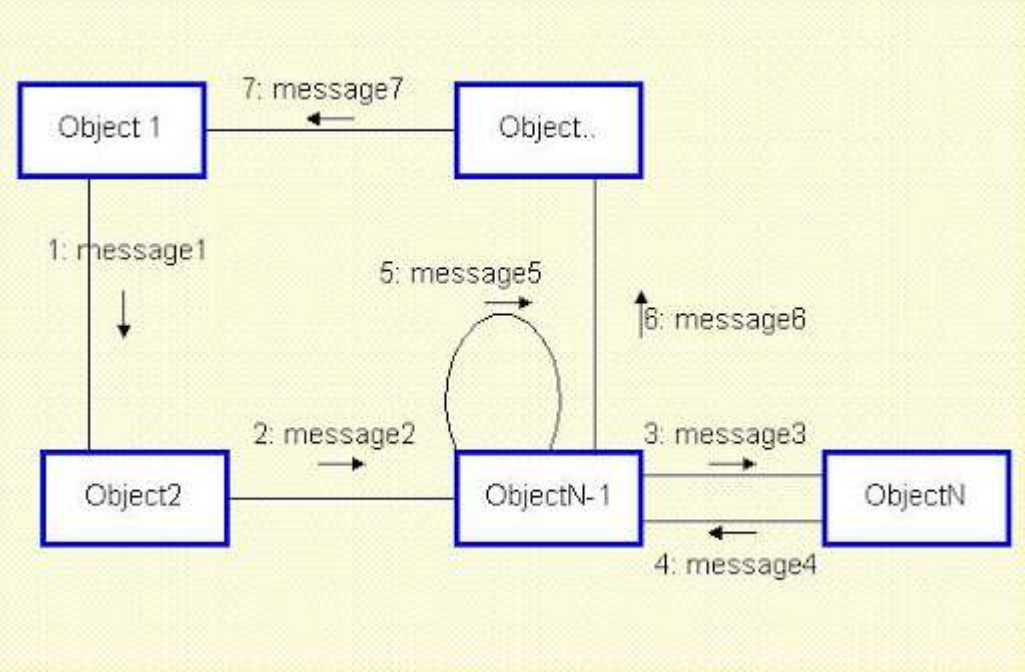
Gambar II.8 Sequence Activity

Sumber: Sholiq "*Pemodelan Sistem Informasi Berorientasi Objek dengan UML*" thn 2006. Hal 10

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 leveltertinggi memiliki nomor 1. Messages dari level yang sama memiliki prefiks yang sama.

Contoh Collaboration Diagram:



Gambar II.9 Collaboration Diagram

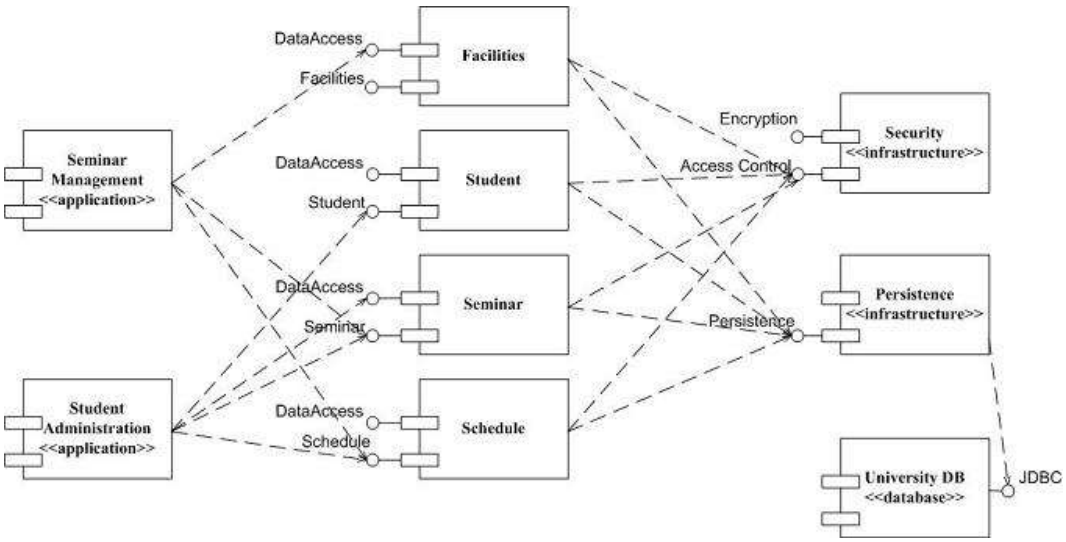
Sumber: Sholiq "Pemodelan Sistem Informasi Berorientasi Objek dengan UML" thn 2006. Hal 12

g. 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 II.10 Component Diagram

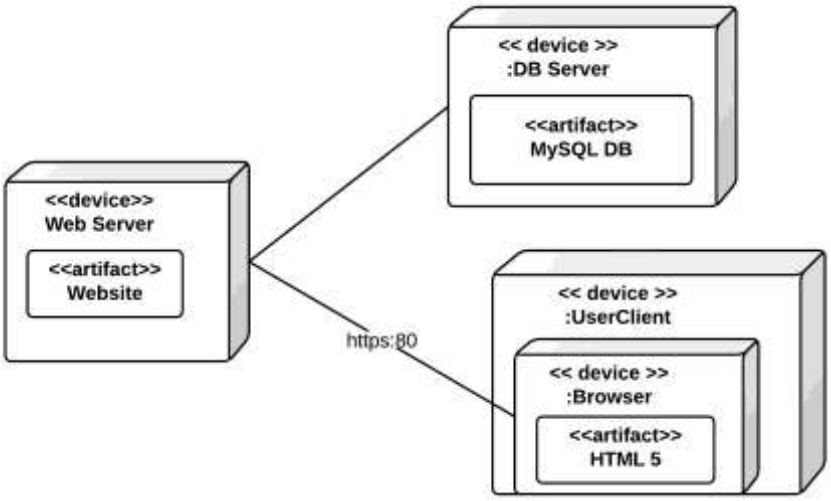
Sumber: Sholiq "Pemodelan Sistem Informasi Berorientasi Objek dengan UML" thn 2006. Hal 16

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 II.11 Deployment Diagram

Sumber: Sholiq "Pemodelan Sistem Informasi Berorientasi Objek dengan UML" thn 2006. Hal 18

2.10 Bahasa Pemprograman Yang di Gunakan

Bahasa pemprograman yang digunakan dalam penulisan skripsi ini adalah PHP.

2.10.1 PHP (Personal Home Page)

Menurut Kadir (2009 : 2) PHP merupakan singkatan dari *PHP Hypertext Preprocessor*, merupakan bahasa berbentuk skrip yang ditempatkan dalam server dan diproses di server. Secara khusus PHP dirancang untuk

membentuk aplikasi web dinamis. Skrip PHP berkedudukan sebagai tag dalam bahasa HTML. HTML (*HyperText Markup Language*) adalah bahasa standar untuk membuat halaman-halaman web. Program HELLO WORLD yang dituliskan PHP adalah sebagai berikut:

```
<?php
        echo "<b>Hello World!!! Ready For Code</b>";
?>
```

2.10.2 Framework

Framework secara sederhana dapat diartikan kumpulan dari fungsi-fungsi/prosedur-prosedur dan class-class untuk tujuan tertentu yang sudah siap digunakan sehingga bisa lebih mempermudah dan mempercepat pekerjaan seorang programer, tanpa harus membuat fungsi atau class dari awal.

Ada beberapa alasan mengapa menggunakan Framework:

1. Mempercepat dan mempermudah pembangunan sebuah aplikasi web.
2. Relatif memudahkan dalam proses maintenance karena sudah ada pola tertentu dalam sebuah framework (dengan syarat programmer mengikuti pola standar yang ada)
3. Umumnya framework menyediakan fasilitas-fasilitas yang umum dipakai sehingga kita tidak perlu membangun dari awal (misalnya validasi, ORM, pagination, multiple database, scaffolding, pengaturan session, error handling, dll
4. Lebih bebas dalam pengembangan jika dibandingkan CMS.

2.10.2.1 CodeIgniter

Menurut Hakim (2010:8) CodeIgniter adalah sebuah framework PHP yang dapat membantu mempercepat developer dalam pengembangan aplikasi web berbasis PHP dibanding jika menulis semua kode program dari awal.

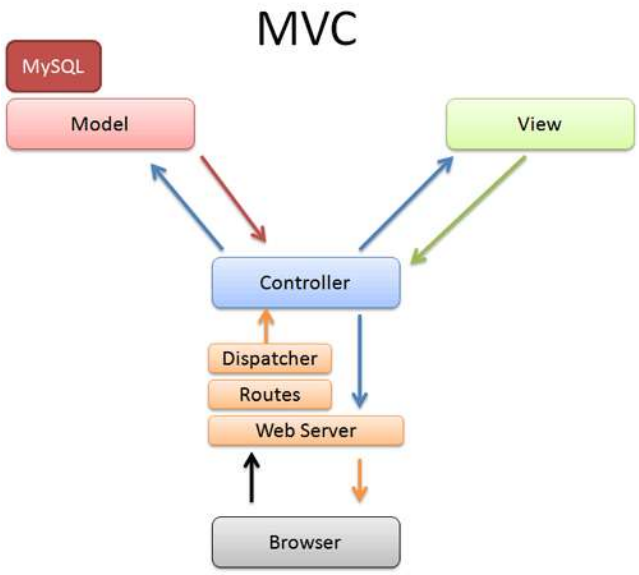


Gambar II.12 Logo CodeIgniter

Sumber : Hakim (2010 : 9) Membangun Web Berbasis PHP dengan Framework CodeIgniter

2.10.2.2 Design Patern: MVC (Model, View, Controller)

CodeIgniter adalah framework PHP yang dibuat berdasarkan kaidah model-View-controller. Dengan MVC, maka memungkinkan pemisahan antara layer application-logic dan presentation. Sehingga, dalam sebuah pengembangan web, seorang programmer bisa berkonsentrasi pada core-system, sedangkan web designer bisa berkonsentrasi pada tampilan web. Menariknya, skrip PHP, query MySQL, Javascript dan CSS bisa saling terpisah, tidak dibuat dalam satu skrip berukuran besar yang membutuhkan resource besar pula untuk mengesekusinya. Adapun alur program aplikasi berbasis *framework* Codeigniter dapat dilihat pada gambar dibawai ini.



Gambar II.13 Arsitektur MVC

Sumber : Hakim (2010 : 4) Membangun Web Berbasis PHP dengan *Framework* CodeIgniter

Gambar diatas menerangkan bahwa ketika datang sebuah user request, maka akan ditangani oleh controller, kemudian controller akan memanggil model jika memang diperlukan operasi database. Hasil dari query oleh model kemudian akan dikembalikan ke controller. Selanjutnya controller akan memanggil view yang tepat dan mengkombinasikannya dengan hasil query model. Hasil akhir dari operasi ini akan ditampilkan dibrowser.

Dalam konteks CodeIgniter dan aplikasi berbasis web, maka penerapan konsep MVC mengakibatkan kode program dapat dibagi menjadi tiga kategori, yaitu

1. *Model*

   Kode program (berupa OOP *class*) yang digunakan untuk memanipulasi *database*.

2. *View*

   Berupa *template* html/xml atau php untuk menampilkan data pada *browser*

3. *Controller*

   Kode program (berupa OOP *class*) yang digunakan untuk mengontrol aliran aplikasi (sebagai pengontol *model* dan *View*)

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

Menurut Sibero (2012:19) HTML (Hyper Text Markup Language) adalah bahasa yang digunakan pada dokumen web sebagai bahasa untuk pertukaran dokumen web. Contoh dokumen HTML sederhana:

```
<!DOCTYPE html>
<html>
    <head>
        <title>Hallo HTML</title>
    </head>
    <body>
        <p>Hallo Erwin!</p>
    </body>
</html>
```

Yang bisa dilakukan dengan HTML yaitu:

1. Mengontrol tampilan dari web dan content nya
2. Mempublikasikan dokumen secara online sehingga bisadi akses dari seluruh dunia.
3. Membuat online form yang bisa digunakan untuk menangani pendaftaran, transaksi secara online.
4. Menambahkan objek-objek 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. Secara teoritis bisa menggunakan style sheet technologi denga HTML. Akan tetapi pada prakteknya hanya Cascading StyleSheet (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.

2.10.4 Java Script

Menurut Aloysius Sigit W. (2011:1) jQuery adalah librari atau kumpulan kode JavaScript siap pakai. Keunggulan menggunakan jQuery dibandingkan dengan JavaScript standar, yaitu menyederhanakan kode JavaScript dengan cara memanggil fungsi-fungsi yang disediakan oleh jQuery. JavaScript sendiri merupakan bahasa Scripting yang bekerja disisi Client/Browser sehingga website bisa lebih interaktif.

JavaScript bisa digunakan untuk banyak tujuan, misalnya untuk membuat efek rollover baik di gambar maupun teks, dan yang penting juga adalah untuk membuat AJAX. JavaScript adalah bahasa yang digunakan untuk AJAX.

Kode JavaScript biasanya dituliskan dalam bentuk fungsi yang ditaruh di tag <head> yang dibuka dengan tag <script type="teks/javascript">.

Berikut ini adalah contoh yang akan menampilkan sebuah dialog box berisi "Halo Erwin!" ketika sebuah tombol diklik oleh pengguna:

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

2.10.5 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.
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.

2.11 Perangkat Lunak Yang Digunakan

Alat bantu perangkat lunak dalam pembangunan aplikasi yang akan dibuat adalah MySQL, Web Aplikasi, XAMPP.

2.11.1 Database MySQL

Menurut Kadir (2009 : 14) MySQL merupakan *software* yang tergolong *database server* dan bersifat *Open Source. Open Source* menyatakan bahwa *software* ini dilengkapi dengan *source code* (kode yang dipakai untuk membuat MySQL), selain tentu saja untuk *executable*-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi, dan bisa diperoleh dengan cara mengunduh di Internet secara gratis. MySQL juga bersifat multiplatform. MySQL dapat dijalankan pada berbagai sistem operasi.

Pengaksesan data dalam *database* dapat dilakukan dengan mudah melalui SQL (*Structured Query Language*). Data dalam *database* bisa diakses melalui aplikasi non-Web (misalnya dengan Visual Basic) maupun aplikasi Web (misalnya dengan PHP).

2.11.2 Aplikasi Web

Menurut Kadir (2009 : 2) Aplikasi web adalah jenis aplikasi yang diakses melalui *browser*, misalnya Internet Explorer dan *Mozilla Firefox*.

2.11.3 XAMPP

Menurut Riyanto (2010 : 1) XAMPP merupakan paket PHP dan MySQL berbasis *open source,* yang dapat digunakan sebagai *tool* pembantu pengembangan aplikasi berbasis PHP. XAMPP mengombinasikan beberapa paket perangkat lunak berbeda ke dalam satu paket. Sampai XAMPP versi 1.7.3, beberapa paket yang dibundel adalah Apache HTTPD, mod_autoindex_color module, FileZilla, OpenSLL, MySQL, PHP, FTP Server, PEAR, phpMyAdmin, dan lainya.

2.12 Definisi Online

Menurut Duwi Priyatno (2009:9) Online adalah keadaan dimana suatu perangkat seperti komputer terhubung ke jaringan internet.

Berdasarkan definisi diatas, maka penyusun menyimpulkan bahwa Sistem Informasi Online suatu perangkat komputer yang terhubung ke jaringan internet.

2.13 Internet

Menurut O`Brien (2003 :10) Internet merupakan jaringan komputer yang berkembang pesatdari jutaan bisnis, pendidikan, dan jaringan pemerintahan yang saling berhubungan dengan jumlah penggunanya lebih dari 200 negara.

2.14 WWW (Word Wide Web)

Menurut Yuhefizar (2008:10), website atau world wide web (www) adalah kumpulan halaman-halaman web yang mengandung informasi. *World Wide Web*,

biasa lebih terkenal disingkat sebagai WWW adalah suatu ruang informasi yang dipakai oleh pengenal global yang disebut Pengidentifikasi Sumber Seragam untuk mengenal pasti sumber daya berguna. WWW sering dianggap sama dengan Internet secara keseluruhan, walaupun sebenarnya ia hanyalah bagian daripada Internet.

WWW merupakan kumpulan peladen web dari seluruh dunia yang mempunyai kegunaan untuk menyediakan data dan informasi untuk dapat digunakan bersama. WWW adalah bagian yang paling menarik dari Internet. Melalui web, para pengguna dapat mengakses informasi-informasi yang tidak hanya berupa teks tetapi bisa juga berupa gambar, suara, video dan animasi.

Kegunaan ini tergolong masih baru dibandingkan surat elektronik, sebenarnya WWW merupakan kumpulan dokumen yang tersimpan di peladen web, dan yang peladennya tersebar di lima benua termasuk Indonesia yang terhubung menjadi satu melalui jaringan Internet. Dokumen-dokumen informasi ini disimpan atau dibuat dengan format HTML (Hypertext Markup Language).

Suatu halaman dokumen informasi dapat terdiri atas teks yang saling terkait dengan teks lainnya atau bahkan dengan dokumen lain. Keterkaitan halaman lewat teks ini disebut pranala. Dokumen infomasi ini tidak hanya terdiri dari teks tetapi dapat juga berupa gambar, mengandung suara bahkan klip video. Kaitan antar-dokumen yang seperti itu biasa disebut hipermedia.

Jadi dapat disimpulkan bahwa WWW adalah sekelompok dokumen multimedia yang saling bertautan dengan menggunakan tautan hiperteks. Dengan

mengeklik pranala (hipertaut), maka para pengguna bisa berpindah dari satu dokumen ke dokumen lainnya.

2.15 Topologi Jaringan

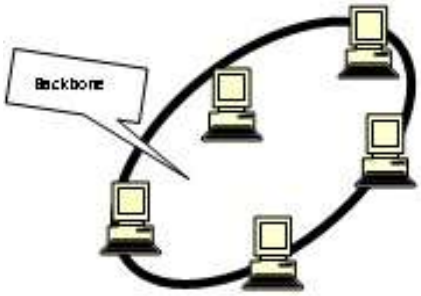
Menurut Abdul Kadir (2003 : 44) Yang dimaksud topologi jaringan adalah susunan fisik bagaimana node-node saling dihubungkan. Ada tiga topologi jaringan komputer yaitu :

1. Topologi Bus

   Pada topologi ini semua simpul (umumnya komputer) dihubungkan melalui kabel yang disebut bus. Kabel yang digunakan adalah kabel koaksial. Jika seorang pemakai mengirimkan pesan ke seorang pemakai lain maka pesan tersebut akan melalui bus. Setiap komputer perlu membaca alamat dalam pesan. Sekiranya alamat pesan cocok dengan alamat komputer pembaca, komputer tersebut segera mengambil pesan tersebut.

2. Topologi Cincin (*Ring*)

   Topologi cincin mirip dengan topologi bus. Informasi dikirim oleh sebuah komputer akan dilewatkan ke media transmisi, melewati satu komputer ke komputer berikutnya.



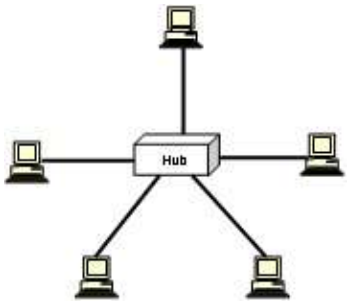
Gambar 2.1 Topologi Cincin

Sumber: Abdul Kadir : “Pengenalan Sistem Informasi” Th 2003 hal : 42

3. Topologi Bintang (*Star*)

Pada topologi ini terdapat komponen yang bertindak sebagai pusat pengontrol. Semua simpul yang hendak berkomunikasi selalu melalui pusat pengontrol tersebut. Dalam hal ini, pusat pengontrol berupa hub atau switch.

Gambar 2.2 Topologi Bintang



Sumber: Abdul Kadir : "Pengenalan Sistem Informasi” Th 2003 hal : 44