BAB II

LANDASAN TEORI

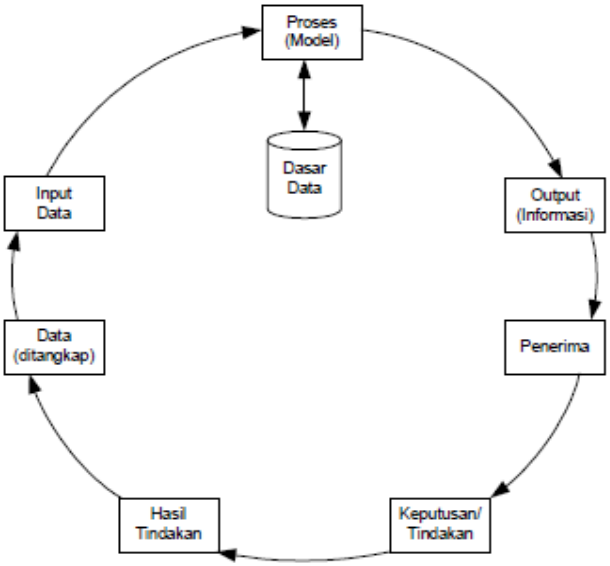
2.1. Pengertian Informasi

Information is a power, informasi merupakan suatu kekuatan. Informasi sangat penting bagi suatu organisasi, suatu organisasi yang kurang mendapatkan informasi akan menjadi lemah dan akhirnya berakhir. Robert N. Anthony dan John Dearden di dalam buku Jogiyanto (2001:7) menyebutkan keadaan dari sistem dalam hubungannya dengan keberakhirannya dengan istilah *entropy*. Informasi yang berguna bagi sistem akan menghindari proses *entropy* yang disebut dengan *negative entropy* atau *negentropy*.

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penggunanya. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian pada saat tertentu dan kesatuan nyata.

2.1.1. Siklus Informasi

Data yang telah diolah menjadi informasi diberikan kepada yang membutuhkan informasi, kemudian penerima informasi tersebut akan membuat suatu keputusan dan melakukan tindakan, yang berarti melakukan tindakan yang lain yang akan menghasilkan kembali suatu data. Data tersebut akan ditangkap sebagai input, diproses kembali melalui serangkaian langkah-langkah. Seterusnya membentuk suatu siklus. Siklus ini disebut siklus informasi (*information cycle*) dan disebut juga dengan siklus pengolahan data (*data processing cycle*).



Gambar 2.1. Siklus Informasi

[Sumber : Analisis dan Desain Sistem Informasi, Jogiyanto Hartono MBA.,Ph.D., Penerbit Andi Yogyakarta (2001)]

2.1.2 Nilai Informasi

Nilai dari informasi (*value of information*) ditentukan oleh manfaat dan biaya yang diperlukan untuk mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibanding dengan biaya mendapatkannya.

Informasi tidak dapat diukur dengan biaya semata, tetapi dapat diukur dengan efektivitas yang diperoleh. Pengukuran nilai informasi sering dikaitkan dengan analisis *cost effectiveness* atau *cost benefit*.

2.2 Pengertian Sistem

Terdapat dua kelompok pendekatan yang biasa digunakan dalam mendefinisikan sistem, yaitu pendekatan yang menekankan pada prosedur dan yang menekankan pada komponen atau elemennya.

Pendekatan sistem yang lebih menekankan pada prosedur, mendefinisikan sistem sebagai berikut [HM, Jogiyanto 3]:

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

Sedangkan pendekatan sistem yang lebih menekankan pada komponen atau elemen, mendefinisikan sistem yaitu [HM, Jogiyanto 3] :

"Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu".

Sistem itu sendiri memiliki karakteristik atau beberapa sifat tertentu, yaitu mempunyai komponen (*components*), batas sistem (*boundary*), lingkungan luar system (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*process*), dan sasaran (*objectives*) atau suatu tujuan (*goal*).

2.2.1. Karakteristik Sistem

Sistem mempunyai karakteristik atau sifat-sifat tertentu, di antaranya :

1. Komponen Sistem

   Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan. Komponen-komponen dapat terdiri dari

beberapa subsistem atau subbagian, di mana setiap subsistem tersebut memiliki fungsi khusus dan akan mempengaruhi proses sistem secara keseluruhan.

2. Batas Sistem

   Batas sistem *(boundary)* merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya.

3. Lingkungan Luar Sistem

   Lingkungan dari suatu sistem *(environment)* adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar dapat bersifat menguntungkan dan merugikan.

4. Penghubung Sistem

   Merupakan media penghubung antara subsistem, yang memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainnya. Keluaran *(output)* dari satu subsistem akan menjadi masukan *(input)* untuk subsistem lainnya melalui penghubung disamping sebagai untuk mengitegrasikan subsistem-subsistem menjadi satu kesatuan.

5. Masukan Sistem

   Masukan *(input)* adalah energi yang di masukan ke dalam sistem, yang dapat berupa masukan perawatan *(maintenance input)* dan masukan sinyal *(signal input)*. Masukan perawatan adalah energi yang dimasukan supaya sistem dapat beroperasi, sedangkan masukan sinyal adalah energi yang diproses untuk mendapatkan keluaran.

6. Keluaran Sistem

Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lainnya.

7. Pengolah sistem

   Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan lain menjadi keluaran berupa barang jadi.

8. Sasaran *(objectives)* atau tujuan *(goal)*

   Suatu sistem mempunyai suatu tujuan *(goal)* dan sasaran *(objectives)*. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengena pada sasaran atau tujuannya.

2.3. Definisi Sistem Informasi

Sistem informasi didefinisikan oleh Robert A. Leitch dan K. Roscoe Davis di dalam buku Jogiyanto (2001:11) sebagai berikut:

"sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan yang diperlukan".

Sistem informasi memiliki sejumlah komponen tertentu. Seperti yang dikemukakan oleh Robert dan Donald Symanzky dalam buku Jogiyanto

(2001:12), bahwa sistem informasi terdiri dari beberapa komponen yang berbeda yaitu, manusia, data, *hardware*, dan *software*. Sebagai suatu sistem, setiap komponen tersebut berinteraksi satu dengan lainnya membentuk satu kesatuan untuk mencapai sasarannya.

Faktor – faktor yang menentukan kehandalan dari suatu sistem informasi dapat dikatakan baik jika memenuhi kriteria-kriteria sebagai berikut :

1. Keunggulan (*usefulness*)

   Yaitu suatu sistem yang harus dapat menghasilkan informasi yang tepat dan relevan untuk mengambil keputusan manajemen dan personil operasi dalam organisasi.

2. Ekonomis

   Kemampuan sistem yang mempengaruhi sistem harus bernilai manfaat minimal, sebesar biayanya.

3. Kehandalan (*Reliability*)

   Keluaran dari sistem harus mempunyai tingkat ketelitian tinggi dan system tersebut harus beroperasi secara efektif.

4. Pelayanan (*Customer Service*)

   Yakni suatu sistem memberikan pelayanan yang baik dan efisien kepadapara pengguna sistem pada saat berhubungan dengan organisasi.

5. Kapasitas (*Capacity*)

   Setiap sistem harus mempunyai kapasitas yang memadai untuk menangani setiap periode sesuai yang dibutuhkan.

6. Sederhana dalam kemudahan (*Simplicity*)

Sistem tersebut lebih sederhana (umum) sehingga struktur dan operasinya dapat dengan mudah dimengerti dan prosedurnya mudah diikuti.

7. Fleksibel (*Fleksibility*)

Sistem informasi ini harus dapat digunakan dalam kondisi yang bagaimana yang diinginkan oleh organisasi tersebut atau pengguna tertentu.

2.4 Pengembangan dan Pendekatan Perancangan Sistem Informasi

.Pengembangan sistem (sistem *development*) dapat berarti menyusun sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem yang telah ada. Sistem yang lama perlu diganti atau diperbaiki disebabkan karena beberapa hal :

1. Adanya permasalahan-permasalahan (*Problems)* yang timbul di dalam system lama. Permasalahan-permasalahan yang timbul dapat berupa :

a. Ketidakberesan

Ketidakberesan dalam sistem yang lama menyebabkan sistem yang lama tidak dapat beroperasi sesuai dengan yang diharapkan. Ketidakberesan ini dapat berupa:

1. Kecurangan-kecurangan yang disengaja yang menyebabkan tidak amannya harta kekayaan perusahaan dan kebenaran dari data menjadi tidak terjamin.
2. Kesalahan-kesalahan yang tidak disengaja yang juga dapat menyebabkan kebenaran dari data tidak terjamin.
3. Tidak efisiennya operasi.

4. Tidak ditaatinya kebijaksanaan manajemen yang telah ditetapkan.

b. Pertumbuhan Organisasi

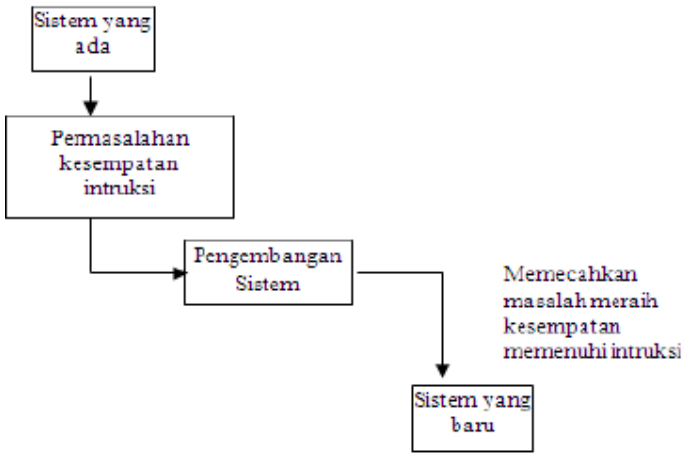
Pertumbuhan organisasi menyebabkan harus disusunnya sistem yang baru. Pertumbuhan organisasi diantaranya adalah kebutuhan informasi yang semakin luas, *volume* pengolahan data semakin meningkat, perubahan prinsip akuntansi yang baru.

2. Untuk meraih kesempatan-kesempatan (*Opportunities*)

Kecepatan informasi atau efisiensi waktu sangat menentukan berhasil atau tidaknya strategi dan rencana yang telah disusun untuk meraih kesempatan kesempatan yang ada.

3. Adanya intruksi-intruksi (*Directives*)

Pengembangan sistem yang baru dapat juga terjadi karena adanya intruksiintruksi dari atas pimpinan ataupun dari luar organisasi, seperti misalnya peraturan pemerintah.



Gambar 2.2 Pengembangan SIstem Informasi

[Sumber : Analisis dan Desain Sistem Informasi, Jogiyanto Hartono MBA.,Ph.D., Penerbit Andi Yogyakarta (2001)]

Terdapat beberapa metode pendekatan dan perancangan sistem informasi yang digunakan oleh berbagai pengembang sesuai dengan tujuan dari pengembagan system informasi yang dihadapi, berikut merupakan metode pendekatan yang akan digunakan oleh penulis berikut penjelasannya.

2.4.1 Pendekatan Sistem Berorientasi Objek

Metodologi (analisis dan perancangan perangkat lunak) berorientasi objek diusulkan oleh penulis-penulis buku/artikel kemudian praktek-praktek penerapannya oleh para penulis tersebut terlembagakan, contoh-contoh metode /metodologi :

– Shlaer/Mellor

– Coad/Yourdon

– Object Modelling Techniques (Rumbaugh, et. al.)

– Responsibility Driven Design (Wirfs-Brock, et. al.)

– Objectory

– OOAD Booch

– Fusion (Coleman, et. al.)

Tiap metode/metodologi berorientasi objek memiliki teknik dan notasi sendiri yang satu sama lain berbeda Kesamaan umum dari semua metode/metodologi berorientasi objek terletak pada cara pandang, yaitu **:**

a. **Static,** menggambarkan keterkaitan logik antar entitas pada sistem

b. **Dynamic**, menggambarkan hubungan yang terbentuk akibat dinamika sistem.

2.4.1.1 Konsep *Objek Oriented Analisis And Desain*(*OOAD*)

Object Oriented Analysis Design (OOAD) adalah rekayasa perangkat lunak yang model pendekatan sistem sebagai kelompok interaksi objek. Beberapa entitas mewakili setiap obyek, yang menarik dalam sistem menjadi model dan kelas, elemen data, dan perilaku. Analisis berorientasi objek (OOA) berlaku objek modeling teknik untuk menganalisis fungsional persyaratan untuk sistem.

Secara singkat OOAD adalah metode analisis yang memeriksa requirements dari sudut pandang kelas-kelas dan objek yang ditemui dalam ruang lingkup permasalahan yang mengarahkan arsitektur software yang didasarkan

pada manipulasi objek-objek sistem atau subsistem. Sekarang ini terdapat berbagai metodologi yang digunakan dalam object oriented analisys and design (OOAD).

Beberapa konsep dasar dalam OOAD adalah :

a. Objek (Object)

Objek adalah benda yang secara fisik dan konseptual yang ada disekitar kita. Beberapa contoh objek, seperti hardware, software, dokumen, manusia, konsep, dan lainnya. State adalah objek yang mempunyai keadaan sesaat yang menggambarkan keadaan dari objek tersebut. State dinyatakan dengan atribut, behaviour, interface, service, dan method.

b. Kelas (Class)

Class adalah definisi umum dari himpunan objek yang sejenis. Kelas menetapkan spesifikasi perilaku (behaviour) dan atribut-atribut dari objek tersebut. Class adalah abstraksi dari entitas dunia nyata.

c. Kotak hitam (black Box)

Black box adalah dasar dari implementasi objek. Dalam operasi OO, hanya para developer yang dapat memahami detail dari proses-proses yang ada dalam kotak hitam, sedangkan para user tidak mengetahui. Menggunakan proses enkapsulasi dan message.

d. Asosiasi dan Agregasi

Asosiasi adalah hubungan yang mempunyai makna antara sejumlah objek. Asosiasi digambarkan dengan sebuah garis penghubung di antara objeknya.

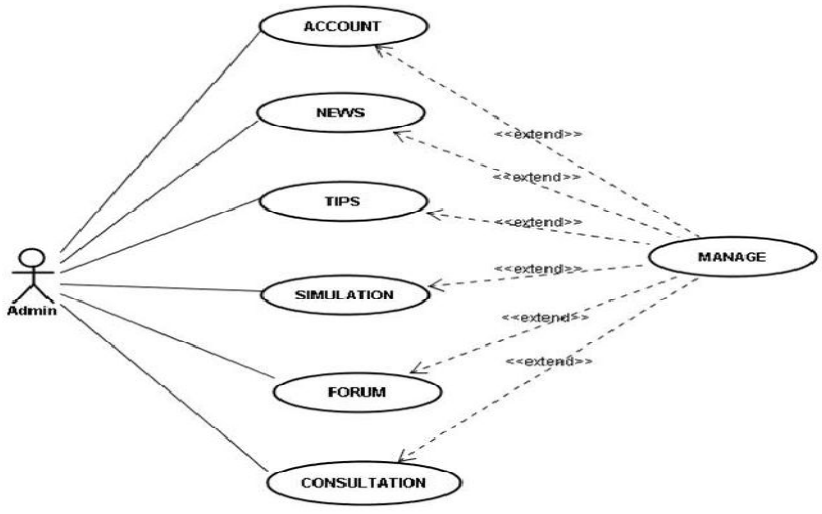
Agregasi adalah bentuk khusus sebuah asosiasi yang menggambarkan seluruh bagian pada suatu objek merupakan bagian dari objek yang lain.

2.4.1.2 Pengenalan UML

Unified Modelling Language (*UML*) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah sistem. UML mendefinisikan diagram-diagram berikut ini :

1. Use case diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara actor dengan sistem.



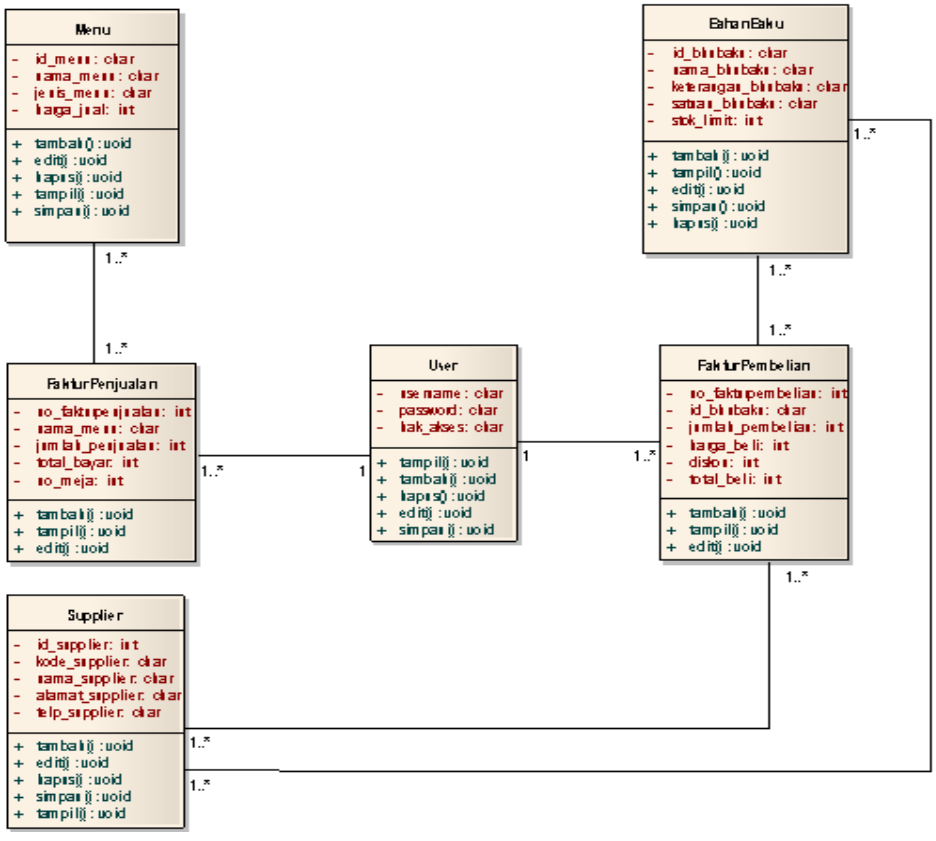
Gambar 2.3 Contoh *Use Case*

2. 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 memiliki tiga area pokok :

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



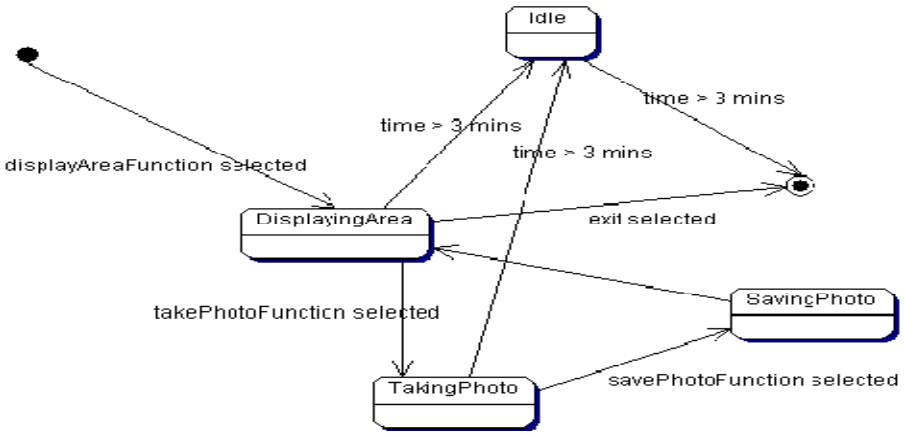
Gambar 2.4 Contoh *Class Diagram*

3. Behaviour Diagram

Terbagi kedalam 2 diagram yaitu :

a. Statechart diagram

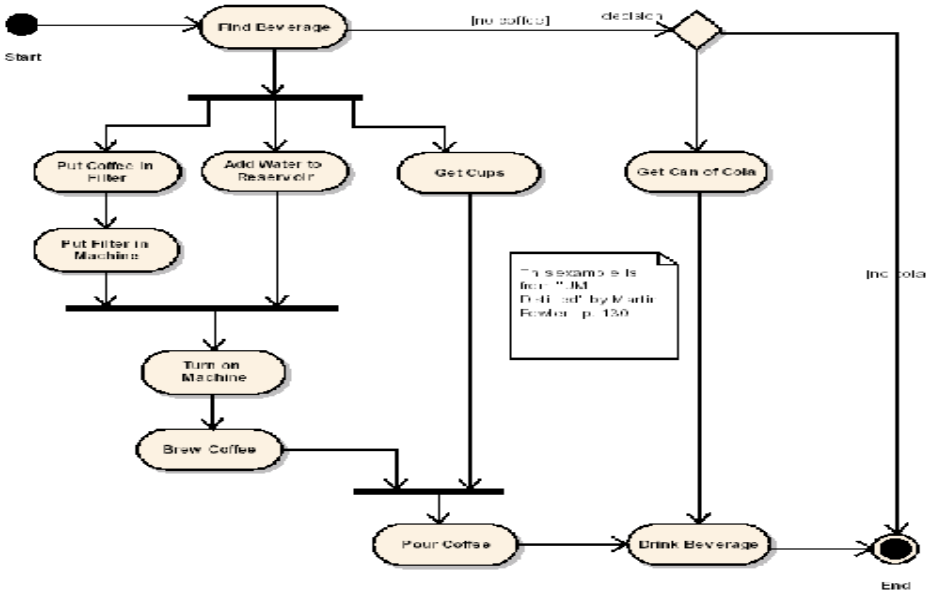
Menggambarkan transisi dan perubahan keadaan (dari satu *state* ke *state* lainnya) suatu objek pada sistem sebagai akibat dari stimuli yang diterima. Pada umumnya *statechart* diagram menggambarkan *class* tertentu (satu *class* dapat memiliki lebih dari satu *statechart diagram*).



Gambar 2.5 Contoh *Statechart Diagram*

b. Activity Diagram

Activity diagram 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.



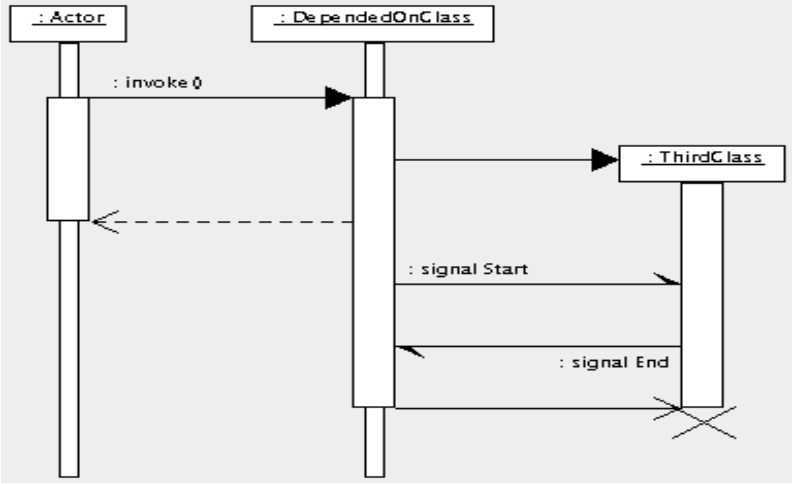
Gambar 2.6 Contoh *Activity Diagram*

4. Interaction diagram

Terbagi menjadi 2 diagram, yaitu :

a. Sequence Diagram

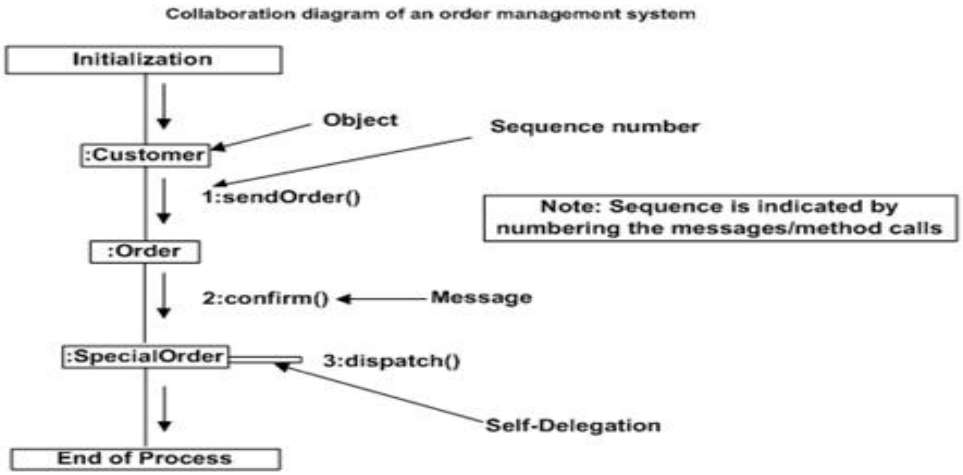
Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence diagram* terdiri atar dimensi vertical (waktu) dan dimensi horizontal (objek-objek yang terkait).



Gambar 2.7 Contoh *Sequence Diagram*

b. Collaboration Diagram

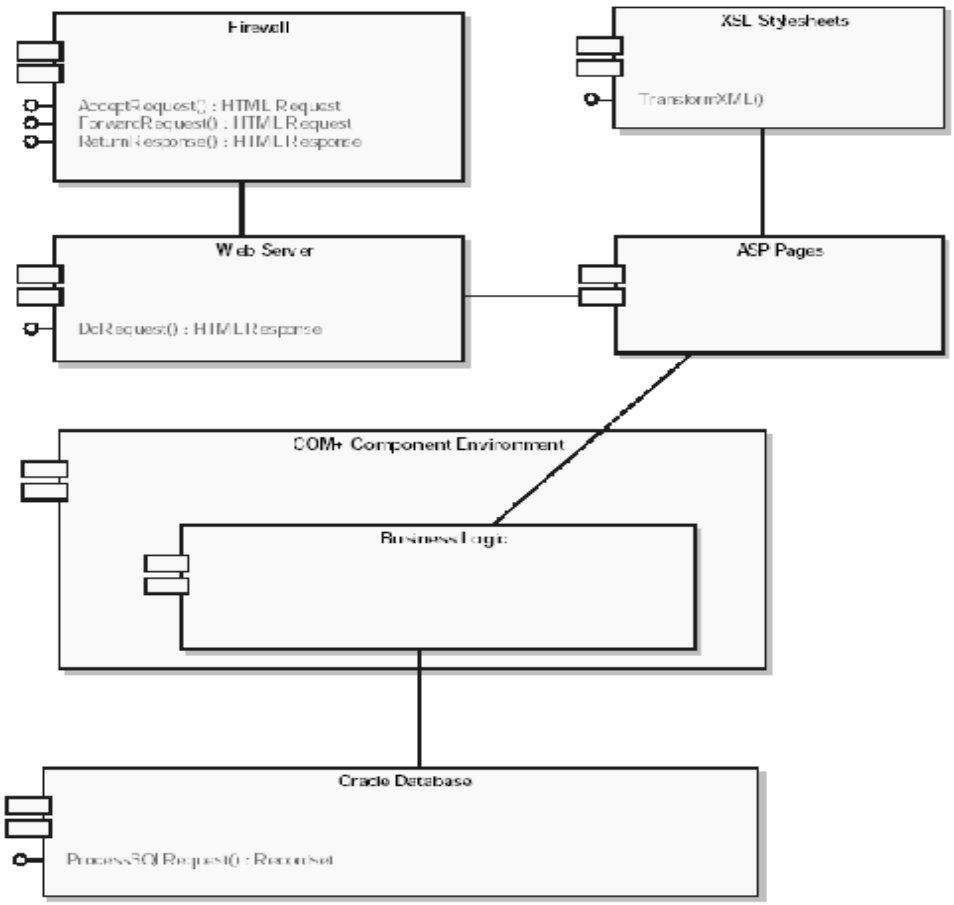
Collaboration diagram juga menggambarkan interaksi antar objek seperti *sequence diagram*, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian *message.*



Gambar 2.8 Contoh *Collaboration Diagram*

5. Component diagram

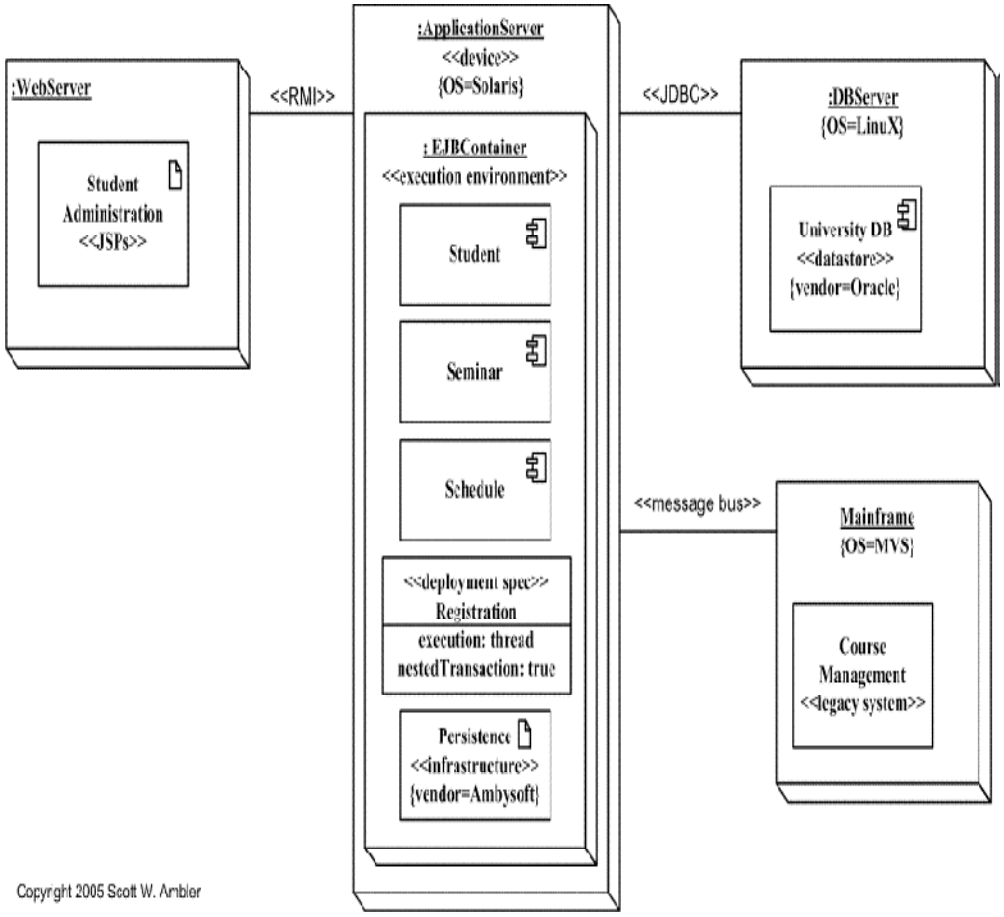
Component diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya. Komponen piranti lunak adalah modul berisi code, baik berisi source code maupun binary code, baik library maupun executable, baik yang muncul pada compile time, link 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.



Gambar 2.9 Contoh *Componen Diagram*

6. 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.

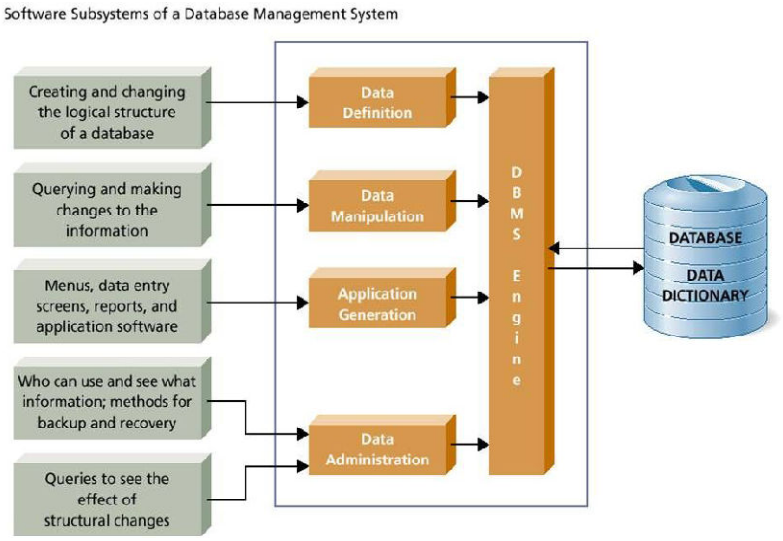


Gambar 2.10 Contoh *Deployment diagram*

Scott W. Ambler (2005) from http://www.agilemodeling.com/

2.4.1.3 Konsep Dasar Basis Data

Secara sederhana *database* dapat diungkapkan sebagai suatu pengorganisasian data dengan bantuan komputer yang memungkinkan data dapat diakses dengan mudah dan cepat. Dalam hal ini, pengertian akses dapat mencakup pemerolehan data maupun pemanipulasian data seperti menambah serta menghapus data. Sebagai mana diketahui, manajemen modern mengikutsertkana informasi sebagai sumber daya penting yang setara dengan sumber daya manusia, uang, mesin dan material.



Gambar 2.11 RDBMS

Sumber : www.embeddedtechjournal.com/articles_2006/200...

Dengan adanya komputer, data dapat disimpan dalam media pengingat yang disebut *hardisk*. Dengan menggunakan media ini kehadirian kerta yang digunakan dapat untuk menyimpan data dapat dikurangi. Selain itu, data menjadi lebih cepat untuk diakses, terutama dalam bentuk *database*.

DBMS singkatan dari *Database Management* Sistem. DBMS merupakan perangkat lunak atau program komputer yang dirancang khusus untuk memudahkan pengkodean database. Salah satu macam DBMS yang popoluer dewasa ini berupa RDBMS dan ORDBMS , yang menggunakan model basis data atau dalam bentuk tabel-tabel yang saling berhubungan.

2.5 Jaringan Komputer

Jaringan komputer adalah sekumpulan komputer yang berhubungan antara yang satu dengan yang lainnya menggunakan *protocol* komunikasi sehingga dapat saling bertukar data dan informasi serta dapat menggunakan perangkat keras secara bersama. Selain itu jaringan komputer dapat diartikan sebagai sekumpulan terminal komunikasi yang berada di berbagai lokasi berbeda yang terdiri dari dua computer atau lebih yang saling berhubungan (Budi Irawan (2005:6).

2.5.1. Manfaat Jaringan Komputer

Tujuan dibangunnya suatu jaringan komputer adalah untuk membawa data informasi dari sisi pengirim menuju penerima secara cepat dan tepat tanpa adanya kesalahan melalui media transmisi atau media komunikasi tertentu.

2.5.2. Definisi Internet

"Internet adalah suatu jaringan komputer global yang terbentuk dari jaringan - jaringan komputer lokal dan regional yang memungkinkan komunikasi data antar komputer yang terhubung ke jaringan tersebut"

Internet awalnya merupakan suatu recana dari Departemen Pertahanan Amerika Serikat (*US Departement of Defenses*) pada sekitar tahun 1960. Dimulai dari suatu proyek yang dinamakan ARPANET atau *Advanced Research Project agency Network*. Beberapa universitas di Amerika Serikat diantaranya UCLA, Stanford, UC Santa Barbara dan University of Utah, diminta bantuan dalam mengerjakan proyek ini dan awalnya telah berhasil menghubungkan empat komputer di lokasi universitas yang berbeda tersebut. Perkembangan ARPANET ini cukup pesat dilihat perkembangan komputer pada saat itu. Sebagai gambarannya pada tahun 1977 ARPANET telah menghubungkan lebih dari 100 mainframe komputer dan saat ini terdapat 4 juta *host* jaringan yang terhubung pada jaringan ini.

2.6. Perangkat Lunak Pendukung

2.6.1. PHP (*Pheriperal HyperText Preprocessor*)

PHP secara resmi merupakan kependekan dari *Pheriperal HyperText Preprocessor*, merupakan bahasa *script server side*yang disisipkan pada HTML. PHP berawal saat Rasmus Lerdorf membuat sejumlah skrip *"Perl"* yang dapat mengamati siapa saja yang melihat-lihat daftar riwayat hidupnya, yakni pada tahun 1994. Skrip-skrip ini selanjutnya dikemas menjadi tool yang disebut *"Personal Home Page"*. Paket inilah yang menjadi cikal bakal PHP. Pada tahun 1995 , Rasmus menciptakan *PHP/FI* Versi 2. Pada Versi inilah pemrogram dapat menempelkan kode tersruktur di dalam tag *HTML (Hypertext Markup Language)*

Kode PHP juga bisa berkomunikasi dengan database dan melakukan perhitunganperhitungan yang kompleks sambil jalan.

Pada saat ini, PHP cukup popular sebagai piranti pemrograman Web, terutama di lingkungan Linux. Walaupun demikian, PHP sebenarnya juga dapat berfungsi pada server-server yang berbasis *UNIX, Windows NT*. PHP juga dirancang untuk diintegrasikan dengan *web server Apache.*

2.6.2. MySQL

MySQL sebenarnya produk yang berjalan pada *platform* Linux. Karena sifatnya yang *open sorce*, dia dapat dijalankan pada semua *platform* baik windows maupaun Linux. Selain itu, MySQL juga merupakan program pengakses *database* yang bersifat jaringan sehingga dapat digunakan untuk aplikasi *Multi User* (banyak pengguna). Saat ini *database* MySQL telah digunakan oleh semua *programmer database*, apalagi dalam pemrograman *web*.

Kelebihan lain dari MySQL adalah menggunakan bahasa Query standar yang dimiliki SQL (*Structure Query Language*). SQL adalah suatu bahasa permintaan yang terstruktur yang telah distandarkan untuk semua program pengakses *database* seperti Oracle, PostgreSQL, SQL Server, dan lain-lain. Adapun program-program yang menggunakan bahasa SQL antara lain MySQL, PostgreSQL, Oraxcle, SQL Server 97, 2000 dan Interbase.

Program-program aplikasi pendukung MySQL antara lain PHP, Visual Delphi, Visual Basic dan Cold Fusion.

2.6.3. Macromedia *Dreamweaver 8*

Macromedia *Dreamweaver 8* adalah suatu bentuk program *web editor* yang dibuat oleh Makromedia. Dengan program ini seorang *programer web* dapat dengan mudah membuat dan mendesain webnya (Nugroho, 2004:91). Sebagai editor, *Dreamweaver 8* mempunyai sifat *WYSIWYG* dibaca (*Waysiwig*), artinya apa yang kamu lihat akan kamu peroleh (*what you see is what you get*). Dengan kelebihan ini seorang *programmer* dapat langsung melihat hasil buatannya tanpa harus dibuka di *browser.*