BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Definisi sistem menurut Jogiyanto HM (1995 : 5) adalah sebagai berikut :

Terdapat dua kelompok pendekatan di dalam pendefinisian sistem, yaitu yang menekankan pada prosedur dan menekankan pada komponen atau elemen. Pendekatan sistem yang lebih menekankan pada prosedur mendefinisikan sistem sebagai berikut :

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

Pendekatan sistem yang menekankan pada komponen atau elemen-elemen mendefinisikan sebagai berikut :

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

2.1.1 Elemen Sistem

1. Sistem Fisik (*Physical System*) :

Kumpulan elemen-elemen/ unsur-unsur yang saling berinteraksi satu sama lain secara fisik serta dapat diidentifikasikan secara nyata tujuan-tujuannya.

Contoh :

a. Sistem transportasi, elemen : petugas, mesin, organisasi yang menjalankan transportasi

b. Sistem Komputer, elemen : peralatan yang berfungsi bersama - sama untuk menjalankan pengolahan data.

2. Sistem Abstrak (*Abstract System*) :

Sistem yang dibentuk akibat terselenggaranya ketergantungan dan tidak dapat diidentifikasikan secara nyata, tetapi dapat diuraikan elemen-elemennya.

Contoh : Sistem Teologi, hubungan antara manusia dengan Tuhan.

2.1.2 Karakteristik Sistem

1. Organisasi

Mencakup struktur dan fungsi organisasi

2. Interaksi.

Saling ke terhubungan antara bagian yang satu dengan lainnya.

3. Interdependensi.

Bagian yang satu mempunyai ketergantungan dengan bagian yang lainnya.

4. Integritas.

Suatu keterpaduan antara subsistem-subsistem untuk mencapai tujuan.

5. Main objection (tujuan utama).

Pemusatan tujuan yang sama dari masing-masing subsistem.

2.1.3 Klasifikasi Sistem

1. *Deterministik Sistem*

Sistem dimana operasi-operasi (input/output) yang terjadi didalamnya dapat ditentukan/ diketahui dengan pasti.

Contoh :

a. Program komputer, melaksanakan secara tepat sesuai dengan rangkaian instruksinya.

b. Sistem penggajian.

2. *Probabilistik Sistem*

Sistem yang input dan prosesnya dapat didefinisikan, tetapi output yang dihasilkan tidak dapat ditentukan dengan pasti; (Selalu ada sedikit kesalahan/penyimpangan terhadap ramalan jalannya sistem).

Contoh :

a. Sistem penilaian ujian

b. Sistem pemasaran.

3. *Open Sistem*

Sistem yang mengalami pertukaran energi, materi atau informasi dengan lingkungannya. Sistem ini cenderung memiliki sifat adaptasi, dapat menyesuaikan diri dengan lingkungannya sehingga dapat meneruskan eksistensinya.

Contoh :

Sistem keorganisasian memiliki kemampuan adaptasi.(Bisnis dalam menghadapi persaingan dari pasar yang berubah. Perusahaan yang tidak dapat menyesuaikan diri akan tersingkir).

4. *Closed Sistem*

Sistem fisik di mana proses yang terjadi tidak mengalami pertukaran materi, energi atau informasi dengan lingkungan di luar sistem tersebut.

Contoh : reaksi kimia dalam tabung berisolasi dan tertutup.

5. *Relatively Closed Sistem*

Sistem yang tertutup tetapi tidak tertutup sama sekali untuk menerima pengaruh-pengaruh lain. Sistem ini dalam operasinya dapat menerima pengaruh dari luar yang sudah didefinisikan dalam batas-batas tertentu.

Contoh :

Sistem komputer. (Sistem ini hanya menerima masukan yang telah ditentukan sebelumnya, mengolahnya dan memberikan keluaran yang juga telah ditentukan sebelumnya. Tidak terpengaruh oleh gejolak di luar sistem).

6. *Artificial Sistem*

Sistem yang meniru kejadian dalam alam. Sistem ini dibentuk berdasarkan kejadian di alam di mana manusia tidak mampu melakukannya. Dengan kata lain tiruan yang ada di alam.

Contoh :

a. Sistem AI, yaitu program komputer yang mampu membuat komputer seolah-olah berpikir.

b. Sistem robotika.

c. Jaringan neutral network.

7. *Natural Sistem*

Sistem yang dibentuk dari kejadian dalam alam.

Contoh : laut, pantai, atmosfer, tata surya, dll.

8. *Manned Sistem*

Sistem penjelasan tingkah laku yang meliputi keikutsertaan manusia.

2.2 Pengertian Informasi

Informasi sangat penting bagi perusahaan. Selain itu informasi juga penting bagi sistem, karena apabila suatu sistem tidak mempunyai informasi maka akan menjadi suatu sistem yang susah berkembang. Defenisi Informasi menurut Jogiyanto HM (1995 : 3) adalah sebagai berikut : "*Data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya*".

Menurut Joku Bruch dan Gary Grudnitski dalam buku Analisa dan Desain Sistem Informasi mengatakan bahwa kualitas sistem terdiri dari tiga hal, yaitu :

a. Akurat

Informasi harus bebas dari kesalahan dan tidak menyesatkan.

b. Tepat pada waktunya

Informasi yang datang pada penerima tidak boleh terlambat.

c. Relevan

Informasi tersebut mempunyai manfaat untuk pemakainya.

2.3 Pengertian Sistem Informasi

Telah diketahui bahwa informasi merupakan hal yang sangat penting bagi manajemen di dalam pengambilan keputusan. Pertanyaannya adalah darimana informasi tersebut bisa didapatkan?. Informasi dapat diperoleh dari sistem informasi (*information sistem*) atau disebut juga dengan *processing sistems* atau *information processing sistem* atau *information-genering sistems*. Sistem informasi didefinisikan oleh Robert A. Leitch dan K. Roscoe Davis sebagai berikut:

"Sistem Informasi adalah 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-laporan yang diperlukan".

2.4 Sistem Informasi Perpustakaan pada Ponsel

Sistem Informasi Perpustakaan pada ponsel mengelola data komponen perpustakaan ke dalam suatu aplikasi dan basis data (*Data Base*) yang di bangun dan ditujukan untuk perangkat yang berbasis *mobile*, artinya sistem informasi tersebut dapat diakses dengan perangkat *mobile* seperti telepon seluler atau PDA/*Pocket PC*. Hal ini yang membedakan dengan sistem informasi akademik berbasis komputer yang hanya dapat menggunakan perangkat komputer sebagai media antarmuka kepada penggunanya.

1.5 Teknik Object Oriented

Object Oriented Programming adalah suatu teknik atau cara untuk membuat program (perangkat lunak) objek. Pemrograman yang dibangun dengan cara membungkus (enkapsulasi) komponennya menjadi kelompok data dan fungsi yang dapat mewarisi atribut dan sifat dari komponen lainnya, dan komponen-komponen tersebut dapat berinteraksi satu sama lainnya.

1 Pemrograman yang didasarkan pada objek-objek, yaitu suatu unit terkecil dari program yang mengandung data dan fungsi yang bekerja atas objek tersebut.

2 Perangkat lunak objek merupakan perangkat lunak yang dibangun dari kelas-kelas dan objek-objek yang saling berinteraksi satu sama lainnya. Kelas adalah deskripsi statis dari sesuatu objek-objek yang memiliki karakteristik yang sama. Objek adalah sesuatu yang diinstansiasi dari kelas.

1.5.1 Object Oriented Analysis (OOA)

Object Oriented Analysis (OOA) mencari domain masalah, dengan tujuan menghasilkan model konseptual yang terdapat pada area yang dianalisa. OOA merupakan tahapan yang harus dipenuhi sebelum melakukan Object Oriented Design (OOD). Sasaran OOA adalah pengembangan sederetan yag menggambarkan perangkat lunak komputer pada saat perangkat itu bekerja untuk memenuhi serangkai persyaratan yang ditentukan oleh pemakai. Proses OOA tidak dimulai dengan suatu pemikiran mengenai obyek, melainkan dengan memahami cara sistem akan digunakan oleh manusia atau sistem yang lain.

2.5.2 *Object Oriented Design* (OOD)

Object Oriented Design (OOD) mentranformasi model analisis yang dibuat dengan menggunakan OOA kedalam suatu model desain yang berfungsi sebagai *blue print* perangkat lunak. OOD mempunyai sifat unik untuk membangun empat konsep desain perangkat lunak yang penting yaitu : abstraksi, penyembunyian informasi, indepedensi fungsional dan modularitas.

2.5.3. Komponen – komponen Object Oriented Analysis and Design (OOAD)

Dalam konsep OOAD terdapat banyak komponen yang harus diketahui dan dipahami sebelum melakukan proses analisis dan desain. Komponen – komponen yang ada pada OOAD antara lain :

1. Object

Objek (*Object*) adalah “benda”, secara fisik atau konseptual, yang terdapat disekeliling kita. Sebuah obyek memiliki keadaan sesaat (*state*) dan prilaku (*behavior*). State dari sebuah obyek adalah kondisi objek tersebut atau himpunan dari keadaan yang menggambarkan objek tersebut. Sebagai contoh, salah satu state dari obyek jam adalah waktu saat ini. *State* dinyatakan dengan nilai dari atribut (*atribute*) objeknya. *Atribute* adalah nilai internal suatu objek, kondisi sesaat, koneksi dengan objek yang lain dan identitas. Perubahan state dicerminkan oleh perilaku (*behavior*) objek tersebut. Perilaku (*behavior*) suatu obyek mendefinisikan bagaimana suatu objek bertindak dan memberi reaksi. *Behavior* ditentukan oleh himpunan semua atau beberapa operasi yang dapat dilakukan dalam objek sendiri. *Behavior* dari sebuah objek dicerminkan oleh :

a. *interface*, adalah pintu untuk mengakses service objek.

b. *Service*, adalah fungsi yang bisa diemban oleh objek.

c. *Method*, adalah mekanisme internal objek yang mencerminkan perilaku (*behavior*) objek tersebut.

2. Kelas (Class)

Kelas (*class*) adalah definisi umum (pola, tamplate atau cetak biru) untuk himpunan objek sejenis. Kelas menetapkan spesifikasi perilaku (*behavior*) atau atribut objek – objek tersebut. Kelas adalah keniskalan atau abterasi dari entitas dalam dunia nyata. Objek adalah “contoh” (*instace*) dari sebuah kelas.

3. Encapsulation

Encapsulation adalah proses menyembunyikan detail implementasi sebuah objek. Untuk dapat berkomunikasi dengan objek diperlukan pesan (*message*). *Message* adalah permintaan untuk obyek penerimaan (*receiver object*) untuk membawa metode yang ditunjukkan atau perilaku dan mengembalikan result dari aksi tersebut kepada obyek pengirim (*sender object*).

4. Association dan Aggregation

Association adalah hubungan antara objek yang saling membutuhkan. *Aggregation* adalah bentuk khusus dari asosiasi yang menggambarkan seluruh bagian dari suatu objek dimana objek tersebut merupakan bagian dari objek lainnya.

1.6 *Unified Modelling Language* (UML)

Menurut Martin Fowler (2004, 1) pengertian Unified Modelling Language (UML) adalah keluarga notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain system perangkat lunak, khusunya system yang dibangun menggunakan pemrograman berorientasi object. Pada dasarnya UML dikembangkan dengan tujuan – tujuan sebagai berikut :

1. Untuk menyediakan pemodelan visual bagi pengguna, dimana model ini siap digunakan dan dapat saling bertukar model obyek yang dapat dimengerti.

2. Agar konsep dasarnya dapat dikembangkan dengan mudah dan dibuat lebih spesifik.

3. Tidak tergantung pada bahasa pemrograman dan proses pengembangan.

4. Menyediakan dasar yang formal dengan aturan – aturan dasar didalam membuat diagram guna memodelkan suatu objek.

5. Untuk mendorong pertumbuhan alat bantu yang berorientasi objek sehingga diharapkan akan muncul cukup banyak alat bantu yang memudahkan pengembangan software berorientasi obyek untuk menerapkan konsep *object oriented* dengan mudah.

6. Memudahkan pengembangan konsep dasar tanpa harus terpaku pada kode dalam bahasa pemrograman.

7. Memudahkan penggabungan beberapa abstraksi, sudut pandang, teknologi untuk membuat suatu sistem dari permasalahan yang dihadapi menjadi suatu sistem yang lebih kompleks.

8. Dengan UML diharapkan dapat menggambarkan keseluruhan sistem yang hendak dikembangkan dengan benar dan tepat tanpa harus mengurangi maksud dan informasi yang terkandung didalamnya.

9. Memudahkan untuk memodelkan suatu masalah dari berbagai sudut pandang.

10. Memudahkan jika nantinya hendak melakukan pembuatan atau pengembangan dari obyek yang lebih selesai diimplemantasikan.

11. Dapat digunakan sebagai alat untuk mendokumentasikan obyek – obyek yang telah dikembangkan sehingga mudah untuk dibaca dan dimengerti oleh pihak lainnya.

2.6.1 Diagram – diagram dalam UML

Menurut Martin Fowler (2004, 17), UML 2 terdiri dari 13 jenis diagram resmi. Pada akhirnya, jenis – jenis diagram bukanlah hal yang mutlak. Kita dapat secara legal menggunakan elemen – elemen dari satu jenis diagram untuk diagram yang yang lain. Oleh karena itu, penulis hanya menggunakan 3 jenis diagram UML yaitu :

1. Use Case Diagram

Diagram ini memperlihatkan bagaimana pengguna berinteraksi dengan sebuah sistem. Use case diagram menampilkan actor, use, dan hubungan antar mereka. Diagram ini sangat penting untuk mengorganisasi dan memodelkan perilaku sesuatu sistem yang dibutuhkan serta diharapkan pengguna.

Terdapat 2 jenis hubungan pada use case diagram, yaitu :

a) Bentuk <<include>> adalah stereotype ketergantungan. Hal ini menunjukkan bahwa jalur sebuah use case dimasukkan kedalam use case lain. Bentuk ini muncul saat use case melakukan berbagai langkah yang sama. Contoh dari mesin ATM adalah menarik uang dan melakukan transfer bersama – sama saat melakukan validasi nasabah. Pada use case terdahulu digunakan <<uses>>

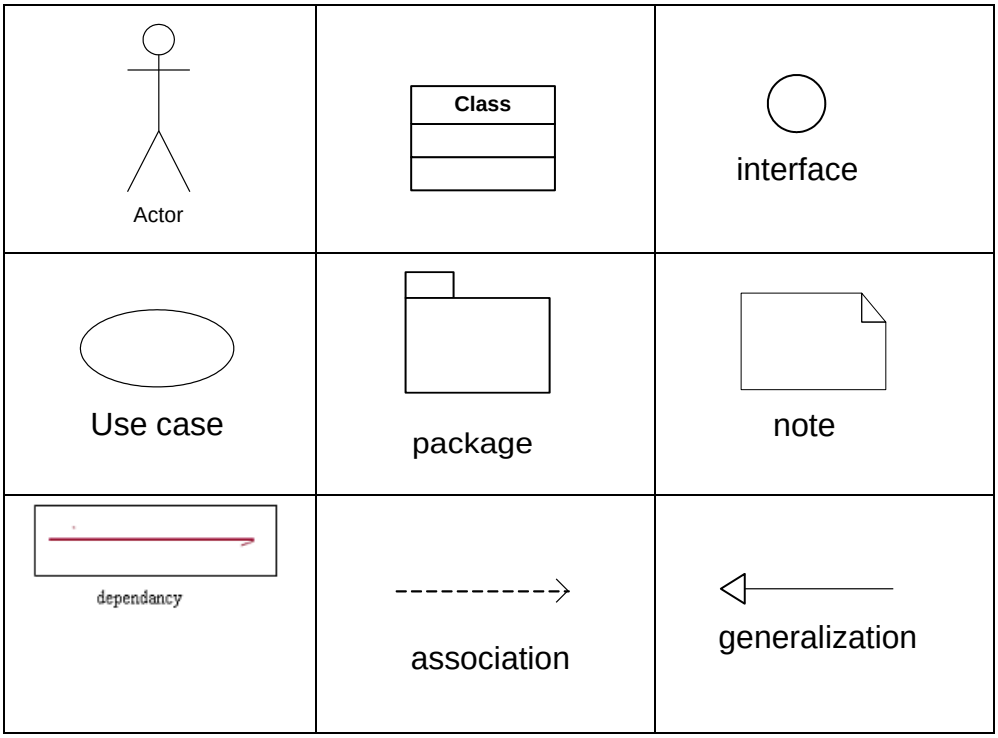
b) Bentuk <<extends>> merupkan sebuah stereotype sebuah ketergantungan. Hal ini memberi bentuk perluasan yang lebih terkontrol daripada hubungan generealisasi.

2. Activity Diagram

Activity diagram adalah teknik untuk menggambarkan logika procedural, proses bisni, dan jalur kerja. Diagram ini mirip dengan flowmap, tapi perbedaan prinsip antara diagram ini dan notasi flowmap adalah diagram ini mendukung behavior parallel. Dalam UML, activity diagram dapat menjelaskan kegiatan dan operasional komponen secara bertahap.

3. Sequence Diagram

Sequence diagram terkadang disebut sebagai event diagram, event scenario, dan timing diagram. Sequence diagram menjabarkan behavior sebuah scenario tunggal. Sequence diagram menunjukkan bagaimana proses operasi antara satu class dengan yang lainnya.



Gambar 2.1 Notasi pada UML

1.7 Java 2 Micro Edition (J2ME)

Menurut M. Salahuddin dan Rosa A.S (2010, 3) J2ME adalah lingkungan pengembangan yang didesain untuk meletakkan perangkat lunak Java pada barang elektronik beserta perangkat pendukungnya. J2ME biasanya digunakan pada telepon selular, pager, personal digital assistant (PDA) dan sejenisnya.