

**PAPER TENTANG UML
(UNIFIED MODELING LANGUAGE)**

Tugas ini disusun untuk melengkapi Tugas Mata Kuliah Rekayasa Perangkat Lunak



Disusun oleh
EKO SAPUTRA

**TEKNIK INFORMATIKA
PROGRAM STUDI ILMU KOMPUTER
UNIVERSITAS BINA DARMA PALEMBANG
2021**

DAFTAR ISI

DAFTAR ISI	i
KATA PENGANTAR	iv
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	1
1.3 Tujuan Penulisan Makalah.....	1
BAB II SISTEM INFORMASI	2
2.1 Sistem.....	2
2.2 Informasi.....	5
2.3 Sistem Informasi.....	6
2.4 Komponen Sistem Operasi.....	7
2.5 Elemen Sistem Informasi.....	9
2.6 Pengembangan Sistem Informasi.....	11
2.7 Jenis-Jenis Sistem Informasi.....	12
2.8 Analisis Sistem.....	14
2.8.1 Langkah-Langkah Analisa Sistem.....	15
2.8.2 Mengidentifikasi Masalah.....	15
2.8.3 Memahami Kerja Sistem.....	17
2.8.4 Menganalisis Hasil.....	19
2.8.5 Membuat Laporan Hasil Analisis.....	20
2.9 Analisis Berorientasi Objek.....	21
2.9.1 Metode Analisis Berorientasi Objek.....	22
2.9.2 Metode Analisis Berorientasi Objek Secara Umum.....	24
2.9.3 Identifikasi Kelas dan Objek.....	25
3.1 Definisi.....	26
3.2 Sejarah.....	26
3.3 Penggunaan UML Secara Resmi.....	27
BAB IV USE CASE DIAGRAM	28
4.1 Definisi.....	28
4.2 Karakteristik.....	28
4.3 Relasi dalam Use Case.....	29
4.4 Elemen – Elemen Diagram Use Case.....	30
4.4.1 Sistem.....	30
4.4.2 Actor.....	30
4.4.3 Use Case.....	31
4.4.4 Assosiasion.....	31
4.4.5 Stereotape.....	31

4.4.6	Dependency.....	31
4.5	Aplikasi UML.....	32
BAB V SYSTEM REQUIREMENT SPECIFICATION.....		34
5.1	Kebutuhan.....	34
5.2	Analisis Kebutuhan Sistem.....	34
5.3	Spesifikasi Kebutuhan Sistem (SyRS).....	35
5.4	Tujuan Pembuatan SyRS.....	35
5.5	Syarat Pembentukan SYRS.....	35
5.6	Atribut Penulisan SyRS.....	36
BAB VI CLASS DIAGRAM.....		38
6.1	Mendefinisikan Class Diagram.....	38
6.2	Cara Mendapatkan Kelas.....	39
6.3	Nilai Kardinalitas.....	39
BAB VII OBJEK DIAGRAM DAN INTERFACE.....		40
7.1	Objek Diagram.....	40
7.2	Interface dan Realisasi.....	40
BAB VIII INTERACTION DIAGRAM DAN ACTIVITY DIAGRAM.....		41
8.1	Interaction Diagram.....	41
8.2	Use Case Realization.....	42
8.3	Sequence Diagram.....	42
8.4	Collaburation.....	42
8.5	Activity Diagram.....	42
BAB IX DIAGRAM STATUS.....		43
9.1	Cara Mendefinisikan Statua dan Transisi.....	43
BAB X COMPONENT DIAGRAM DAN DEPLOYMENT DIAGRAM.....		45
10.1	Component Diagram.....	45
10.2	Cara Mendefinisikan Diagram Component.....	45
BAB XI PENUTUP.....		47
11.1	Kesimpulan.....	47
11.2	Saran.....	47
DAFTAR PUSTAKA.....		48

KATA PENGANTAR

Rasa syukur yang dalam saya sampaikan kehadiran Tuhan Yang Maha Esa, karena berkat kemurahannya paper “UML (UNIFIED MODELING LANGUAGE)” dapat saya selesaikan.

Paper ini dibuat sebagai referensi dan ilmu yang perlu kita ketahui . Dalam proses pendalaman materi ini, semua rekan-rekan saya yang telah banyak memberikan masukan untuk paper ini.

Demikian paper ini saya sampaikan, dan dengan penuh kesadaran masih adanya kekurangan-kekurangan dalam tulisan ini, mohon memakluminya dan bantuannya untuk memperbaiki segala kekurangannya.

Palembang, 23 November 2015
Penyusun
Eko Saputra

BAB I PENDAHULUAN

1.1 Latar Belakang

Sistem Informasi adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen. Dalam arti yang sangat luas, istilah sistem informasi yang sering digunakan merujuk kepada interaksi antara orang, proses algoritmik, data, dan teknologi. Dalam pengertian ini, istilah ini digunakan untuk merujuk tidak hanya pada penggunaan organisasi teknologi informasi dan komunikasi, tetapi juga untuk cara di mana orang berinteraksi dengan teknologi ini dalam mendukung proses bisnis. Sistem informasi yang berbeda dari teknologi informasi dalam sistem informasi biasanya terlihat seperti memiliki komponen.

Sistem Informasi sebagai tipe khusus dari sistem kerja adalah suatu sistem di mana manusia dan atau mesin melakukan pekerjaan dengan menggunakan sumber daya untuk memproduksi produk dan jasa bagi pelanggan.

1.2 Rumusan Masalah

Rumusan masalah dalam paper ini antara lain,

1. Bagaimana UML itu ?
2. Bagaimana gambaran dari UML ?
3. Bagaimana manfaat menggunakan UML ?

1.3 Tujuan Penulisan paper

Tujuan dari penulisan paper ini adalah,

1. Menjelaskan UML
2. Mengetahui UML
3. Memahami tujuan dari UML

BAB II SISTEM INFORMASI

2.1 Sistem

Sistem adalah suatu kesatuan yang terdiri komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi untuk mencapai suatu tujuan. Istilah ini sering dipergunakan untuk menggambarkan suatu set entitas yang berinteraksi.

Sistem juga merupakan kesatuan bagian-bagian yang saling berhubungan yang berada dalam suatu wilayah serta memiliki item-item penggerak, contoh umum misalnya seperti negara. Negara merupakan suatu kumpulan dari beberapa elemen kesatuan lain seperti provinsi yang saling berhubungan sehingga membentuk suatu negara dimana yang berperan sebagai penggeraknya yaitu rakyat yang berada dinegara tersebut.

Kata “sistem” banyak sekali digunakan dalam percakapan sehari-hari, dalam forum diskusi maupun dokumen ilmiah. Kata ini digunakan untuk banyak hal, dan pada banyak bidang pula, sehingga maknanya menjadi beragam. Dalam pengertian yang paling umum, sebuah sistem adalah sekumpulan benda yang memiliki hubungan di antara mereka.

Elemen sistem

Ada beberapa elemen yang membentuk sebuah sistem, yaitu : tujuan, masukan, proses, keluaran, batas, mekanisme pengendalian dan umpan balik serta lingkungan.

Berikut penjelasan mengenai elemen-elemen yang membentuk sebuah sistem :

1. Tujuan

Setiap sistem memiliki tujuan (Goal), entah hanya satu atau mungkin banyak. Tujuan inilah yang menjadi pemotivasi yang mengarahkan sistem. Tanpa tujuan, sistem menjadi tak terarah dan tak terkendali. Tentu saja, tujuan antara satu sistem dengan sistem yang lain berbeda.

2. Masukan

Masukan (input) sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan yang diproses. Masukan dapat berupa hal-hal yang berwujud (tampak secara fisik) maupun yang tidak tampak. Contoh masukan yang berwujud adalah bahan mentah, sedangkan contoh yang tidak berwujud adalah informasi (misalnya permintaan jasa pelanggan).

3. Proses

Proses merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna dan lebih bernilai, misalnya berupa informasi dan produk, tetapi juga bisa berupa hal-hal yang tidak berguna, misalnya saja sisa pembuangan atau limbah. Pada pabrik kimia, proses dapat berupa bahan mentah.

4. Keluaran

Keluaran (output) merupakan hasil dari pemrosesan. Pada sistem informasi, keluaran bisa berupa suatu informasi, saran, cetakan laporan, dan sebagainya.

5. Batas

Yang disebut batas (boundary) sistem adalah pemisah antara sistem dan daerah di luar sistem (lingkungan). Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem. Sebagai contoh, tim sepakbola mempunyai aturan permainan dan keterbatasan kemampuan pemain. Pertumbuhan sebuah toko kelontong dipengaruhi oleh pembelian pelanggan, gerakan pesaing dan keterbatasan dana dari bank. Tentu saja batas sebuah sistem dapat dikurangi atau dimodifikasi sehingga akan mengubah perilaku sistem. Sebagai contoh, dengan menjual saham ke publik, sebuah perusahaan dapat mengurangi keterbatasan dana.

6. Mekanisme Pengendalian dan Umpan Balik

Mekanisme pengendalian (control mechanism) diwujudkan dengan menggunakan umpan balik (feedback), yang mencuplik keluaran. Umpan balik ini digunakan untuk mengendalikan baik masukan maupun proses. Tujuannya adalah untuk mengatur agar sistem berjalan sesuai dengan tujuan.

7. Lingkungan

Lingkungan adalah segala sesuatu yang berada di luar sistem. Lingkungan bisa berpengaruh terhadap operasi sistem dalam arti bisa merugikan atau menguntungkan sistem itu sendiri. Lingkungan yang merugikan tentu saja harus ditahan dan dikendalikan supaya tidak mengganggu kelangsungan operasi sistem, sedangkan yang menguntungkan tetap harus terus dijaga, karena akan memacu terhadap kelangsungan hidup sistem.

2.2 Informasi

Informasi adalah pesan (ucapan atau ekspresi) atau kumpulan pesan yang terdiri dari order sekuens dari simbol, atau makna yang dapat ditafsirkan dari pesan atau kumpulan pesan. Informasi dapat direkam atau ditransmisikan. Hal ini dapat dicatat sebagai tanda-tanda, atau sebagai sinyal berdasarkan gelombang. Informasi adalah jenis acara yang mempengaruhi suatu negara dari sistem dinamis. Para konsep memiliki banyak arti lain dalam konteks yang berbeda. Informasi bisa di katakan sebagai pengetahuan yang didapatkan dari pembelajaran, pengalaman, atau

Namun, istilah ini memiliki banyak arti bergantung pada konteksnya, dan secara umum berhubungan erat dengan konsep seperti arti, pengetahuan, negentropy, Persepsi, Stimulus, komunikasi, kebenaran, representasi, dan rangsangan mental.

Dalam beberapa hal pengetahuan tentang peristiwa-peristiwa tertentu atau situasi yang telah dikumpulkan atau diterima melalui proses komunikasi, pengumpulan intelegen, ataupun didapatkan dari berita juga dinamakan informasi. Informasi yang berupa koleksi data dan fakta seringkali dinamakan informasi statistik. Dalam bidang ilmu komputer, informasi adalah data yang disimpan, diproses, atau ditransmisikan. Penelitian ini memfokuskan pada definisi informasi sebagai pengetahuan yang didapatkan dari pembelajaran, pengalaman, atau instruksi dan alirannya.

Informasi adalah data yang telah diberi makna melalui konteks. Sebagai contoh, dokumen berbentuk spreadsheet (semisal dari Microsoft Excel) seringkali digunakan untuk membuat informasi dari data yang ada di dalamnya. Laporan laba rugi dan neraca merupakan bentuk informasi, sementara angka-angka di dalamnya merupakan data yang telah diberi konteks sehingga menjadi punya makna dan manfaat.

2.3 Sistem Informasi

adalah kombinasi dari teknologi informasi dan aktivitas orang yang menggunakan teknologi itu untuk mendukung operasi dan manajemen. Dalam arti yang sangat luas, istilah sistem informasi yang sering digunakan merujuk kepada interaksi antara orang, proses algoritmik, data, dan teknologi. Dalam pengertian ini, istilah ini digunakan untuk merujuk tidak hanya pada penggunaan organisasi teknologi informasi dan komunikasi (TIK), tetapi juga untuk cara di mana orang berinteraksi dengan teknologi ini dalam mendukung proses bisnis.

Ada yang membuat perbedaan yang jelas antara sistem informasi, dan komputer sistem TIK, dan proses bisnis. Sistem informasi yang berbeda dari teknologi informasi dalam sistem informasi biasanya terlihat seperti memiliki komponen TIK. Hal ini terutama berkaitan dengan tujuan pemanfaatan teknologi informasi. Sistem informasi juga berbeda dari proses bisnis. Sistem informasi membantu untuk mengontrol kinerja proses bisnis.

Alter berpendapat untuk sistem informasi sebagai tipe khusus dari sistem kerja. Sistem kerja adalah suatu sistem di mana manusia dan atau mesin melakukan pekerjaan dengan menggunakan sumber daya untuk memproduksi produk tertentu dan/atau jasa bagi pelanggan. Sistem informasi adalah suatu sistem kerja yang kegiatannya ditujukan untuk pengolahan (menangkap, transmisi, menyimpan, mengambil, memanipulasi dan menampilkan) informasi.

Dengan demikian, sistem informasi antar-berhubungan dengan sistem data di satu sisi dan sistem aktivitas di sisi lain. Sistem informasi adalah suatu bentuk komunikasi sistem di mana data yang mewakili dan diproses sebagai bentuk dari memori sosial. Sistem informasi juga dapat dianggap sebagai bahasa semi formal yang mendukung manusia dalam pengambilan keputusan dan tindakan.

Sistem informasi merupakan fokus utama dari studi untuk disiplin sistem informasi dan organisasi informatika. Sistem informasi adalah gabungan yang terorganisasi dari manusia, perangkat lunak, perangkat keras, jaringan komunikasi dan sumber data dalam mengumpulkan, mengubah, dan menyebarkan informasi dalam organisasi.

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-laporan yang diperlukan.

2.4 Komponen Sistem Operasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (building blok), yang terdiri dari komponen input, komponen model, komponen output, komponen teknologi, komponen hardware, komponen software, komponen basis data, dan komponen kontrol. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran.

1. Komponen input

Input mewakili data yang masuk kedalam sistem informasi. Input disini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumendokumen dasar.

2. Komponen model

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

3. Komponen output

Hasil dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua pemakai sistem.

4. Komponen teknologi

Teknologi merupakan “tool box” dalam sistem informasi, Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, neghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan.

5. Komponen hardware

Hardware berperan penting sebagai suatu media penyimpanan vital bagi sistem informasi. Yang berfungsi sebagai tempat untuk menampung database atau lebih mudah dikatakan sebagai sumber data dan informasi untuk memperlancar dan mempermudah kerja dari sistem informasi.

6. Komponen software

Software berfungsi sebagai tempat untuk mengolah, menghitung dan memanipulasi data yang diambil dari hardware untuk menciptakan suatu informasi.

7. Komponen basis data

Basis data (database) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di pernagkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi

menggunakan perangkat lunak paket yang disebut DBMS (Database Management System).

8. Komponen kontrol

Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.5 Elemen Sistem Informasi

Sistem informasi terdiri dari elemen-elemen yang terdiri dari orang, prosedur, perangkat keras, perangkat lunak, basis data, jaringan komputer dan komunikasi data. Semua elemen ini merupakan komponen fisik :

1. Orang

Orang atau personil yang dimaksudkan yaitu operator komputer, analis sistem, programmer, personil data entry, dan manajer sistem informasi/EDP

2. Prosedur

Prosedur merupakan elemen fisik. Hal ini disebabkan karena prosedur disediakan dalam bentuk fisik seperti buku panduan dan instruksi. Ada 3 jenis prosedur yang dibutuhkan, yaitu instruksi untuk pemakai, instruksi untuk penyiapan masukan, instruksi pengoperasian untuk karyawan pusat komputer.

3. Perangkat keras

Perangkat keras bagi suatu sistem informasi terdiri atas komputer (pusat pengolah, unit masukan/keluaran), peralatan penyiapan data, dan terminal masukan/keluaran.

4. Perangkat lunak

Perangkat lunak dapat dibagi dalam 3 jenis utama :

- a. Sistem perangkat lunak umum, seperti sistem pengoperasian dan sistem manajemen data yang memungkinkan pengoperasian sistem komputer.
- b. Aplikasi perangkat lunak umum, seperti model analisis dan keputusan.
- c. Aplikasi perangkat lunak yang terdiri atas program yang secara spesifik dibuat untuk setiap aplikasi.

5. Basis data

File yang berisi program dan data dibuktikan dengan adanya media penyimpanan secara fisik seperti diskette, hard disk, magnetic tape, dan sebagainya. File juga meliputi keluaran tercetak dan catatan lain diatas kertas, mikro film, an lain sebagainya.

6. Jaringan komputer

Jaringan komputer adalah sebuah kumpulan komputer, printer dan peralatan lainnya yang terhubung dalam satu kesatuan. Informasi dan data bergerak melalui kabel-kabel atau tanpa kabel sehingga memungkinkan pengguna jaringan komputer dapat saling bertukar dokumen dan data.

7. Komunikasi data

Komunikasi data adalah merupakan bagian dari telekomunikasi yang secara khusus berkenaan dengan transmisi atau pemindahan data dan informasi diantara komputerkomputer dan pirant-piranti yang lain dalam bentuk digital yang dikirimkan melalui media komunikasi data. Data berarti informasi yang disajikan oleh isyarat digital. Komunikasi data merupakan bagian vital dari suatu sistem informasi karena sistem ini menyediakan infrastruktur yang memungkinkan komputer-komputer dapat berkomunikasi satu sama lain.

2.6 Pengembangan Sistem Informasi

Pengembangan sistem dapat berarti menyusun suatu sistem yang baru untuk menggantikan sistem yang lama secara keseluruhan atau memperbaiki sistem telah ada.

Sistem yang lama perlu diperbaiki atau diganti disebabkan beberapa hal, yaitu:

- a. Adanya permasalahan-permasalahan yang timbul di sistem yang lama, permasalahan yang timbul dapat berupa ketidakberesan, pertumbuhan organisasi,
- b. Untuk meraih kesempatan-kesempatan.
- c. Adanya instruksi-instruksi (dari pimpinan atau dari luar organisasi misalnya pemerintah).

Pengembangan sistem informasi yang berbasis komputer dapat merupakan tugas kompleks yang membutuhkan banyak sumber daya dan dapat memakan waktu yang lama untuk menyelesaikannya.

2.7 Jenis-Jenis Sistem Informasi

Sistem informasi dikembangkan untuk tujuan yang berbeda-beda, tergantung pada kebutuhan bisnis. Sistem informasi dapat dibagi menjadi beberapa bagian :

1. Transaction Processing Systems (TPS)

TPS adalah sistem informasi yang terkomputerisasi yang dikembangkan untuk memproses data dalam jumlah besar untuk transaksi bisnis rutin seperti daftar gaji dan inventarisasi. TPS berfungsi pada level organisasi yang memungkinkan organisasi bisa berinteraksi dengan lingkungan eksternal. Data yang dihasilkan oleh TPS dapat dilihat atau digunakan oleh manajer.

2. Office Automation Systems (OAS) dan Knowledge Work Systems (KWS)

OAS dan KWS bekerja pada level knowledge. OAS mendukung pekerja data, yang biasanya tidak menciptakan pengetahuan baru melainkan hanya menganalisis informasi sedemikian rupa untuk mentransformasikan data atau memanipulasikannya dengan cara-cara tertentu sebelum menyebarkannya secara keseluruhan dengan organisasi dan kadang-kadang diluar organisasi. Aspek-aspek OAS seperti word processing, spreadsheets, electronic scheduling, dan komunikasi melalui voice mail, email dan video conferencing.

KWS mendukung para pekerja profesional seperti ilmuwan, insinyur dan doktor dengan membantu menciptakan pengetahuan baru dan memungkinkan mereka mengkontribusikannya ke organisasi atau masyarakat.

3. Sistem Informasi Manajemen (SIM)

SIM tidak menggantikan TPS , tetapi mendukung spektrum tugas-tugas organisasional yang lebih luas dari TPS, termasuk analisis keputusan dan pembuat keputusan. SIM menghasilkan informasi yang digunakan untuk membuat keputusan, dan juga dapat membantu menyatukan beberapa fungsi informasi bisnis yang sudah terkomputerisasi (basis data).

4. Decision Support Systems (DSS)

DSS hampir sama dengan SIM karena menggunakan basis data sebagai sumber data. DSS bermula dari SIM karena menekankan pada fungsi mendukung pembuat keputusan diseluruh tahap-tahapnya, meskipun keputusan aktual tetap wewenang eksklusif pembuat keputusan.

5. Sistem Ahli (ES) dan Kecerdasan Buatan (AI)

AI dimaksudkan untuk mengembangkan mesin-mesin yang berfungsi secara cerdas. Dua cara untuk melakukan riset AI adalah memahami bahasa alamiahnya dan menganalisis kemampuannya untuk berfikir melalui problem sampai kesimpulan logiknya. Sistem ahli menggunakan pendekatan-pendekatan pemikiran AI untuk menyelesaikan masalah serta memberikannya lewat pengguna bisnis. Sistem ahli (juga disebut knowledge-based systems) secara efektif menangkap dan

menggunakan pengetahuan seorang ahli untuk menyelesaikan masalah yang dialami dalam suatu organisasi.

Berbeda dengan DSS, DSS meninggalkan keputusan terakhir bagi pembuat keputusan sedangkan sistem ahli menyeleksi solusi terbaik terhadap suatu masalah khusus. Komponen dasar sistem ahli adalah knowledge-base yakni suatu mesin inferensi yang menghubungkan pengguna dengan sistem melalui pengolahan pertanyaan lewat bahasa terstruktur dan antarmuka pengguna.

6. Group Decision Support Systems (GDSS) dan Computer-Support Collaborative Work Systems (CSCW)

Bila kelompok, perlu bekerja bersama-sama untuk membuat keputusan semi-terstruktur dan tak terstruktur, maka group Decision support systems membuat suatu solusi. GDSS dimaksudkan untuk membawa kelompok bersama-sama menyelesaikan masalah dengan memberi bantuan dalam bentuk pendapat, kuesioner, konsultasi dan skenario. Kadang-kadang GDSS disebut dengan CSCW yang mencakup pendukung perangkat lunak yang disebut dengan “groupware” untuk kolaborasi tim melalui komputer yang terhubung dengan jaringan.

7. Executive Support Systems (ESS)

ESS tergantung pada informasi yang dihasilkan TPS dan SIM dan ESS membantu eksekutif mengatur interaksinya dengan lingkungan eksternal dengan menyediakan grafik-grafik dan pendukung komunikasi di tempat-tempat yang bisa diakses seperti kantor.

2.8 Analisis Sistem

Analisa sistem dapat didefinisikan sebagai penguraian dari suatu system informasi yang utuh ke dalam bagian-bagian komponennya dengan maksud untuk mengidentifikasikan dan mengevaluasi permasalahan-permasalahan, kesempatan-kesempatan, hambatan yang terjadi dan kebutuhan-kebutuhan yang diharapkan sehingga dapat diusulkan perbaikan-perbaikannya.

Analisa Sistem Secara Umum : adalah Memandang, Pengamatan dan menyimpulkan konsep sistem berdasarkan Sistem Informasi secara fisik dan konseptual.

2.8.1 Langkah-Langkah Analisa Sistem

Langkah-langkah dalam tahap analisa sistem hampir sama dengan yang akan langkah-langkah yang dilakukan dalam mendefinisikan proyek-proyek sistem yang akan dikembangkan di tahap perencanaan sistem. Perbedaannya terletak pada ruang-lingkup tugasnya. Di analisa sistem, ruang lingkup tugasnya adalah lebih terinci. Di analisa sistem ini, penelitian yang dilakukan oleh analis sistem adalah penelitian terinci, sedang di perencanaan sistem sifatnya hanya penelitian pendahuluan

Di dalam tahap analisa sistem terdapat langkah-langkah dasar yang harus dilakukan oleh analis sistem, sebagai berikut:

- Identify, yaitu mengidentifikasikan masalah
- Understand, yaitu memahami kerja dari sistem yang ada
- Analyze, menganalisis sistem
- Report, yaitu membuat laporan hasil analisis.

2.8.2 Mengidentifikasi Masalah

Mengidentifikasi (mengenal) masalah merupakan langkah pertama yang dilakukan dalam tahap analisis sistem. Masalah dapat didefinisikan sebagai suatu pertanyaan yang diinginkan untuk dipecahkan. Masalah inilah yang menyebabkan sasaran dari sistem tidak dapat dicapai. Oleh karena itu langkah pertama yang harus dilakukan oleh analis sistem adalah mengidentifikasi terlebih dahulu masalah-masalah yang terjadi.

Tugas analis sistem dalam mengidentifikasi masalah adalah :

1. Mengidentifikasi Penyebab Masalah

Analisis sistem harus mempunyai pengetahuan yang cukup tentang aplikasi yang sedang dianalisisnya. Untuk aplikasi bisnis, analisis sistem perlu mempunyai pengetahuan tentang sistem bisnis yang diterapkan di organisasi, sehingga dapat mengidentifikasi penyebab terjadinya masalah ini.

Tugas mengidentifikasi penyebab masalah dimulai dengan mengkaji ulang terlebih dahulu subyek permasalahan yang telah diutarakan oleh manajemen atau yang telah ditemukan oleh analis sistem ditahap perencanaan sistem.

2. Mengidentifikasi Titik Keputusan

Setelah penyebab terjadinya masalah dapat diidentifikasi, selanjutnya juga harus mengidentifikasikan titik keputusan penyebab masalah tersebut. Titik keputusan menunjukkan suatu kondisi yang menyebabkan sesuatu terjadi.

Analisis sistem bila telah dapat mengidentifikasi terlebih dahulu titik-titik keputusan penyebab masalah, maka dapat memulai penelitiannya dititik-titik keputusan tersebut. Sebagai dasar identifikasi titik-titik keputusan ini, dapat digunakan dokumen paperwork flow atau form flowchart bila dokumentasi ini dimiliki oleh perusahaan.

3. Mengidentifikasi Personil-personil Kunci

Setelah titik-titik keputusan penyebab masalah dapat diidentifikasi beserta lokasi terjadinya, maka selanjutnya yang perlu diidentifikasi adalah personil-personil kunci baik yang langsung maupun yang tidak langsung dapat menyebabkan terjadinya masalah tersebut. Identifikasi personil-personil kunci ini dapat dilakukan dengan mengacu pada bagan alir dokumen perusahaan serta dokumen deskripsi kerja (job description).

2.8.3 Memahami Kerja Sistem

Langkah ini dapat dilakukan dengan mempelajari secara terinci bagaimana sistem yang ada beroperasi. Diperlukan data yang dapat diperoleh dengan cara melakukan penelitian. Bila di tahap perencanaan sudah pernah diadakan penelitian, sifatnya masih penelitian pendahuluan (preliminary survey). Sedangkan pada tahap analisis sistem, penelitiannya bersifat penelitian terinci (detailed survey).

Analisis sistem perlu mempelajari apa dan bagaimana operasi dari sistem yang ada sebelum mencoba untuk menganalisis permasalahan, kelemahan dan kebutuhan pemakai sistem untuk dapat memberikan rekomendasi pemecahannya. Sejumlah data perlu dikumpulkan, dengan menggunakan teknik pengumpulan data yang ada, yaitu wawancara, observasi, daftar pertanyaan dan pengambilan sampel.

Tugas analisis sistem Dalam Memahami Kerja Sistem:

1. Menentukan Jenis Penelitian

Jenis penelitian perlu ditentukan untuk masing-masing titik keputusan yang akan diteliti. Jenis penelitian tergantung dari jenis data yang diperoleh, dapat berupa data tentang operasi sistem, data tentang perlengkapan sistem, pengendalian sistem, atau I/O yang digunakan oleh sistem.

2. Merencanakan Jadwal Penelitian

Supaya penelitian dapat dilakukan secara efisien dan efektif, maka jadwal penelitian harus direncanakan terlebih dahulu yang meliputi :

Dimana penelitian akan dilakukan. Apa dan siapa yang akan diteliti, siapa yang akan meneliti, kapan penelitian dilakukan.

3. Membuat Penugasan Penelitian

Setelah rencana jadwal penelitian dibuat, maka tugas dilanjutkan dengan menentukan tugas dari masing-masing anggota tim analisis sistem, yang ditentukan oleh koordinator analisis sistem melalui surat penugasan dengan menyertakan lampiran kegiatan penelitian yang harus dilakukan.

4. Membuat Agenda Wawancara

Sebelum wawancara dilakukan, waktu dan materi wawancara perlu didiskusikan. Rencana ini dapat ditulis di agenda wawancara dan dibawa selama wawancara berlangsung. Tujuannya adalah supaya wawancara dapat diselesaikan tepat pada waktunya dan tidak ada materi yang terlewatkan.

5. Mengumpulkan Hasil Penelitian

Fakta atau data yang diperoleh dari hasil penelitian harus dikumpulkan sebagai suatu dokumentasi sistem lama, yaitu :

- Waktu untuk melakukan suatu kegiatan
- Kesalahan melakukan kegiatan di sistem yang lama
- Pengambilan sampel

- Formulir dan laporan yang dihasilkan oleh sistem lama
- Elemen-elemen data
- Teknologi yang digunakan di sistem lama
- Kebutuhan informasi pemakai sistem / manajemen

2.8.4 Menganalisis Hasil

Langkah ini dilakukan berdasarkan data yang telah diperoleh dari hasil penelitian yang telah dilakukan.

1. Menganalisis Kelemahan Sistem

Penelitian dilakukan untuk menjawab pertanyaan :

- Apa yang dikerjakan ?
- Bagaimana mengerjakannya ?
- Siapa yang mengerjakan ?

Menganalisis kelemahan sistem sebaliknya dilakukan untuk menjawab pertanyaan :

Mengapa dikerjakan ?

Perluakah dikerjakan ?

Apakah telah dikerjakan dengan baik ?

Sasaran yang diinginkan oleh sistem yang baru ditentukan oleh kriteria penilaian sebagai berikut :

- Relevance,
- Capacity,
- Efficiency,
- Timeliness,
- Accessibility,
- Flexibility,
- Accuracy,
- Reliability,
- Security,
- Economy,
- Simplicity

Berdasarkan pertanyaan dan kriteria ini, selanjutnya analisis sistem akan dapat melakukan analisis dari hasil penelitian dengan baik untuk menemukan kelemahan dan permasalahan dari sistem yang ada.

2. Menganalisis Kebutuhan Informasi Pemakai / Manajemen

Tugas lain dari analisis sistem yang diperlukan sehubungan dengan sasaran utama sistem informasi, yaitu menyediakan informasi yang dibutuhkan bagi para pemakainya perlu dianalisis.

2.8.5 Membuat Laporan Hasil Analisis

Laporan hasil analisis diserahkan ke Panitia Pengarah (Steering Committee) yang nantinya akan diteruskan ke manajemen. Pihak manajemen

bersama-sama dengan panitia pengarah dan pemakai sistem akan mempelajari temuan-temuan dan analisis yang telah dilakukan oleh analisis sistem yang disajikan dalam laporan ini.

Tujuan utama dari penyerahan laporan ini kepada manajemen adalah :

- Analisis telah selesai dilakukan
- Meluruskan kesalah-pengertian mengenai apa yang telah ditemukan dan dianalisis oleh analisis sistem tetapi tidak sesuai menurut manajemen
- Meminta pendapat dan saran dari pihak manajemen
- Meminta persetujuan kepada pihak manajemen untuk melakukan tindakan selanjutnya (dapat berupa meneruskan ke tahap disain sistem atau menghentikan proyek bila dipandang tidak layak lagi)
- Semua hasil yang didapat dari penelitian perlu dilampirkan pada laporan hasil analisis ini, sehingga manajemen dan user dapat memeriksa kembali kebenaran data yang telah diperoleh.

2.9 Analisis Berorientasi Objek

Investigasi masalah untuk menemukan (mengidentifikasi) dan mendefinisikan objek-objek atau konsep-konsep yang ada di ruang masalah. Proses untuk menentukan objek-objek potensial yang ada dalam sistem dan mendeskripsikan karakteristik dan hubungannya dalam sebuah notasi formal. Aplikasi konsep berorientasi objek untuk memodelkan permasalahan dan sistem, baik untuk lingkup perangkat lunak maupun non-perangkat lunak.

Tujuan Analisis

- Memahami permasalahan secara menyeluruh.
- Mengungkapkan apa yang harus dikerjakan oleh sistem untuk memenuhi kebutuhan pemakai.
- Mengetahui ruang lingkup produk (product space) dan pemakai yang akan menggunakan produk tersebut.

Tahap Analisis

- Mempelajari permasalahan
- Menentukan kebutuhan pemakai
- Mengubah kebutuhan yang belum terstruktur menjadi model-model atau gambar-gambar dengan memanfaatkan metode dan teknik analisis tertentu.
- Mendokumentasikan hasil analisis, misalnya Software Requirement Specification (SRS).

2.9.1 Metode Analisis Berorientasi Objek

Cara kerja yang sistematis untuk mengerjakan tahap analisis berdasarkan pendekatan objek. Ada kumpulan aturan-aturan tertentu yang harus diikuti untuk menyelesaikan pekerjaan analisis tersebut.

Mempunyai urutan aktivitas, teknik, dan alat bantu (tools) tertentu untuk memodelkan (mendokumentasikan) hasil dari setiap aktivitas. Ada beberapa metode yang dapat digunakan untuk melakukan analisis berorientasi objek, dan diantaranya adalah sebagai berikut.

Metode Coad & Yourdan

Diperkenalkan oleh Peter Coad dan Edward Yourdan pada tahun 1990. Disebut juga dengan nama Object Oriented Analysis (OOA), dan dipandang sebagai salah satu teknik yang mudah untuk dipelajari. Notasi model relatif sederhana karena didasarkan pada struktur fisik dunia nyata, dan petunjuk untuk melakukan analisis cukup jelas.

Tahap atau skema pelaksanaan:

- Identifikasi kelas dan objek
- Identifikasi struktur

Struktur "generalization-specification", Struktur "whole-part" atau "a-part-of"

Identifikasi subjek

Definisikan atribut

- Atribut implisi objek
- Koneksi instan (instance connection)

Definisikan layanan

- Layanan implisit objek
- Layanan yang berasosiasi dengan atribut
- Layanan yang berasosiasi dengan “message-connection”

Metode Rumbaugh

Diperkenalkan oleh James Rumbaugh, Michael Blaha, William Premerlan, Frederick Eddy dan William Lorensen pada tahun 1991.

Lebih dikenal dengan Object Modeling Technique (OMT) yang dapat digunakan baik untuk analisis maupun desain. Selain model-model fisik dari objek, pendekatan analisis dilakukan juga untuk model-model dinamik dan model fungsional.

Metode Jacobson

Diperkenalkan oleh Ivar Jacobson dengan nama Object Oriented Software Engineering (OOSE) pada tahun 1992. Merupakan versi yang juga sederhana dari metode berorientasi objek. Sudut pandang atau fokus analisis ditekankan pada “use case”, yaitu deskripsi atau skenario yang menggambarkan bagaimana pemakai berinteraksi dengan produk atau sistem yang akan dikembangkan.

Metode Booch

Diperkenalkan oleh Grady Booch pada tahun 1994. Meliputi proses pengembangan makro dan mikro, dengan anggapan bahwa analisis dan desain merupakan rangkaian kesatuan aktivitas yang tidak dipisahkan.

2.9.2 Metode Analisis Berorientasi Objek Secara Umum

Pada prinsipnya semua metode analisis berorientasi objek adalah sama, perbedaan hanya terletak pada sudut pandang dan teknis pelaksanaannya. Secara umum, metode analisis berorientasi objek mencakup representasi kelas dan hirarki kelas, model hubungan objek, dan model perilaku objek.

Tahap atau skema pelaksanaan analisis berorientasi objek :

- Tentukan kebutuhan pemakai untuk sistem berorientasi objek
- Identifikasi kelas dan objek
- Identifikasi atribut dan layanan untuk setiap objek
- Definisikan struktur dan hirarki
- Buat model hubungan objek
- Buat model perilaku objek

Menentukan Kebutuhan Pemakai untuk Sistem Berorientasi Objek

- Mengidentifikasi proses-proses bisnis dan kebutuhan pemakai dan mengekspresikan dengan ‘use-case’.
- Sebenarnya bukan merupakan aktivitas analisis berorientasi objek, karena tidak membicarakan pembahasan tentang objek.
- Diperlukan karena dapat menjelaskan aktivitas-aktivitas apa saja yang harus dikerjakan oleh sistem, dan menjelaskan juga perilaku dari komponen-komponen sistem.
- Ada diagram tertentu yang dapat merepresentasikan model kebutuhan dari “use-case” yang diperoleh.

2.9.3 Identifikasi Kelas dan Objek

Mengidentifikasi kelas-kelas dan objek-objek yang ada dalam lingkup aplikasi:

- eksplisit pada pernyataan masalah.
- implisit pada lingkup aplikasi atau pengetahuan atas lingkup aplikasi.

Kelas dan objek dapat diidentifikasi dari:

- entitas eksternal yang memproduksi dan memakai informasi yang akan digunakan oleh sistem berbasis komputer
- sesuatu yang merupakan bagian dari wilayah informasi dari permasalahan
- kejadian, misalnya prosedur operasional, yang muncul dalam lingkup operasional sistem .
- peran yang dimainkan oleh orang- orang yang berinteraksi dengan sistem .
- unit organisasi yang relevan dengan aplikasi
- tempat yang menentukan ruang lingkup masalah dan seluruh fungsi dari sistem .
- struktur yang mendefinisikan kelas dari objek atau yang menghubungkan kelas-kelas objek.

BAB III UNIFIED MODELING LANGUAGE

3.1 Definisi

UML (Unified Modeling Language) adalah bahasa yang berdasarkan grafik atau gambar untuk memvisualisasi, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan software berbasis OO (Object Oriented).

UML sendiri juga memberikan standar penulisan sebuah sistem blue print, yang meliputi konsep bisnis proses, penulisan kelas – kelas dalam bahasa program yang spesifik, skema database dan komponen – komponen yang diperlukan dalam software.

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 sebuah sistem.

Dengan menggunakan UML kita dapat membuat model untuk semua jenis piranti lunak, dimana aplikasi tersebut dapat berjalan pada piranti keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun

3.2 Sejarah

Pendekatan analisa dan rancangan dengan menggunakan model OO mulai diperkenalkan sekitar pertengahan tahun 1970 hingga akhir 1980 dikarenakan pada saat itu aplikasi software sudah meningkat dan mulai kompleks. Jumlah yang menggunakan metode OO mulai diuji cobakan dan diaplikasikan antara tahun 1989 – 1994, seperti halnya oleh grady Booch dari Rational Software Co., serta James Rumbaugh dari General Electric, dikenal dengan OMT (Object Modelling Technique).

Kelemahan saat itu disadari oleh Booch maupun Rumbaugh adalah tidak adanya standar penggunaan model yang berbasis OO, ketika mereka bertemu ditemani rekan lainnya Ivar Jacobson dari Objectory mulai mendiskusikan untuk mengadopsi masing-masing pendekatan metode OO untuk membuat suatu model bahasa yang uniform atau seragam yang disebut UML (Unified Modeling language) dan dapat digunakan oleh seluruh dunia.

3.3 Penggunaan UML Secara Resmi

Secara resmi bahasa UML dimulai pada bulan oktober 1994, ketika Rumbaugh bergabung dengan Booch untuk membuat sebuah project pendekatan metode yang uniform atau seragam dari masing – masing metode mereka. Saat itu baru dikembangkan draft metode UML version 0.8 dan diselesaikan serta direlease pada bulan oktober 1995. Bersamaan dengan saat itu, Jacobson bergabung dan UML tersebut diperkaya ruang lingkupnya dengan metode OOSE sehingga muncul release version 0.9 pada bulan juni 1996. Hingga saat ini sejak Juni 1998 UML version 1.3 telah diperkaya dan direspons oleh OMG (Object Management Group). Anderson Consulting, Ericson, Platinum technology,

ObjectTime Limited dan lain lain serta dipelihara oleh OMG yang dipimpin oleh Cris Kobryn.

UML adalah standar dunia yang dibuat oleh Object Management Group, sebuah badan yang bertugas mengeluarkan standar-standar teknologi objectoriented dan software component.

BAB IV USE CASE DIAGRAM

4.1 Definisi

Use case class digunakan untuk memodelkan dan menyatakan unit fungsi/layanan yang disediakan oleh sistem (or bagian sistem: subsistem atau class) ke pemakai. Use case dapat dilingkupi dengan batasan sistem yang diberi label nama sistem. Use case adalah sesuatu yang menyediakan hasil yang dapat diukur ke pemakai atau sistem eksternal.

4.2 Karakteristik

Diagram Use case sendiri memiliki karakteristik sebagai berikut :

- Use cases adalah interaksi atau dialog antara sistem dan actor, termasuk pertukaran pesan dan tindakan yang dilakukan oleh sistem.
- Use cases diprakarsai oleh actor dan mungkin melibatkan peran actor lain. Use cases harus menyediakan nilai minimal kepada satu actor.
- Use cases bisa memiliki perluasan yang mendefinisikan tindakan khusus dalam interaksi atau use case lain mungkin disisipkan.
- Use case class memiliki objek use case yang disebut skenario. Skenario menyatakan urutan pesan dan tindakan tunggal.

4.3 Relasi dalam Use Case

Ada beberapa relasi yang terdapat pada use case diagram:

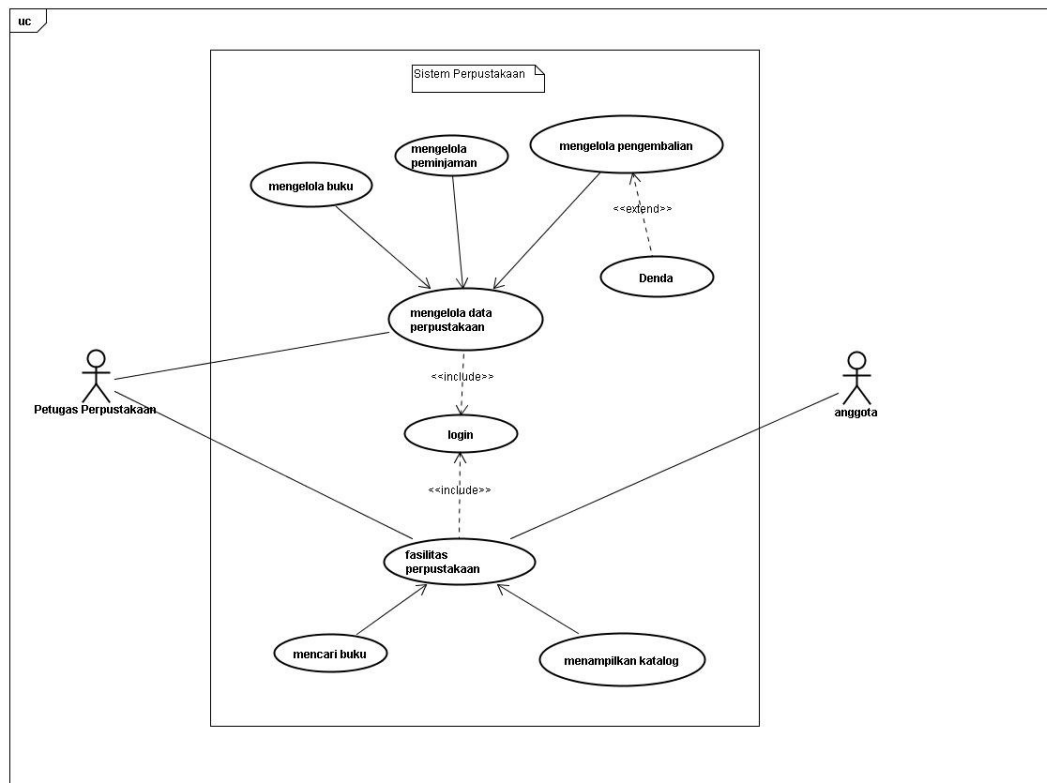
1. Association, menghubungkan link antar element.
2. Generalization, disebut juga inheritance (pewarisan), sebuah elemen dapat merupakan spesialisasi dari elemen lainnya.
3. Dependency, sebuah element bergantung dalam beberapa cara ke element lainnya.
4. Aggregation, bentuk assosiation dimana sebuah elemen berisi elemen lainnya.

Tipe relasi (stereotype) yang mungkin terjadi pada use case diagram:

1. <<include>> , yaitu kelakuan yang harus terpenuhi agar sebuah event dapat terjadi, dimana pada kondisi ini sebuah use case adalah bagian dari use case lainnya.
2. <<extends>>, kelakuan yang hanya berjalan di bawah kondisi tertentu seperti menggerakkan alarm.

3. <<communicates>>, mungkin ditambahkan untuk asosiasi yang menunjukkan asosiasinya adalah communicates association . Ini merupakan pilihan selama asosiasi hanya tipe relationship yang dibolehkan antara actor dan use case.

Contoh Use Case Diagram



4.4 Elemen – Elemen Diagram Use Case

4.4.1 Sistem

sistem menyatakan batasan sistem dalam relasi dengan actor yang menggunakannya (diluar sistem) dan fitur-fitur yang harus disediakan (di dalam sistem)

sistem digambarkan dengan segi empat yang membatasi semua use case dalam sistem terhadap pihak mana sistem akan berinteraksi. Sistem disertai label yang menyebutkan nama dari sistem, tapi umumnya tidak digambarkan karena tidak terlalu memberi arti dalam diagram use case.

4.4.2 Actor

Actor dapat berupa merupakan manusia, sistem atau device yang memiliki peranan dalam keberhasilan operasi dari sistem.

4.4.3 Use Case

Mengidentifikasi fitur utama dari sistem , setiap use case menunjukkan tujuan utama dari sistem, dan memiliki nama sesuai dari apa yang menjadi tujuannya di gambar dengan bentuk elips.

4.4.4 Assosiation

Manandai interaksi anatar aktor dengan use case yang bersangkutan asosiasi bisa saja memiliki arah atau satu arah , namun pada kebanyakan kasus asosiasi biasanya memiliki 2 arah (tanpa tanda anak panah) karena pada umumnya seperti itu

4.4.5 Stereotype

Dapat digunakan untuk memperluas cakupan pada UML tanpa merubahnya , memiliki peran sebagai kualifier pada sebuah elemen model, yang memiliki informasi yang banyak tanpa disebutkan implementasinya , utamanya digunakan untuk menggambarkan :

- Use case dependency
- Class-class
- Package
- Classifier

4.4.6 Dependency

Berfungsi sebagai :

- Mengidentifikasi antara 2 use case
- Memudahkan pemeliharaan dengan memisahkan use case yang memiliki aktivitas yang sama.
- Digambarkan dengan garis putus-putus dengan notasi <<include>> pada garis.
- Arah mata panah menuju pemanggil.
- Depedensi <<extend>>.
 - a. Jika memerlukan kondisi tertentu berlaku depedensi.
 - b. Penggambaran sama dengan <<include>>, namun arah panah berbeda

4.4.7 generalization

mendefinisikan relasi antara 2 aktor atau 2 use case , penggambarannya menggunakan garis bermata panah kosong, yang meng-inherit mengarah ke yang di-inherit.

4.5 Aplikasi UML

Baiklah UML memang banyak digunakan untuk membuat gambaran tentang sistem yang akan dibuat , karena dengan diagram diagram dari UML ini dapat memperjelas bagaimana alur dari suatu sistem. berikut adalah program atau aplikasi yang berguna untuk membuat diagram UML.

Argo UML

Argo UML ini adalah software gratis dan tidak berbayar , dan juga mudah digunakan untuk membuat UML , dan argo UML bisa dijalankan disemua platform/operating sistem computer. jika belum mempunyai aplikasi dan ingin mendownload aplikasi Argo UML,

Star UML

Star UML ini juga termasuk dalam software gratis dan tidak berbayar, penggunaannya pun juga mudah menggunakan bahkan menurut saya lebih mudah dari semua aplikasi pembuat uml.

BAB V SYSTEM REQUIREMENT SPECIFICATION

5.1 Kebutuhan

Bahwasanya, kebutuhan sistem adalah kondisi, kriteria, syarat atau kemampuan yang harus dimiliki oleh sistem untuk memenuhi apa yang disyaratkan atau diinginkan pemakai.

Secara kategoris, ada 3 buah jenis kebutuhan sistem :

1. Kebutuhan fungsional
2. Kebutuhan antar-muka
3. Kebutuhan ujuk kerja

5.2 Analisis Kebutuhan Sistem

Analisis kebutuhan dapat diartikan sebagai proses mempelajari kebutuhan pemakai untuk mendapatkan definisi kebutuhan sistem atau sistem itu sendiri. Pada tahap ini, seorang analis akan mempelajari masalah yang ada pada sistem yang sedang dikembangkan.

Cara yang digunakan oleh pengembang, khususnya analis, dalam memahami masalah sistem biasanya dilakukan :

1. Wawancara dengan pemakai
2. Pengamatan lapangan
3. Mempelajari dokumen-dokumen yang digunakan tentang analisa dan perancangan sistem

Pada tahap ini, kebutuhan pemakai yang belum terstruktur akan dianalisis menjadi kebutuhan fungsional, antar-muka, dan unjuk kerja sesuai dengan kebutuhan. Selanjutnya kebutuhan tersebut akan dimodelkan dengan alat bantu tertentu, contohnya menggunakan :

1. Data flow diagram, kamus, data, dan spesifikasi proses jika menggunakan analisis terstruktur
2. Use case diagram dan skenario sistem jika menggunakan analisis berorientasi objek

5.3 Spesifikasi Kebutuhan Sistem (SyRS)

Suatu dokumentasi yang saling melengkapi dari pernyataan lengkap yang dilakukan sistem tanpa menjelaskan “bagaimana” hal tersebut dikerjakan.

5.4 Tujuan Pembuatan SyRS

Tujuan penulisan SyRS adalah mendefinisikan keinginan yang biasanya dinyatakan dalam bentuk penjelasan umum.

Berikut manfaat dan kegunaan SyRS :

1. Memastikan kesamaan antara kebutuhan untuk pengembangan dengan kebutuhan yang ditulis di dalam dokumen.

2. Menjelaskan antar-muka bagi para pihak yang terlibat dalam proses pengembangan sistem.
3. Mengenali tugas, tahapan, aktivitas kaji ulang, dan dokumentasinya.
4. Menghilangkan masalah yang sudah terjadi sebelumnya.

5.5 Syarat Pembentukan SYRS

Ada empat syarat dalam pembentukan SyRS yaitu :

1. Mudah dipelajari
2. Diuraikan dengan jelas, mudah, dan sederhana.
3. Bisa divalidasi dan bisa diuji.
4. Mampu untuk ditelusuri kembali.

Adapun yang harus dihindari dalam pembuatan SyRS adalah :

1. Over specification
2. Tindakan unconcistency
3. Ambiguity dalam kata atau kalimat
4. Menuliskan hal yang tidak perlu ditulis.

Dalam suatu SyRS ada 2 aspek yang harus terlihat, yaitu :

1. Fungsi
Menjelaskan bagaimana sistem digunakan serta manfaatnya.
2. Non-fungsi
Dependability, ergonomic, perform ance, constraint.

5.6 Atribut Penulisan SyRS

Dokumen SyRS yang baik dapat dibuat secara :

1. Benar
2. Tepat
3. Unambiguouity
4. Lengkap
5. Bisa diverifikasi
6. Konsisten
7. Understandable
8. Bisa dimodifikasi
9. Dapat ditelusuri
10. Dapat dibedakan dibagian spesifikasi
11. Dapat melingkupi semua lingkungan operasional
12. Sistemnya dapat digambarkan
13. Dapat diterima terhadap ketidaklengkapan dan ketidakkonsistenan.
14. Harus bisa dilokalisasi dengan hubungan antar model.

Berikut adalah macam orang yang terlibat dalam pembuatan SyRS :

1. Pemakai
2. Client
3. Sistem analis
4. Software engineer
5. Programmer

6. Test integration group
7. Maintenance group
8. Technical group
9. Staff dan clerical work

BAB VI CLASS DIAGRAM

Diagram kelas ini akan membahas tentang objek yang berada dalam sistem dan berbagai jenis hubungan statis yang ada disekitarnya yang memiliki karakteristik yang sama seperti atribut, operasi hubungan, dan semantik.

6.1 Mendefinisikan Class Diagram

Kelas (class) adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan perancangan orientasi objek.

Dalam pemodelan statis, diagram kelas digunakan salah satunya untuk memodelkan dari 3 hal berikut :

1. Perbendaharaan dari sistem
2. Kolaborasi
3. Skema basis data logical

Kelas memiliki tiga area pokok :

1. Nama
2. Atribut
3. Metode atau operasi

Atribut dan metode dapat memiliki salah satu sifat yaitu :

1. Private
Tidak dapat dipanggil dari luar kelas yang bersangkutan
2. Protected
Hanya dapat dipanggil dari kelas yang bersangkutan
3. Public
Dapat dipanggil oleh siapa saja

Kelas dapat berupa implementasi yang hanya memiliki metode, sehingga dapat membentuk interface yang mendukung resolusi metode pada saat run-time.

6.2 Cara Mendapatkan Kelas

Berikut ini beberapa alternatif cara mendefinisikan kandidat kelas :

1. Mengidentifikasi kelas-kelas dan objek-objek yang ada dalam lingkup sistem
2. Kelas dan objek dapat diidentifikasi dari salah satu atau lebih dari segi entitas eksternal dari dalam lingkup operasional sistem data

6.3 Nilai Kardinalitas

Nilai kardinalitas atau mutiplicity menunjukkan jumlah suatu objek yang dapat berhubungan dengan objek lain yang ditujukan dengan “satu” atau “banyak” dengan bilangan integer lebih besar atau sama dengan nol.

Indikator	Arti
0..1	Nol atau satu
1	Hanya satu

0..*	Nol atau lebih
1..*	Satu atau lebih
N	Hanya n (dengan $n > 1$)
0..n	Nol sampai n (dengan $n > 1$)
1..n	Satu sampai n (dengan $n > 1$)

Jenis-jenis mutplicity

BAB VII OBJEK DIAGRAM DAN INTERFACE

Object diagram adalah diagram yang memberikan pendapat dari sebuah kelas. Diagram ini dapat memeriksa keabsahan kelas-kelas diagram berikut aturan-aturan multiplitasnya dengan “real data” dan mengujinya dengan skenario-skenario tertentu.

7.1 Objek Diagram

Objek diagram juga dapat mengandung asosiasi yang biasanya constraint, detil relationship, multiplisitas yang ada di class diagram tidak disertakan dalam objek diagram sebagai upaya memfokuskan perhatian hanya terhadap objek dan property.

Asosiasi antar dua buah objek biasanya dinotasikan dengan sebuah garis yang menghubungkan kedua objek diagram yang digunakan untuk menggambarkan sebuah sistem pada sebuah sudut pandang waktu tertentu.

Objek diagram merupakan :

1. Deskripsi atas objek dan hubungannya dalam urutan waktu, umumnya sebuah special case dari class diagram atau commnication diagram.
2. Deskripsi sebagai sample diagram yang menggambarkan suatu relasi diantaranya.
3. Hubungan antara objek digambarkan dengan sebuah garis tanpa panah yang berisi sebuah peran yang boleh atau tidak untuk dipakai.

7.2 Interface dan Realisasi

Interface adalah satu set operation yang memberikan spesifikasi beberapa aspek dari perilaku dan operasi disuatu kelas ke kelas yang lain. Relasi antara class dan interface disebut realization.

BAB VIII INTERACTION DIAGRAM DAN ACTIVITY DIAGRAM

8.1 Interaction Diagram

Interaction diagram adalah model yang menjelaskan bagaimana sejumlah objek dapat bekerja sama dalam beberapa tindakan yang menerangkan tentang suatu kelakuan dari suatu case yang digunakan. Diagram ini menggambarkan sejumlah objek dan pesan yang dijalankan antara objek dengan case yang digunakan, ketika kita memberi pesan, aksi yang dihasilkan adalah sebuah pernyataan tereksekusi yang membentuk abstraksi dari prosedur komputasi. Dalam UML dapat memodelkan beberapa jenis tindakan berikut :

1. Call
Memanggil operasi yang ada pada objek yang juga memungkinkan objek sedang mengirim dirinya sendiri hingga menghasilkan pemanggilan lokal dari sistem operasi.
2. Return
Mengembalikan nilai dari return.
3. Send
Mengirimkan sinyal ke objek.
4. Create
Membuat sebuah objek.
5. Destroy
Mematikan sebuah objek yang memungkinkan mematikan dirinya sendiri.

Interaction diagram digunakan ketika kita ingin mengetahui kelakuan dari beberapa objek dalam use case tunggal yang bekerja baik saat menunjukkan kolaborasi dengan objek-objek, namun kurang baik dalam mendefinisikan behavior. Ada dua macam interaction diagram, yaitu sequence diagram dan collaboration diagram.

8.2 Use Case Realization

Skenario digunakan untuk menggambarkan bagaimana use case-use case direalisasikan sebagai interaksi antara objek-objek. Untuk menggambarkan realisasinya dapat digunakan beberapa diagram.

8.3 Sequence Diagram

Diagram ini menggambarkan tentang interaksi antara sejumlah objek dalam urutan waktu untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antar objek yang terjadi pada titik tertentu dalam eksekusi sistem.

8.4 Collaboration

Collaboration diagram merupakan cara alternatif untuk menggambarkan skenario dari suatu sistem dimana interaksi objek diatur oleh objek sekelilingnya dan hubungan antara setiap objek dengan objek yang lainnya.

8.5 Activity Diagram

Activity diagram menggambarkan tentang workflow proses dan urutan aktifitas dalam sebuah proses yang sangat mirip dengan flowchart. Membuat activity diagram sangat menguntungkan karena membantu memahami seluruh proses pada beberapa use case.

BAB IX DIAGRAM STATUS

Objek dalam kelas dapat dimiliki dan menyebabkan state berubah yang disebut juga transisi. Suatu transisi juga dapat memiliki aksi yang dihubungkan pada state.

Diagram status akan menjelaskan kondisi dari berbagai komponen dalam sistem yang merupakan objek dalam satu sistem. Diagram status juga menjelaskan aliran kontrol dari satu status ke status yang lain dimana suatu kondisi dari suatu objek yang ada dan perubahan terjadi sekiranya ada event yang terpicu.

Tujuan utama dalam pembuatan diagram status antara lain :

1. Memodelkan aspek dinamis dari sistem
2. Memodelkan life time terhadap reaksi dari sistem
3. Menjelaskan perbedaan antara status selama objek berjalan didalam sistem

9.1 Cara Mendefinisikan Statua dan Transisi

Sebelum memulai diagram status, harus terlebih dahulu diidentifikasi hal-hal berikut :

1. Mengidentifikasi objek penting untuk dianalisa
2. Mengidentifikasi status
3. Mengidentifikasi event

Sedangkan untuk mendefinisikan status terdapat hal yang harus diidentifikasi terlebih dahulu :

1. Perhatikan sebuah kelas
2. Perhatikan atribut-atribut yang dimiliki
3. Perhatikan hubungan antar kelas, jika ada maka kemungkinan akan menjadi status

Terdapat 3 atribut dalam mendefinisikan transisi :

1. Event sebagai peristiwa yang terjadi
2. Guard sebagai syarat terjadinya transisi yang bersangkutan
3. Action sebagai akibat dari event yang dilakukan

Penggambaran transisi ini minimal memiliki satu keterangan yaitu berupa event atau guard.

BAB X COMPONENT DIAGRAM DAN DEPLOYMENT DIAGRAM

UML dirancang untuk fokus pada dokumen perangkat lunak dari suatu sistem atau aplikasi tetapi diagram komponen dan diagram deployment merupakan diagram yang memiliki tujuan untuk perangkat lunak dan komponen perangkat keras. Diagram komponen dan diagram deployment menunjukkan bagaimana aplikasi menggunakan perangkat keras.

10.1 Component Diagram

Diagram komponen digunakan untuk memvisualisasikan aspek dari suatu sistem yang berupa modul-modul tentang kode librari maupun executable, file atau dokumen yang ada dalam node. Pada umumnya, komponen ini terbentuk dari beberapa class atau package maupun komponen yang lebih kecil.

Diagram ini juga berfungsi untuk menjelaskan cara sistem dapat dieksekusi dalam suatu sistem, tujuan dari diagram komponen secara global yaitu :

1. Memvisualisasikan komponen dari suatu sistem
2. Membangun file yang dapat dieksekusi
3. Menjelaskan organisasi dan hubungan dari komponen

Apabila diagram komponen diabaikan, maka aplikasi tidak dapat dilaksanakan secara efisien dimana sangat penting keberadaannya untuk aspek-aspek lain dalam aplikasi, misalnya kinerja, perawatan dan lain sebagainya.

10.2 Cara Mendefinisikan Diagram Component

Beberapa hal yang harus diperhatikan untuk mendefinisikan komponen yaitu :

1. File apa yang harus digunakan
2. Librari dan dokumen apa saja yang berhubungan dan relevan dengan aplikasi
3. Bagaimana hubungan antara dokumen-dokumen tersebut
Setelah menentukan dan mengidentifikasi dokumen yaitu :
 1. Tentukan nama yang ada pada diagram, yang digunakan dalam suatu blok tertentu misalnya database.
 2. Siapkan rancangan posisi layout pada diagram komponen sebelum menggunakan perangkat lunak permodelan UML untuk menggambarinya.
 3. Jika diperlukan, buatlah catatan-catatan kecil untuk menjelaskan hal khusus yang bersifat optional.

BAB XI PENUTUP

11.1 Kesimpulan

Unified Modeling Language adalah bahasa untuk menspesifikasi, memvisualisasi, membangun dan mendokumentasikan artifacts (bagian dari informasi yang digunakan atau dihasilkan oleh proses pembuatan perangkat lunak, artifact tersebut dapat berupa model, deskripsi atau perangkat lunak) dari sistem perangkat lunak, seperti pada pemodelan bisnis dan sistem non perangkat lunak lainnya.

11.2 Saran

Apabila dalam penyusunan tulisan ini terdapat suatu kekurangan, maka saya sebagai penyusun menerima dengan besar hati apabila ada kritik dan saran guna kesempurnaan dari tulisan-tulisan selanjutnya.

DAFTAR PUSTAKA

Tohari Hamim (2014). “ASTAH” Analisis Serta Perancangan Sistem Informasi Melalui Pendekatan UML, Yogyakarta: Penerbit Andi.
Nugroho Adhi (2009). Rekayasa Perangkat Lunak Menggunakan UML dan JAVA, Yogyakarta: Penerbit Andi.