

BAB II

TINJAUAN PUSTAKA

II.1. Analisis Sistem

Analisis sistem merupakan sekumpulan prosedur untuk membuat spesifikasi sistem informasi yang baru atau sistem informasi yang dimodifikasi. Agar efektif, maka seorang analis sistem harus memiliki pengetahuan dalam bidang komputer dan bisnis. Jika analis sistem hanya memiliki latar belakang komputer, maka di dalam tim pengembangan sistem harus ada orang yang memiliki keahlian dalam bidang bisnis (Sistem Informasi : 2011 dan Jurnal Akutansi Universitas Udayana 6.3 : 2014).

II.2. Tujuan Analisis Sistem

Tujuan analisis sistem adalah mengembangkan persyaratan bagi sistem baru. Analisis sistem memerlukan studi terhadap sistem yang ada dan solusi yang diajukan jauh lebih rinci dari pada tahap *survey* atau investigasi sistem. Jika survei atau investigasi sistem membantu manajemen menentukan masalah dan membantu memilih apakah akan melanjutkan perkembangan sistem atau tidak, analisis sistem dilakukan untuk memperoleh informasi tambahan yang berguna untuk menjelaskan masalah secara keseluruhan dan memilih serta mengevaluasi solusi masalah, sehingga manajemen dapat memutuskan apakah mengembangkan sistem dilanjutkan. Apabila pengembangan sistem akan dilanjutkan, maka solusi yang dipakai sudah diketahui dalam analisis sistem ini pilihan dari alternatif solusi

yang dipakai sudah diketahui dalam analisis sistem ini pilihan dari alternatif solusi sudah dijustifikasi dengan pertimbangan biaya manfaat (*Cost benefit*). Selain itu, persyaratan fisik atas desain yang dipilih serta anggaran untuk tahap perancangan sistem (Jurnal Sistem informasi akutansi landasan bagi sistem informasi lain)

II.3 Sistem Informasi Akutansi

Sistem informasi akutansi (SIA) adalah sebuah sistem informasi yang menangani segala sesuatu yang berkenaan dengan akutansi, Akutansi sendiri sebenarnya adalah sebuah sistem informasi. Sistem Informasi Akutansi (SIA) merupakan suatu kerangka pengkoordinasian sumber daya (*data, materials, equipment, suppliers, personal, and funds*) untuk mengkonversi input berupa data ekonomik menjadi keluaran berupa informasi keuangan yang digunakan untuk pelaksanaan kegiatan suatu entitas dan menyediakan informasi bagi pihak-pihak yang berkepentingan. Sistem Informasi Akutansi dapat memberikan kemudahan bagi para akutan manajemen unutkan menghasilkan informasi keuangan yang akurat, relevan, dan tepat waktu, sehingga akan menjadi lebih *efektif* dan *efisien* dalam membuat keputusan (Sistem Informasi : 2011 dan Jurnal Akutansi Universitas Udayana 6.3 : 2014).

Tujuan dari Sistem informasi Akutansi adalah:

1. Mendukung operasi sehari hari
2. Mendukung pengambilan keputusan manajemen
3. Memenuhi kewajiban yang berhubungan dengan pertanggungjawaban

Komponen-Komponen yang terdapat dalam Sistem Informasi Akutansi adalah sebagai berikut:

1. Orang-orang yang mengoperasikan sistem tersebut.
2. Prosedur-prosedur, baik manual terotomatisasi, yang dilibatkan dalam pengumpulan, pemrosesan dan penyimpanan data aktivitas - aktivitas organisasi.
3. Data tentang proses – proses bisnis
4. Software yang dipakai untuk memproses data organisasi
5. Instruktur teknologi informasi

Aktivitas utama dalam Sistem Informasi Akutansi adalah:

1. *Inbound Logistic* : penerimaan, penyimpanan dan distribusi bahan-bahan masukan
2. *Operasi* : aktivitas untuk mengubah masukan menjadi barang atau jasa
3. *Outbond Logistic* : distribusi produk ke pelanggan
4. Pemasaran dan Penjualan
5. Pelayanan : Dukungan purna jual dan *maintenance*

II.4. Pendapatan

Pendapatan merupakan kenaikan modal perusahaan yang timbul akibat adanya suatu kegiatan perusahaan yang menyangkut hal-hal yang mempengaruhi modal perusahaan maupun penerimaan upah dari suatu jasa yang dilakukan perusahaan kepada konsumen. PSAK NO.23 Pernyataan Standart Akutansi Keuangan 06 (2010 : 23.3) menyatakan bahwa pendapatan adalah arus kas

adalah arus masuk bruto dari manfaat ekonomi yang timbul dari aktivitas normal entitas selama periode jika arus masuk tersebut mengakibatkan kenaikan *equitas* yang tidak berasal dari kontribusi penanaman modal. Suatu perusahaan akan meningkatkan kinerja karyawan untuk menghasilkan pendapatan yang semaksimal mungkin. Perusahaan harus dapat menentukan jumlah pendapatan yang dihasilkan untuk dapat menentukan laba perusahaan dalam pembuatan laporan pendapatan perusahaan.

II.5. Microsoft Visual Studio 2010

Visual Studio adalah inkarnasi dari bahasa Visual Basic yang sangat populer dan telah dilengkapi dengan fitur serta fungsi yang setara dengan bahasa tingkat tinggi lainnya seperti C++. Visual Studio juga dapat digunakan untuk dapat membuat aplikasi *Windows, mobile, web, dan Office* yang kompleks dengan menggunakan kode yang telah ditulis dan Visual Studio juga menyediakan berbagai *Tools* dan fitur canggih yang memungkinkan untuk menulis kode, menguji, dan menjalankan program tunggal atau terkadang serangkaian program yang terkait dengan satu aplikasi (Lee Christopher ; 2014 : 1-2)

II.6. Microsoft SQL Server

Sql Server adalah software database RDBMS yang berjenis *client server*. yang memudahkan dalam menyimpan dan mengambil data menggunakan paradigma RDBMS. *Query* merupakan sebuah *statement* yang dibuat menggunakan bahasa Sql (*Structured Query Language*). *Sql* menggunakan bahasa terstandarisasi yang awalnya dikembangkan oleh IMB untuk melakukan

mengambil data, mengubah data, dan menentukan database relasional menggunakan bahasa deklaratif yang mudah dipahami. (Wahana Komputer ; 2012 : 26)

II.2.1. Kamus Data

Kamus Data atau *Data Dictionary* atau disebut juga dengan *System Data Dictionary* adalah katalog fakta tentang data dan kebutuhan - kebutuhan informasi dari suatu sistem informasi. Dengan menggunakan kamus data analisis sistem dapat mendefinisikan data yang mengalir di sistem dengan lengkap. Pada tahap analisis, kamus digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem tentang data yang mengalir di sistem, yaitu tentang data yang masuk ke sistem dan tentang informasi yang dibutuhkan oleh pemakai sistem.

Adapun simbol-simbol kamus data yaitu :

Tabel II.1. Simbol – simbol kamus data

No.	Simbol	Keterangan
1.	=	Terdiri dari, mendefinisikan, diuraikan menjadi
2.	+	Dan
3.	()	Menunjukkan suatu elemen yang bersifat pilihan (opsional). Elemen-elemen yang bersifat pilihan ini bisa dikosongkan pada layar masukan atau bisa juga dengan memuat spasi atau nol untuk field-field numeric pada struktur file
4.	{ }	Menunjukkan elemen-elemen repetitive, juga disebut kelompok berulang atau tabel-tabel. Kemungkinan bisa ada

		satu atau beberapa elemen berulang di dalam kelompok tersebut. Kelompok berulang bisa mengandung keadaankeadaan tertentu, seperti misalnya, jumlah pengulangan yang pasti atau batas tertinggi dan batas terendah untuk jumlah pengulangan.
5.	[]	Menunjukkan salah satu dari dua situasi tertentu. Satu elemen bisa ada sedangkan elemen lainnya juga ada, tetapi tidak bisa kedua-duanya ada seara bersamaan. Elemen-elemen yang ada di dalam tanda kurung ini saling terpisah satu sama lain. (dengan kata lain, memilih salah satu dari sejumlah alternatif, seleksi)
6.		Pemisah sejumlah alternatif pilihan antara symbol []
7.	@	Identifikasi atribut kunci
8.	**	Komentar

(Sumber : *Jurnal Analisis Sistem Informasi ; 2010 : 4*)

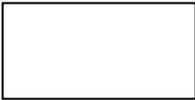
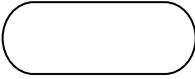
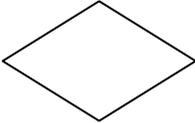
II.2.2. Entity Relationship Diagram

Pada dasarnya ERD (*Entity Relationship Diagram*) adalah Gambar atau diagram yang menunjukkan informasi dibuat, disimpan, dan digunakan dalam sistem bisnis. ERD ini digunakan untuk melakukan permodelan terhadap struktur data dan penggunaannya ERD ini dilakukan untuk mengurangi tingkat kerumitan penyusunan sebuah database yang baik. (Jurnal Iptek Ibnu Aqil ; 2010 : 5)

Entity dapat berarti sebuah obyek yang dapat dibedakan dengan obyek lainnya. Obyek tersebut dapat memiliki komponen-komponen data (atribut atau

field) yang membuatnya dapat dibedakan dari obyek yang lain. Dalam dunia database *entity* memiliki atribut yang menjelaskan karakteristik dari entity tersebut. Ada dua macam atribut yang di kenal deskriptif. Hal ini berarti setiap entity memiliki himpunan yang diperlukan sebuah primary key untuk membedakan anggota-anggota dalam himpunan tersebut.

Tabel II.2. Simbol pada ERD

No.	Simbol	Keterangan
1.		Persegi panjang menyatakan himpunan entitas adalah orang, kejadian ,atau berada di mana data akan dikumpulkan
2.		Atribut merupakan informasi yang diambil tentang sebuah entitas
3.		Belah ketupat merupakan himpunan relasi merupakan hubungan antar entitas
4.		Garis sebagai penghubung antara himpunan, relasi, dan himpunana entitas dengan atributnya.

(Sumber : Jurnal Iptek Ibnu Aqil ; 2010 : 6)

Ada beberapa derajat relasi yang dapat terjadi, yaitu :

1. *One to one*, menggambarkan bahwa antara 1 anggota entity A hanya dapat berhubungan dengna 1 anggota entity B. Biasanya derajat relasi ini digambarkan dengna simbol 1-1.

2. *One to many*, menggambarkan bahwa 1 anggota entity A dapat memiliki hubungan dengan lebih dari 1 anggota entity B. Biasanya derajat relasi ini digambarkan dengan simbol 1-N.
3. *Many to many*, menggambarkan bahwa lebih dari satu anggota A dapat memiliki hubungan dengan lebih dari satu anggota entity B. Simbol yang digunakan adalah N-N.

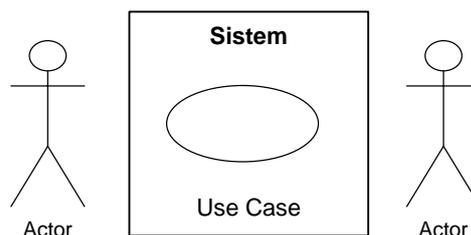
II.9. Karakter Diagram UML

Diagram berbentuk grafik yang menunjukkan simbol elemen model yang disusun untuk mengilustrasikan bagian atau aspek tertentu dari sistem. Sebuah diagram merupakan bagian dari suatu view tertentu dan ketika digambarkan biasanya dialokasikan untuk view tertentu.

II.9.1. Use Case Diagram

Use case adalah abstraksi dari interaksi antara sistem dan actor. Use case bekerja dengan cara mendeskripsikan tipe interaksi antara user sebuah system dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Use case merupakan konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata user. Sedangkan use case diagram memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan client.

Diagram use case menunjukkan 3 aspek yaitu: aktor, use case dan sistem / sub sistem boundary. Actor mewakili peran orang, sytem yang lain atau laot ketika berkomunikasi dengan use case. Gambar berikut mengilustrasikan aktor, use case dan boundary.



Gambar II.1. Use Case Model
 (Sumber : Jurnal Memahami Penggunaan UML ; 2011 : 6)

Tabel II.3. Komponen Use Case Diagram

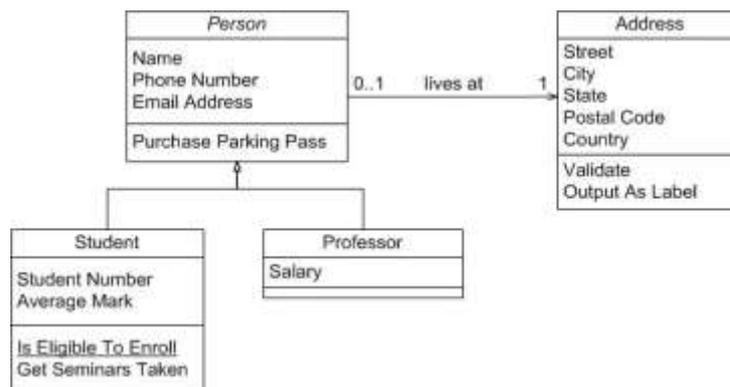
No.	Nama Komponen	Keterangan	Simbol
1	Actor	Actor adalah pengguna sistem. Actor tidak terbatas hanya manusia saja, jika sebuah sistem berkomunikasi dengan aplikasi lain dan membutuhkan input atau mmberikan output, maka aplikasi tersebut juga bisa dianggap sebagai actor.	
2	Association	Asosiasi digunakan untuk menghubungkan actor dengan use case. Asosiasi digambarkan dengan dengan sebuah garis yang menghubungkan antara Actor dengan Use Case.	
3	Use Case	Use Case di gambarkan sebagai lingkaran elips dengan nama Use Case dituliskan didalam elips tersebut.	

(Sumber : Jurnal Grady Booch, Visual Modeling ; 2013 : 3)

II.9.2. Class Diagram.

Class adalah dekripsi kelompok obyek-obyek dengan property, perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya class diagram dapat memberikan pandangan global atas sebuah system. Hal tersebut tercermin dari

class- class yang ada dan relasinya satu dengan yang lainnya. Sebuah sistem biasanya mempunyai beberapa class diagram. Class diagram sangat membantu dalam visualisasi struktur kelas dari suatu sistem.



Gambar II.2. Class diagram

(Sumber : Wahana Komputer ; 2012 : 105)

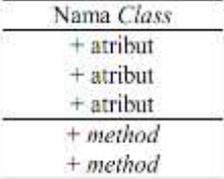
Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi).

Class diagram menggambarkan struktur dan deskripsi class, package dan objek beserta hubungan satu sama lain seperti containment, pewarisan, asosiasi, dan lain-lain. Class memiliki tiga area pokok, yaitu :

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

Berikut ini merupakan komponen-komponen pada class diagram.

Tabel II.4. Komponen Class Diagram

No.	Nama Komponen	Keterangan	Simbol
1	Class	<p>Class Class adalah blok - blok pembangun pada pemrograman berorientasi obyek. Sebuah class digambarkan sebagai sebuah kotak yang terbagi atas 3 bagian. Bagian atas adalah bagian nama dari class. Bagian tengah mendefinisikan property/atribut class. Bagian akhir mendefinisikan methodmethod dari sebuah class.</p>	
2	Association	<p>Association Sebuah asosiasi merupakan sebuah relationship paling umum antara 2 class dan dilambangkan oleh sebuah garis yang menghubungkan antara 2 class. Garis ini bisa melambangkan tipe-tipe relationship dan juga dapat menampilkan hukum-hukum multiplisitas pada sebuah relationship. (Contoh: One-to-one, one-to-many, many-to-many).</p>	
3	composition	<p>Composition Jika sebuah class tidak bisa berdiri sendiri dan harus merupakan bagian dari class yang lain, maka class tersebut memiliki relasi Composition terhadap class tempat dia bergantung tersebut. Sebuah relationship composition</p>	

		digambarkan sebagai garis dengan ujung berbentuk jajaran genjang berisi/solid.	
4	Dependency	Dependency Kadangkala sebuah class menggunakan class yang lain. Hal ini disebut dependency. Umumnya penggunaan dependency digunakan untuk menunjukkan operasi pada suatu class yang menggunakan class yang lain. Sebuah dependency dilambangkan Sebagai panah bertitik-titik.	
5.	Aggregation	Aggregation mengindikasikan keseluruhan bagian relationship dan biasa disebut relasi	

(Sumber : Jurnal Grady Booch. Visual Modeling ; 2013 : 1)

II.9.3. Activity Diagram.

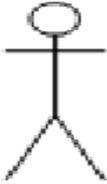
Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti use case atau interaksi.

II.9.4. Sequence Diagram.

Sequence Diagram digunakan untuk menggambarkan perilaku pada sebuah scenario. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara object juga interaksi antaraobject, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

Berikut ini merupakan komponen-komponen pada sequence diagram.

Tabel II.5. Komponen Sequence Diagram

No.	Nama Komponen	Keterangan	Simbol
1	Object	Object merupakan instance dari sebuah class dan dituliskan tersusun secara horizontal. Digambarkan sebagai sebuah class (kotak) dengan nama object didalamnya yang diawali dengan sebuah titik koma.	
2	Actor	Actor juga dapat berkomunikasi dengan object, maka actor juga dapat diurutkan sebagai kolom. Simbol Actor sama dengan simbol pada Actor Use Case Diagram.	

(Sumber : Jurnal Grady Booch, Visual Modeling ; 2013 : 3)