

BAB II

TINJAUAN PUSTAKA

II.1. Pengertian Sistem

Pada bagian ini akan dibahas definisi sistem informasi akuntansi dan istilah yang berkaitan dengan hal tersebut untuk membantu anda memahami seluruh materi dalam buku ini. Sistem merupakan serangkaian bagian saling berhubungan dan bekerja sama untuk mencapai tujuan tertentu. Suatu sistem pasti tersusun dari sub-sub sistem yang lebih kecil yang juga saling tergantung dan bekerja sama untuk mencapai tujuan.

Tujuan dasar suatu sistem tergantung pada jenis sistem itu sendiri. Sebagai contoh, sistem peredaran darah manusia merupakan sistem biologi yang memiliki tujuan untuk mengedarkan darah yang mengandung oksigen dan sari makanan ke seluruh tubuh. Sedangkan sistem buatan manusia seperti sistem yang terdapat di sekolah, organisasi bisnis, atau instansi pemerintah juga mempunyai tujuan yang berbeda-beda. Organisasi bisnis biasanya memiliki tujuan yang lebih jelas (Anastasia Diana dan Lilis Setiawati : 2011 ; 2).

II.2. Pengertian Informasi

Sistem informasi manajemen berhubungan dengan informasi. Berapa banyak informasi yang diberikan oleh sebuah sistem informasi ? belum ada metode untuk mengukur informasi dalam sebuah sistem untuk menghitung isinya. Informasi adalah sebuah istilah yang tepat dalam pemakaian umum. Informasi

dapat mengenai data mentah, data tersusun, kapasitas sebuah saluran komunikasi , dan lain sebagainya. Informasi ibarat darah yang mengalir dalam tubuh suatu organisasi sehingga informasi ini sangat penting di dalam suatu organisasi. Suatu sistem yang kurang mendapatkan informasi akan menjadi luruh, kerdil, dan akhirnya mati (Tata Sutabri ; 2013 : 29).

II.3. Sistem Informasi Akuntansi

Informasi merupakan suatu data yang diorganisasi yang dapat mendukung ketepatan pengambilan keputusan. Sistem merupakan sekumpulan sumber daya yang dapat terkait untuk mencapai suatu tujuan. SIA (Sistem Informasi Akuntansi) merupakan kumpulan sumber daya, seperti manusia dan peralatan, yang dirancang untuk mengubah data keuangan dan data lainnya ke dalam informasi. Informasi tersebut dikomunikasikan kepada para pembuat keputusan. Sistem informasi akuntansi melakukan hal tersebut dengan sistem manual atau melalui sistem terkomputerisasi. (George H. Bodnar;2006:3)

II.4. Metode FIFO

Dalam bab-bab yang lalu telah dijelaskan bahwa harga pokok penjualan untuk suatu periode tertentu dihitung berdasarkan rumus: persediaan awal ditambah dengan pembelian bersih dikurangi dengan persediaan akhir. Angka untuk pembelian bersih diambil dari saldo akun yang bersangkutan di buku besar. Angka-angka kuantitas persediaan awal dan akhir diperoleh dengan jalan melakukan perhitungan fisik. Harga pokok persediaan dengan mengalikan

kuantitas pada harga pokok per unit. Harga pokok per unit mana yang dipakai tergantung pada metode penetapan harga pokok yang dipilih (Soermarso S.R : 2009 : 387).

Tabel II.1. Metode FIFO

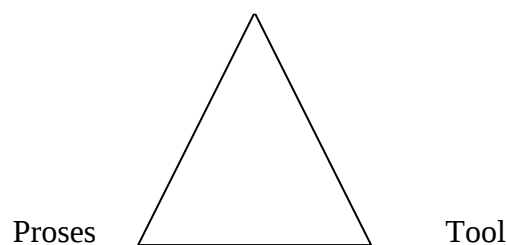
Kode	Nama	Masuk			Keluar			Sisa		
		Unit	Harga	Jumlah	Unit	Harga	Jumlah	Unit	Harga	Jumlah
01	Besi	50	100	5000	10	100	1000	40	100	4000
01	Besi	40	100	4000	20	100	2000	20	100	2000
01	Besi	20	100	2000	20	100	2000	0	100	0
02	Plat	100	200	20000	50	200	10000	50	200	10000
02	Plat	50	200	10000	20	200	4000	30	200	6000
02	Plat	30	200	6000	30	200	6000	0	200	0

(Sumber : Bantu Tampubolon, dkk : 2009 : 250)

II.5. UML (Unified Modelling Language)

Menurut Yuni Sugiarti (2013 : 34) *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.

Metodologi Pemodelan



Gambar II.1. The Triangle For Success

(Sumber : Yuni Sugiarti : 2013 ; 34)

Menurut Prabowo Pudjo Widodo dan Herlawati (2011 : 6-7), *UML* singkatan dari *Unified Modelling Language* yang berarti bahasa permodelan standar. *UML* diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :

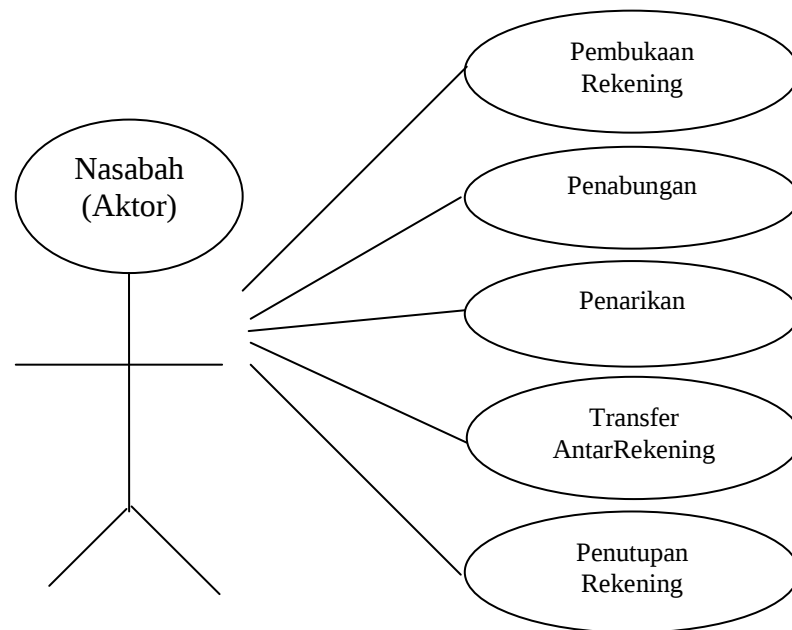
1. Merancang perangkat lunak
2. Sarana Komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasi sistem yang ada, proses-proses dan organisasinya.

II.5.1. Use Case Diagram

Segala sesuatu yang secara akademis dikembangkan pada umumnya berawal dari suatu konsep. Demikian juga halnya dengan pengembangan sistem pada umumnya dikembangkan berdasarkan analisis kebutuhan. Analisis kebutuhan ini adalah tahap konseptualisasi, yaitu suatu tahap yang mengharuskan analis dan perancang sistem untuk berusaha tahu secara pasti mengenai hal yang menjadi kebutuhan dan harapan pengguna sehingga kelak aplikasi yang dibuat memang akan digunakan oleh pengguna (*user*) serta akan memuaskan kebutuhan dan harapannya.

Dalam konteks UML, tahap konseptualisasi dilakukan dengan pembuatan *use case diagram* yang sesungguhnya merupakan deskripsi peringkat tinggi bagaimana perangkat lunak (aplikasi) akan digunakan oleh penggunanya. Selanjutnya, *use case diagram* tidak hanya sangat penting pada saat analisis, tetapi juga sangat penting dalam tahap perancangan (*design*), untuk mencari kelas-kelas yang terlibat dalam aplikasi, dan untuk melakukan pengujian (*testing*). Saat akan mengembangkan *use case diagram*, hal yang pertama kali harus dilakukan adalah mengenali *actor* untuk sistem yang sedang dikembangkan. Dalam hal ini, ada beberapa karakteristik untuk para *actor*, yaitu *actor* yang ada di

luar sistem yang sedang dikembangkan dan *actor* yang berinteraksi dengan sistem yang sedang dikembangkan. (Adi Nugroho ; 2009 : 7).

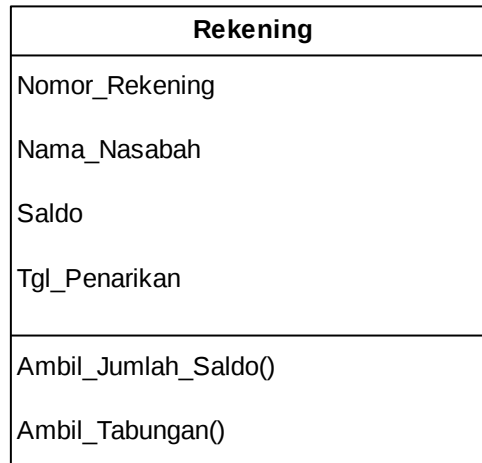


Gambar II.2. Contoh Use Case Diagram
(Sumber : Adi Nugroho ; 2009 : 8)

II.5.2. Class Diagram

Menurut Rosa A.S dan M. Shalahuddin (2011 : 122), *Class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

- a. Atribut merupakan *varabel-variabel* yang dimiliki oleh suatu kelas.
- b. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.



Gambar II.3. Contoh Class Diagram
(Sumber : Adi Nugroho ; 2009: 61)

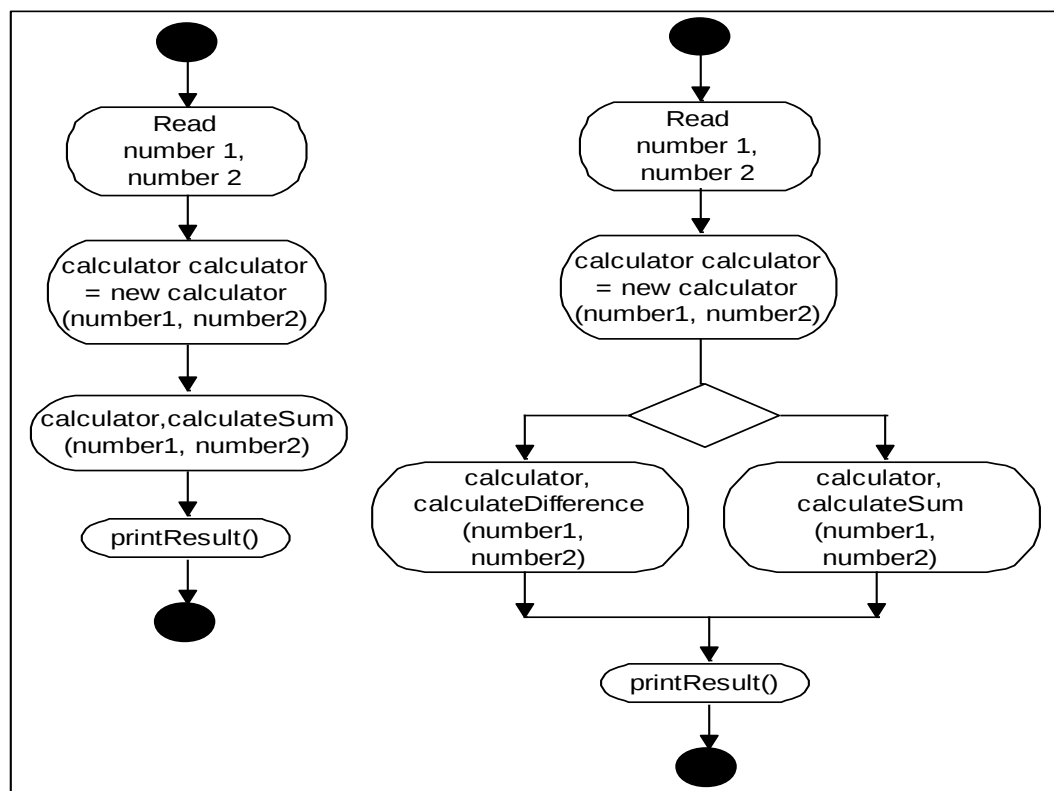
II.5.3. Activity Diagram

Diagram aktivitas atau *activity diagram* menggambarkan aliran fungsionalitas sistem. Pada tahap pemodelan bisnis, diagram aktivitas dapat digunakan untuk menunjukkan aliran kerja bisnis (*business work flow*). Dapat juga digunakan untuk menggambarkan aliran kejadian (*flow of event*) dalam *use case*.

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. *Activity diagram* merupakan *state diagram* khusus, di mana sebagian besar *state* adalah *action* dan sebagian besar transisi di- *trigger* oleh selesainya *state* sebelumnya (*internal processing*).

Oleh karena itu *activity diagram* tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih

menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. Sama seperti *state*, standar UML menggunakan segiempat dengan sudut membulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan behaviour pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (*fork* dan *join*) digunakan titik sinkronisasi yang dapat berupa titik, garis *horizontal* atau vertikal. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu (Adi Nugroho ; 2009 : 13).



Gambar II.4. Contoh Activity Diagram
(Sumber : Adi Nugroho ; 2009 : 74)

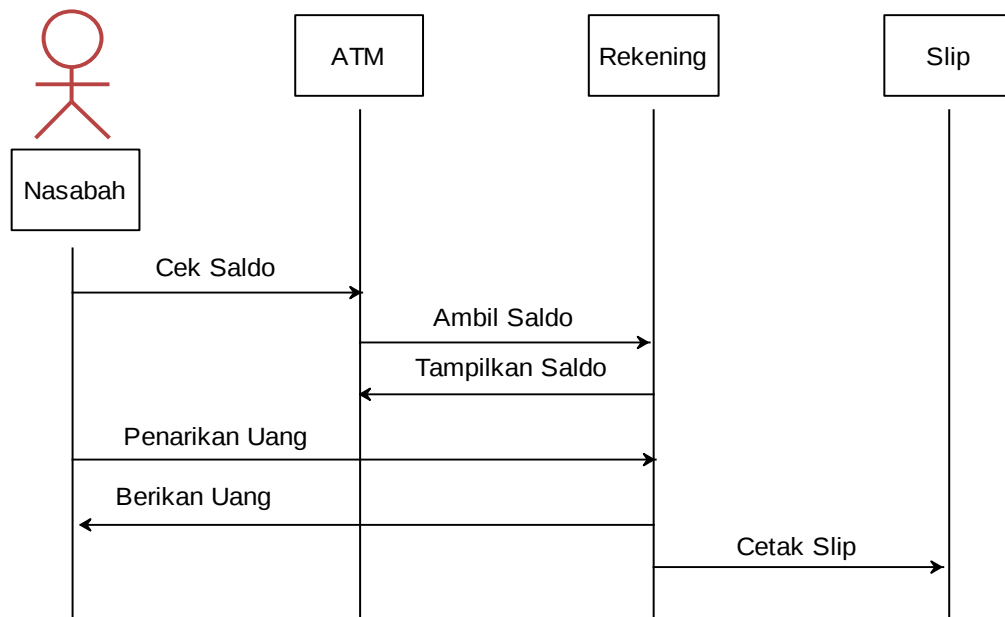
II.5.4. Sequence Diagram

Diagram sekuensial atau *sequence diagram* digunakan untuk menunjukkan aliran fungsionalitas dalam *use case*. Diagram sekuensial adalah diagram yang disusun berdasarkan urutan waktu. Kita membaca diagram sekuensial dari atas ke bawah. Setiap diagram sekuensial mempresentasikan suatu aliran dari beberapa aliran di dalam *use case*.

Jadi dengan kata lain sekuensial diagram menunjukkan aliran fungsionalitas berdasarkan urutan waktu serta kejadian yang nantinya akan menentukan metode/fungsi atribut masing-masing. Dimana fungsi-fungsi tersebut akan diterapkan pada suatu kelas/objek.

Perhatikan gambar II.5. dimana terlihat pengelompokkan *event-event* serta fungsi masing-masing atribut tersebut. Di dalam diagram terlihat jelas bagaimana aliran suatu proses kejadian dimana seorang nasabah yang akan melakukan transaksi dengan sebuah mesin ATM. Dari diagram tersebut kita mengetahui *event-event* yang terjadi, seperti : Nasabah memasukkan kartu ATM, Mesin ATM merespon dengan meminta *password* atau PIN, dan selanjutnya.

Kita dapat melihat setiap fungsi atribut dan *event-event* apa saja yang terjadi. Sehingga melalui diagram sekuensial ini kita dapat merancang suatu program aplikasi yang baik, sehingga dalam menghadapi sebuah kasus yang benar-benar kompleks diagram sekuensial ini sangat membantu.



Gambar II.5. Contoh Sequence Diagram
(Sumber : Adi Nugroho ; 2009 : 102)

II.6. Database

Menurut Budi Raharjo (2011 : 245) dalam bekerja dengan *database*, anda perlu memiliki pengetahuan yang cukup tentang konsep *client/server* dan tipe-tipe arsitektur aplikasi *database* yang ada. Dengan demikian, anda akan mengetahui dengan pasti tujuan dan fungsi (untuk apa) anda membuat dan mengolah *database*.

II.7. Basis Data

Menurut Janner Simarmata (2007 : 1), Kata “*basis data*” bisa digunakan untuk menguraikan segala sesuatu dari sekumpulan data tunggal, seperti daftar telepon. *Basis data* terdiri dari *file-file* fisik yang ditetapkan berdasarkan komputer saat menerapkan perangkat lunak basis data. Sedangkan menurut Stephens dan Plew (Dalam Janner Simarmata dan Iman Paryudi; 2006 : 1), Basis

Data adalah mekanisme yang digunakan untuk menyimpan informasi atau data. Informasi adalah sesuatu yang kita gunakan sehari-hari untuk berbagai alasan. Dengan basis data, pengguna dapat menyimpan data secara terorganisasi. Setelah data disimpan, informasi harus mudah diambil. Kriteria dapat digunakan untuk mengambil informasi. Cara data disimpan dalam basis data menentukan seberapa mudah mencari informasi berdasarkan banyak kriteria. Data pun harus mudah ditambahkan ke dalam basis data, dimodifikasi, dan dihapus.

II.8. Kamus Data Dan Normalisasi

Menurut Budi Raharjo (2011 : 59) dalam suatu rancangan *database*, *data dictionary* digunakan untuk menjelaskan atau mendeskripsikan kolom-kolom pada masing-masing tabel yang akan dibuat dalam *database*. Deskripsi kolom yang dimaksud di sini meliputi tipe data, lebar karakter atau digit, serta keterangan tentang kunci relasi.

Tabel II.2. Tabel kategori

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Kategori_id	INT	11	NOT NULL	Primary Key
Kategori_nama	VARCHAR	25		

Tabel II.3. Tabel pengarang

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Pengarang_id	CHAR	3	NOT NULL	Primary Key
Pengarang_nama	VARCHAR	30		

Tabel II.4. Tabel penerbit

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Penerbit_id	CHAR	4	NOT NULL	Primary Key
Penerbit_nama	VARCHAR	50		

Tabel II.5. Tabel buku

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Buku_isbn	CHAR	13	NOT NULL	Primary Key
Buku_judul	VARCHAR	75		
Penerbit_id	CHAR	4		
Buku_tglterbit	DATE	-		
Buku_jmlhalaman	INT	11		
Buku_deskripsi	TEXT	-		
Buku_harga	DECIMAL	10,0		

Tabel II.6. Tabel link_buku_pengarang

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Buku_isbn	CHAR	13	NOT NULL	Primary Key dan Forign Key
Pengarang_id	CHAR	3	NOT NULL	Primary Key dan Forign Key

Tabel II.6. Tabel link_buku_kategori

Nama Kolom	Tipe Data	Lebar	NULL?	Kunci
Buku_isbn	CHAR	13	NOT NULL	Primary Key dan Foreign Key
kategori_id	CHAR	11	NOT NULL	Primary Key dan Foreign Key

(Sumber : Budi Raharjo : 2011 : 59)

II.9. Entity Relationship Diagram (ERD)

Menurut Budi Raharjo (2011 : 57) *entity relationship diagram* merupakan salah satu alat bantu (berupa gambar) dalam *database relasional* yang berguna untuk menjelaskan hubungan atau relasi antartabel yang terdapat di dalam *database*. Dalam ERD kita juga dapat melihat daftar kolom yang menyusun masing-masing tabel. ERD inilah yang akan kita gunakan sebagai acuan untuk membahas materi dalam buku ini. Berikut keterangan tanda yang digunakan :

1. 1- ∞ menunjukkan relasi 1-ke-banyak
2. n- ∞ menunjukkan relasi banyak-ke-banyak