

BAB 2

LANDASAN TEORI

2.1. Sistem Informasi

2.1.1 Pengertian Sistem

Menurut Romney dan Steinbart (2015:3), sistem adalah suatu rangkaian yang terdiri dari dua atau lebih komponen yang saling berhubungan dan saling berinteraksi satu sama lain untuk mencapai tujuan dimana sistem biasanya terbagi dalam sub system yang lebih kecil yang mendukung system yang lebih besar.

Menurut Gelinas dan Dull (2012:11), Sistem merupakan seperangkat elemen yang saling bergantung yang bersama-sama mencapai tujuan tertentu. Dimana sistem harus memiliki organisasi, hubungan timbal balik, integrasi dan tujuan pokok.

Dengan demikian dapat disimpulkan bahwa sistem merupakan seperangkat elemen yang saling berhubungan yang bersama-sama mencapai suatu tujuan tertentu dalam proses yang teratur yang dapat mendukung sistem yang lebih besar dan saling memiliki ketergantungan untuk mencapai tujuan tertentu.

2.1.2 Pengertian Informasi

Menurut Gellinas and Dull (2012:12) informasi merupakan data yang disajikan dalam suatu bentuk yang berguna terhadap aktifitas pengambilan keputusan.

Menurut Rommey dan Steinbart (2015:4), informasi adalah data yang telah dikelola dan di proses untuk memberikan arti dan memperbaiki proses pengambilan keputusan.

Menurut Gelinas dan Dull (2012:19), Ada beberapa karakteristik informasi yang berkualitas, yaitu:

- A. *Effectiveness*: berkaitan dengan informasi yang relevan dan berkaitan dengan proses bisnis yang di sampaikan dengan tepat waktu, benar, konsisten dan dapat digunakan.
- B. *Efficiency*: informasi yang berkaitan melalui penyediaan informasi secara optimal terhadap penggunaan sumber daya.
- C. *Confidentiality*: karakteristik informasi yang berkaitan dengan keakuratan dan kelengkapan informasi serta validitas nya sesuai dengan nilai-nilai bisnis dan harapan.
- D. *Integrity*: karakteristik informasi yang berkaitan dengan perlindungan terhadap informasi yang sensitif dari pengungkapan yang tidak sah.
- E. *Availability*: suatu karakteristik informasi yang berkaitan dengan informasi yang tersedia pada saat diperlukan oleh proses bisnis baik sekarang, maupun di masa mendatang, hal ini juga menyangkut perlindungan sumber daya yang diperlukan dan kemampuan yang terkait.
- F. *Compliance*: yaitu karakteristik informasi yang berkaitan dengan mematuhi peraturan dan perjanjian kontrak dimana proses bisnis merupakan subjek nya berupa kriteria bisnis secara internal maupun eksternal.
- G. *Reliability*: karakteristik informasi yang berkaitan dengan penyediaan informasi yang tepat bagi manajemen untuk mengoperasikan entitas dan menjalankan tanggung jawab serta tata kelola pemerintahan.

Dengan demikian dapat disimpulkan bahwa informasi adalah data yang diproses menjadi suatu bentuk yang lebih berguna dan berarti bagi yang menerimanya dalam aktivitas pembuatan keputusan.

2.1.3 Pengertian Sistem Informasi

Menurut Satzinger, Jackson, dan Burd (2012:4), Sistem informasi merupakan kumpulan dari komponen-komponen yang mengumpulkan, memproses, menyimpan, dan menyediakan output dari setiap informasi yang dibutuhkan dalam proses bisnis serta aplikasi yang digunakan melalui perangkat lunak, database dan bahkan proses manual yang terkait.

Menurut Stair and Reynolds (2012:415), Sistem Informasi adalah suatu sekumpulan elemen atau komponen berupa orang, prosedur, database dan alat yang saling terkait untuk memproses, menyimpan serta menghasilkan informasi untuk mencapai suatu tujuan (*goal*).

Menurut Gelinas dan Dull (2012:12) Sistem Informasi adalah sistem yang di buat secara umum berdasarkan seperangkat komputer dan komponen manual yang dapat dikumpulkan, disimpan dan diolah untuk menyediakan output kepada user.

Dengan demikian dapat disimpulkan bahwa sistem informasi adalah suatu kombinasi modul yang terorganisir yang berasal dari komponen-komponen yang terkait dengan hardware, software, people dan network berdasarkan seperangkat komputer dan menghasilkan informasi untuk mencapai tujuan.

2.2 Sistem Informasi Akuntansi

2.2.1 Pengertian Sistem Informasi Akuntansi

Menurut Gelinas dan Dull (2012:13), Sistem informasi akuntansi merupakan sistem yang digunakan untuk mengumpulkan, mengolah, dan melaporkan informasi terkait dengan aspek keuangan akan kegiatan bisnis.

Menurut Romney dan Steinbart (2015:10), Sistem informasi akuntansi adalah sistem yang digunakan untuk mengumpulkan, mencatat,

menyimpan dan mengolah data untuk menghasilkan suatu informasi untuk pengambilan keputusan. Sistem ini meliputi orang, prosedur dan instruksi data perangkat lunak, infrastruktur teknologi informasi serta pengendalian internal dan ukuran keamanan.

Menurut Gelinas dan Dull (2012:13), Sistem Informasi Akuntansi adalah sistem yang digunakan untuk mengumpulkan, memproses, dan melaporkan informasi yang berhubungan dengan aspek keuangan dari peristiwa bisnis.

Menurut A. Hall (2008:8), Sistem Informasi Akuntansi adalah suatu proses transaksi keuangan dan transaksi non keuangan yang mempengaruhi pemrosesan transaksi keuangan dan terdapat pemrosesan data di dalam nya. Terdapat 3 subsystem SIA, yaitu:

1. *TPS (Transaction Processing System)* berupa dukungan operasi bisnis dengan sistem laporan, dokumen dan pesan untuk user dalam suatu organisasi.
2. *General Ledger/Financial Reporting System (GL/FRS)* yaitu berupa prosedur tradisional laporan keuangan seperti income statement, balance sheet, laporan arus kas, pajak, dan laporan lain yang tersedia.
3. *Management Reporting System (MRS)* yaitu berupa penyediaan informasi kepada pihak manager internal dengan tujuan laporan keuangan dan informasi yang dibutuhkan dalam pengambilan keputusan seperti biaya, laporan yang bervariasi, dan laporan pertanggungjawaban.

Dengan demikian dapat disimpulkan bahwa Sistem Informasi Akuntansi merupakan proses transaksi keuangan dan transaksi non keuangan yang mempengaruhi pemrosesan transaksi keuangan dan terdapat pemrosesan data di dalam nya. Yaitu berupa sistem yang digunakan untuk mengumpulkan, mengolah, dan melaporkan informasi terkait dengan aspek keuangan akan kegiatan bisnis.

2.2.2 Siklus Sistem Informasi Akuntansi

Menurut Romney dan Steinbart (2015:7-9), siklus SIA terdiri dari 5 komponen, yaitu:

- A. Siklus Pendapatan (*revenue cycle*), dimana barang dan jasa di jual untuk mendapatkan uang tunai atau janji untuk menerima uang tunai di masa depan.
- B. Siklus Pengeluaran (*expenditure cycle*), dimana perusahaan membeli persediaan untuk dijual kembali atau bahan baku untuk digunakan dalam memproduksi barang sebagai pertukaran uang tunai atau janji untuk membayar uang tunai di masa depan.
- C. Siklus produksi atau konversi (*production or conversion cycle*), dimana bahan baku di transformasikan menjadi barang jadi.
- D. Siklus sumber daya manusia/penggajian (*human resources/payroll cycle*), dimana karyawan dipekerjakan, dilatih, diberi kompensasi, dievaluasi, dipromosikan dan diberhentikan.
- E. Siklus pembiayaan (*financing cycle*), dimana perusahaan menjual saham nya kepada investor dan meminjam uang, kemudian investor akan di bayar dengan dividen dan bunga yang di bayar atas pinjaman nya tersebut.

2.3 Pembelian

2.3.1 Pengertian Pembelian

Menurut Arjan J. Van Weele (2010) bahwa setiap perusahaan membutuhkan fungsi proses sistem pembelian yang melibatkan penentuan pembelian barang, pemilihan pemasok yang baik, evaluasi harga yang tepat, menetapkan syarat dan kondisi yang berlaku serta menindaklanjuti pihak yang berkepentingan untuk memastikan pengiriman barang tepat waktu dan sistem pembayaran barang serta pemantauan dan pengendalian untuk mengamankan pasokan barang.

Menurut Romney dan Steinbart (2015:463), terdapat 4 aktivitas bisnis yang utama dalam siklus pengeluaran yaitu:

1. Pemesanan barang, bahan baku, perlengkapan dan jasa – mengirimkan pesanan ke pemasok.
2. Penerimaan – menerima barang atau jasa dari pemasok.
3. Memproses faktur – meninjau dan menyetujui faktur dari pemasok.
4. Pengeluaran Kas – memproses pembayaran ke pemasok.

Menurut Avery, S. (2010), Juntti, M. (2014) menyatakan bahwa pembelian berhubungan dengan pengendalian rantai pasokan dari pemasok kepada pembeli, negosiasi harga kepada pemasok dan keterlibatan pemasok dengan pengelolaan persediaan yang bertujuan untuk pengadaan barang dan aktivitas pembayaran (*procure to pay*) dimana dalam persediaan barang dagang (*Merchandise Inventory*) harus memiliki pengelolaan pembelian secara efektif yang digunakan untuk memaksimalkan pembelian dan menjaga *stock* untuk mencapai keuntungan.

Menurut Romney dan Steinbart (2015:465) Terdapat 4 aktifitas siklus pembelian, yaitu:

1. Memesan bahan baku, perlengkapan dan jasa.
2. Menerima bahan baku, perlengkapan dan jasa.
3. Menyetujui faktur dari pemasok.
4. Pengeluaran kas.

Menurut James. A. Hall (2008:235) prosedur pembelian adalah suatu proses bisnis yang terlibat dalam mengidentifikasi keperluan persediaan dan pengadaan barang, penempatan pemesanan, penerimaan barang dan pengakuan hutang dimana perusahaan membeli barang jadi untuk di jual kembali. Prosedur pembelian meliputi:

1. *Monitor inventory records*, berupa persiapan persediaan yang diperlukan untuk menjual barang jadi dimana pengendalian persediaan

memonitor dan mencatat level persediaan barang jadi dan menyiapkan *purchase requisition* dan mengirimkan *form purchase requisition* kepada fungsi pembelian.

2. *Prepare Purchase Order*. Berupa penerbitan form setelah di terima nya PR yang di urutkan berdasarkan *vendor* dan PO digunakan dalam proses pemesanan barang untuk dikirim ke vendor sampai barang tersebut diterima.
3. *Receive Goods*. Berupa penerimaan barang dari vendor setelah terjadi nya penerbitan PO dan bagian penerimaan mempersiapkan laporan penerimaan barang berisi kuantitas dan kondisi persediaan tersebut yang kemudian melakukan update atas penerimaan barang.
4. *Update Inventory records*. Berupa metode valuasi persediaan ketika bagian penerimaan melakukan penambahan barang menggunakan *standard cost* yang tersedia ketika barang diterima.
5. *Set Up Accounts Payable*. Dilakukan ketika barang sampai dan invoice diterima lalu bagian AP menerbitkan laporan penerimaan dimana ketika penerimaan barang dari vendor sudah di catat ke dalam pencatatan hutang.
6. *Post to general ledger*. Yaitu fungsi G/L ketika menerima voucher dari department A/P dan berupa ringkasan akun dari pengendalian persediaan.

Menurut Mann, L. (2013) menyatakan bahwa proses pembelian harus terdokumentasi dengan baik. Dokumen *purchase order* harus sesuai dengan kebutuhan dan ukuran pembelian. Tugas bagian pembelian harus di pisahkan dari fungsi-fungsi lain. Contoh: bagian pembelian yang bertugas mempersiapkan *purchase order* tidak boleh menyetujui faktur dari pemasok dan untuk penerimaan barang harus

dikerjakan oleh fungsi/bagian lain dimana kontrol organisasi berperan sangat penting dalam proses pembelian yang berfungsi untuk mencegah seseorang mengerjakan tugas secara merangkap.

Dengan demikian dapat disimpulkan bahwa pembelian merupakan salah satu fungsi yang penting dalam berhasilnya operasi suatu perusahaan dalam suatu proses bisnis yang terlibat untuk mengidentifikasi keperluan persediaan dan pengadaan barang, penempatan pemesanan, penerimaan barang dan pengakuan hutang dimana perusahaan membeli barang jadi untuk di jual kembali yang terdapat jaringan prosedur dan fungsi-fungsi yang saling berkaitan.

2.4 Pengendalian Internal (*Internal Control*)

2.4.1 Pengertian Pengendalian Internal

Menurut Romney dan Steinbart (2015:226) Pengendalian internal adalah sebuah proses yang menyebar ke seluruh aktivitas pengoperasian perusahaan dan merupakan bagian integral dari aktivitas manajemen dimana pengendalian internal memberikan jaminan yang memadai untuk tujuan pengendalian berupa mengamankan aset, mengelola catatan secara detail yang baik untuk melaporkan aset perusahaan secara akurat dan wajar, memberikan informasi yang akurat dan reliabel, menyiapkan laporan keuangan yang sesuai dengan kriteria yang ditetapkan dan mendorong dan memperbaiki efisiensi operasional.

Menurut Mardi (2011:59) Pengendalian internal adalah suatu sistem yang melingkupi struktur organisasi beserta semua mekanisme dan ukuran-ukuran yang dipatuhi bersama untuk menjaga seluruh harta kekayaan organisasi dari berbagai arah.

Dengan demikian dapat disimpulkan bahwa pengendalian internal merupakan proses yang menyebar ke seluruh aktivitas pengoperasian perusahaan dan merupakan bagian integral dari aktivitas manajemen dan melingkupi struktur organisasi beserta semua mekanisme dan

ukuran-ukuran yang dipatuhi bersama untuk menjaga seluruh harta kekayaan organisasi dari berbagai arah.

2.4.2 Unsur Sistem Pengendalian Internal.

Menurut Mardi (2011:60) Agar suatu sistem pengendalian internal dapat berjalan secara efektif seperti yang di harapkan, maka harus memiliki unsur pokok sistem pengendalian internal sebagai berikut:

1. Struktur Organisasi.

Merupakan suatu kerangka pemisahan tanggung jawab secara tegas berdasarkan fungsi dan tingkatan unit yang dibentuk. Prinsip dalam menyusun struktur organisasi, yaitu pemisahan antara setiap fungsi yang ada dan suatu fungsi jangan diberikan tanggung jawab penuh dalam melaksanakan semua tahapan kegiatan.

2. Sistem wewenang dan prosedur pencatatan dalam organisasi.

Struktur organisasi harus dilengkapi dengan uraian tugas yang mengatur hak dan kewajiban masing-masing tingkatan beserta seluruh jajaran nya. Uraian tugas harus didukung petunjuk prosedur berbentuk peraturan pelaksanaan tugas disertai penjelasan mengenai pihak-pihak yang berwenang untuk mengesahkan kegiatan.

3. Pelaksanaan kerja sehat. Tata cara kerja secara sehat merupakan pelaksanaan yang dibuat sedemikian rupa sehingga mendukung tercapai nya tujuan pengendalian internal yang ditunjuk kan dalam beberapa cara.

4. Pegawai berkualitas. Salah satu unsur pokok penggerak organisasi adalah karyawan-karyawan harus berkualitas agar organisasi memiliki citra yang berkualitas. Secara umum, kualitas karyawan ditentukan oleh 3 aspek, yaitu pendidikan, pengalaman dan akhlak.

2.5 Persediaan

2.5.1 Pengertian Persediaan

Menurut Weygandt, Kimmel, Kieso (2011:250) Persediaan adalah aset yang di miliki perusahaan yang digunakan untuk di jual kembali kepada pelanggan dari suatu proses pengadaan barang atau persediaan bahan baku, proses pengerjaan yang digunakan dalam memproduksi barang menjadi persediaan barang jadi yang siap untuk dijual.

Menurut Weygandt, Kimmel, Kieso (2011:201) Untuk mengetahui persediaan akhir pada perusahaan dagang adalah persediaan awal di tambah biaya barang yang di beli (*cost of goods purchase*) yang akan menjadi harga barang tersedia untuk di jual (*cost of goods available for sale*) di kurangi harga pokok penjualan (*cost of goods sold*) hasil nya adalah persediaan akhir atau Persediaan Akhir = Persediaan awal + pembelian – HPP.

Dengan demikian dapat disimpulkan bahwa persediaan adalah aset yang di miliki perusahaan yang digunakan untuk di jual kembali kepada pelanggan dari bahan baku sampai barang jadi ataupun barang jadi yang siap dijual dengan rumus persediaan akhir = Persediaan awal + pembelian – HPP.

2.5.2 Metode Pencatatan Persediaan

Menurut Weygandt, Kimmel, Kieso (2011:201) Perusahaan menggunakan salah satu dari dua metode pencatatan persediaan yaitu *perpetual inventory system* atau *periodic inventory system*:

1. *Perpetual inventory system* adalah sistem persediaan di mana perusahaan menyimpan dan mencatat secara rinci dari biaya setiap pembelian persediaan dan penjualan, dengan proses bahwa perusahaan mencatat terus menerus yang menunjukkan persediaan yang dimiliki oleh perusahaan.
2. *Periodic inventory system* adalah sistem persediaan di mana perusahaan tidak menyimpan/merecord pencatatan persediaan secara

rinci selama periode akuntansi tetapi menentukan harga pokok penjualan hanya pada akhir periode akuntansi.

2.5.3 Metode Penilaian Persediaan

Penilaian persediaan di kelompokkan menjadi 3 jenis, yaitu:

1. *FIFO Method.*
2. *LIFO Method.*
3. *Average Method.*

Menurut Weygandt, Kimmel, Kieso (2011:255) FIFO (*First in First out*) adalah bahwa barang yang berasal dari pembelian awal adalah barang yang pertama untuk dijual. Penilaian persediaan FIFO setara dengan aliran fisik barang dagang, umumnya adalah proses bisnis dengan menjual unit pertama yang di beli.

Menurut Weygandt, Kimmel, Kieso (2011:274) LIFO (*Last in First out*) adalah bahwa barang yang terakhir di beli adalah barang pertama untuk dijual. Metode penilaian LIFO jarang bertepatan dengan arus fisik persediaan. Berdasarkan metode LIFO, biaya barang yang terakhir dibeli adalah yang pertama diakui dalam menentukan harga pokok penjualan (HPP).

Menurut Weygandt, Kimmel, Kieso (2011:255) *Average Cost* adalah metode biaya rata-rata dalam sistem persediaan yang menggunakan pencatatan perpetual yang disebut metode rata-rata bergerak (*moving average method*). dengan metode ini, perusahaan menghitung rata-rata baru berdasarkan pembelian persediaan dengan membagi biaya barang yang tersedia untuk dijual oleh unit di tangan (*quantity on hand*). *Average method* berlaku untuk: 1. unit dijual untuk menentukan harga pokok penjualan. 2. unit yang tersisa di tangan untuk menentukan jumlah persediaan akhir.

Dengan demikian dapat disimpulkan bahwa metode penilaian persediaan adalah suatu metode atau cara yang di pakai untuk menghitung nilai persediaan berdasarkan proses pembelian (barang masuk) maupun proses penjualan (barang keluar) yang diidentifikasi dari aktivitas bisnis berdasarkan persediaan fisik yang di nilai dari keluar – masuk nya barang yang terbagi dalam metode FIFO, LIFO dan *Average cost*.

2.6 Akuntansi

2.6.1 Pengertian Akuntansi

Definisi akuntansi menurut Weygandt, et al. (2011:4), Akuntansi adalah suatu sistem informasi yang mengidentifikasi, mencatat dan mengkomunikasikan kejadian ekonomi suatu organisasi kepada pihak yang berkepentingan.

2.6.2 Pengertian Hutang

Menurut Warren, Reeves dan Duchac (2011:11) Hutang Usaha adalah kewajiban yang di hasilkan dari pembelian dengan kredit atau sebagai anggaran.

Menurut Bodnar dan Hopwood (2011:334) Hutang usaha adalah salah satu cara pembayaran yang dapat dilakukan oleh perusahaan yang merupakan tanggung jawab dalam memenuhi pembayaran kepada *vendor*.

Dengan demikian dapat disimpulkan bahwa hutang adalah suatu catatan yang dibuat untuk melaksanakan pengakuan utang akibat ada nya prosedur distribusi pembelian dan merupakan suatu kewajiban perusahaan yang dihasilkan dari pembelian kredit dengan cara pembayaran yang dapat dilakukan perusahaan kepada *vendor*.

2.6.3 Pengertian Jurnal

Menurut Kieso, Weygandt dan kimmel (2011:56), jurnal disebut sebagai entry asli pembukuan, untuk setiap transaksi jurnal menunjukkan efek debit dan kredit pada akun tertentu. Dengan kata lain, jurnal merupakan catatan yang timbul sebagai efek dari transaksi yang terjadi dimana terdapat debit dan kredit untuk mencatatnya dengan akun spesifik.

Jurnal - jurnal yang berhubungan dengan transaksi pembelian, hutang usaha, dan pengeluaran kas adalah sebagai berikut:

Jurnal Pembelian (*Perpetual System*)

1. Pembelian Kredit

Dr – Persediaan	xxxxx
Cr – Hutang	xxxxx

2. Jurnal Pembalik (*Reversing Entries*)

Dr – Hutang	xxxxx
Cr – Persediaan	xxxxx

3. Jurnal Pengeluaran Kas

Dr – Hutang	xxxxx
Cr – Kas	xxxxx

2.6.4 Pengertian Kas

Menurut Weygandt, Kimmel, Kieso (20011:324) kas adalah alat keuangan yang berupa koin, mata uang, cek, wesel, *cash in hand*, atau berupa deposito di bank dimana dalam laporan keuangan perusahaan dapat menggabungkan kas bank dan kas kecil dan melaporkan total akumulasi saldo hanya sebagai kas.

Kas sangat berperan dalam menentukan kelancaran kegiatan perusahaan. Dapat disimpulkan bahwa suatu perusahaan harus memiliki anggaran kas untuk menjaga posisi likuiditas dan untuk mengetahui defisit dan surplus kas.

2.6.5 Pengertian Pengeluaran Kas

Menurut James. A. Hall (2008:243) pengeluaran kas adalah suatu sistem proses pembayaran kewajiban akibat timbul nya hutang yang berasal dari sistem pembelian dimana sistem mengakumulasi kreditur dalam menerima pembayaran dan total pembayaran secara tepat waktu dan benar. Prosedur dalam pengeluaran kas yaitu:

1. *Identify liabilities due.* Proses pengeluaran kas di mulai dari AP department yang mengidentifikasi setiap barang yang akan di bayar sudah jatuh tempo. Fungsi hutang membuka AP file dan mengirim persetujuan oembayaran kepada department pengeluaran kas.
2. *Prepare Cash Disbursement.* Pada proses pengeluaran kas, clerk menerima voucher dan mereview setiap dokumen untuk ketepatan dan kesempurnaan pencatatan kemudian clerk menyiapkan sebuah cek dan catatan nomor cek, jumlah tagihan, nomor voucher dan data data lain terkait check register.
3. *Update AP Record.* Ketika menerima voucher, AP clerk mengurangi jumlah hutang dengan men debit akun hutang atau melalui pencatatan nomor cek dan pembayaran registrasi voucher.
4. *Post to General Ledger.* Fungsi buku besar yaitu menerima voucher jurnal dari pengeluaran kas dan ringkasan akun dari AP. Setiap voucher menampilkan total pengurangan kewajiban perusahaan dan akun kas sebagai hasil pembayaran kepada supplier.

Menurut Gupta, M., Palmer, R. J., & Brandt, J. (2014) Dalam aktivitas pembelian diperlukan kegiatan pembayaran (pengeluaran kas) untuk menunjang proses pembelian yang dilakukan oleh bagian keuangan dengan meninjau strategi pembayaran untuk pembayaran barang atau jasa terutama untuk pengeluaran kas besar.

Dengan demikian dapat disimpulkan bahwa pengeluaran kas suatu catatan yang dibuat untuk melaksanakan kegiatan pengeluaran baik dengan cek maupun dengan uang tunai yang digunakan untuk kegiatan umum perusahaan dan merupakan suatu sistem proses pembayaran kewajiban akibat timbul nya hutang yang berasal dari sistem pembelian

dimana sistem mengakumulasi kreditur dalam menerima pembayaran dan total pembayaran secara tepat waktu dan benar.

2.7 Peranan Sistem Informasi dan Akuntansi Dalam Organisasi

Menurut Mardi (2011:10) Perancangan Sistem Informasi Akuntansi yang tepat dapat memberikan nilai tambah yang optimal bagi perusahaan. Tujuan merancang SIA yaitu dapat menghasilkan hal-hal berikut ini:

1. Jika sistem dan tata prosedur kerja ditata secara tepat maka produk yang dihasilkan lebih efisien, melalui SIA dapat dibuat SOP sehingga tidak ada pekerjaan yang menyimpang dan memudahkan pengendalian produksi oleh manajer.
2. Sebuah pekerjaan yang dilakukan terencana sesuai prosedur dapat meningkatkan efisiensi. Perancangan SIA yang baik dapat membantu memperbaiki efisiensi jalan nya suatu proses, misal nya ketersediaan data dan informasi secara tepat waktu.
3. Informasi yang diterima dengan tepat waktu dapat meningkatkan kualitas pengambilan keputusan. Melalui SIA dapat dihasilkan informasi yang akurat sehingga pengambilan keputusan dengan informasi tepat waktu dapat meningkatkan kinerja perusahaan.
4. Merancang SIA secara baik dan utuh akan mempermudah proses alih pengetahuan dan pengalaman, terutama pada tingkat operator dan desainer. Semua kreativitas yang muncul dari penalaran pengetahuan akan meningkatkan keunggulan perusahaan.

2.8 Pajak

2.8.1 Pengertian Pajak

Menurut Siti Resmi (2014:1) Pajak adalah peralihan kekayaan dari pihak rakyat kepada kas negara untuk membiayai pengeluaran rutin dan “surplus” nya di gunakan untuk *public saving* yang merupakan sumber utama untuk membiayai *public investment*.

Menurut Siti resmi (2014:2) Ciri-ciri yang melekat pada pajak yaitu:

1. Pajak dipungut berdasarkan atau dengan kekuatan undang-undang serta aturan pelaksanaan nya.
2. Dalam pembayaran pajak tidak dapat ditunjukkan ada nya kontraprestasi individual oleh pemerintah.
3. Pajak di pungut oleh negara, baik pemerintah pusat maupun pemerintah daerah.
4. Pajak diperuntukan bagi pengeluaran-pengeluaran pemerintah bila dari pemasukan nya masih terdapat surplus dan digunakan untuk membiayai *public investment*.

2.9 Analisis dan Perancangan Berorientasi Objek

A. Analisis Berorientasi Objek (*Object Oriented Analysis*)

Menurut Satzinger, Jackson and Burd (2012:241), *Object Oriented Analysis* adalah suatu objek atau hal-hal yang berkaitan untuk melakukan pekerjaan dan menentukan setiap interaksi pengguna (*user*) untuk menyelesaikan setiap tugas-tugas nya.

B. Desain Berorientasi Objek (*Object Oriented Design*)

Menurut Satzinger, Jackson and Burd (2012:241), *Object Oriented Design* mendefinisikan semua jenis tipe objek yang diperlukan untuk berkomunikasi dengan orang-orang dan perangkat dalam suatu sistem dan

menunjukkan bagaimana sistem tersebut menyelesaikan tugas nya sehingga dapat di implementasikan dengan bahasa atau lingkungan tertentu.

2.9.1 *Objects*

Menurut Satzinger, Jackson, dan Burd (2012:241), *Objects* adalah wujud di dalam sistem komputer yang memiliki kemampuan merespon pesan. Dan merupakan pandangan yang berbeda dari sistem komputer yang membutuhkan pendekatan yang berbeda dalam hal analisis sistem, desain sistem dan pemrograman.

2.10 *Attributes, Methods, and User Interface*

2.10.1 *Pengertian Attributes*

Menurut Satzinger, Jackson, dan Burd (2012:311), *attributes* adalah suatu karakteristik objek yang memiliki karakteristik yang sama dalam suatu tingkatan kelas, dan berisi nilai serta jenis yang sama.

2.10.2 *Pengertian Methods*

Menurut Satzinger, Jackson, dan Burd (2012:310), *Methods* adalah perilaku atau operasi yang menggambarkan kemampuan dari sebuah objek dan menunjukkan pewarisan dari sebuah *class*. 3 jenis *methods* yang ditemukan yaitu:

1. *Constructor methods.*
2. *Get data and data set methods.*
3. *Use case methods.*

2.10.3 *Pengertian User Interface Object*

Menurut Satzinger, Jackson, dan Burd (2012:163), *User interface object* adalah sebuah objek yang berinteraksi dengan *user* saat menggunakan sistem dan memanfaatkan pengalaman pengguna dengan sistem antar muka yang sama pada platform tersebut.

2.11 *Classes dan Identity*

2.11.1 *Pengertian Classes*

Menurut Satzinger, Jackson, dan Burd (2012:101), *Classes* adalah jenis atau penggolongan yang memiliki kesamaan objek. *Classes* menggambarkan semua objek *class* apa yang diwakili.

2.12 *Encapsulation*

2.12.1 *Pengertian Encapsulation*

Menurut Satzinger, Jackson, dan Burd (2012:241), *Encapsulation* adalah melakukan penggabungan *attributes* dan metode dalam satu unit dan struktur internal objek disembunyikan.

2.13 *Unified Modelling Language (UML)*

2.13.1 *Pengertian Unified Modelling Language (UML)*

Menurut Satzinger, Jackson, dan Burd (2012:46), *unified modelling language* adalah bentuk standar dari model konstruksi dan notasi yang dikembangkan secara khusus untuk pengembangan berorientasi objek yang di definisikan oleh OMG (Object Management Group), ialah sebuah organisasi standar dalam pengembangan UML.

Contoh UML:

1. *Use case Diagram*
2. *Class Diagram*
3. *Sequence diagram*
4. *Communication diagram*
5. *State machine.*

2.14 *Komponen Analisis dan Perancangan Berorientasi Objek*

2.14.1 *Activity Diagram*

Menurut Satzinger, Jackson, dan Burd (2012:57), *Workflow* adalah urutan yang terstruktur dari suatu langkah-langkah pengolahan yang benar dalam menangani suatu transaksi bisnis dimana mungkin memiliki alur kerja yang sederhana atau kompleks yang dapat terdiri dari puluhan atau bahkan ratusan langkah-langkah pengolahan termasuk

dari berbagai bagian organisasi. *Activity diagram* merupakan penggambaran suatu kegiatan yang berurutan dari aliran proses bisnis melalui orang yang melakukan setiap aktivitas, dan aliran yang berurutan dari aktivitas tersebut. Simbol yang digunakan yaitu :

A. *Starting Activity (pseudo)*

Merupakan simbol untuk menandakan dimulainya aktivitas.

B. *Transition Arrow*

Merupakan garis penunjuk arah urutan aktivitas yang menggambarkan transisi dari suatu aktivitas.

C. *Activity*

Merupakan simbol yang menggambarkan aktivitas.

D. *Ending Activity (pseudo)*

Merupakan simbol untuk menandakan berakhirnya aktivitas.

E. *Swimlane*

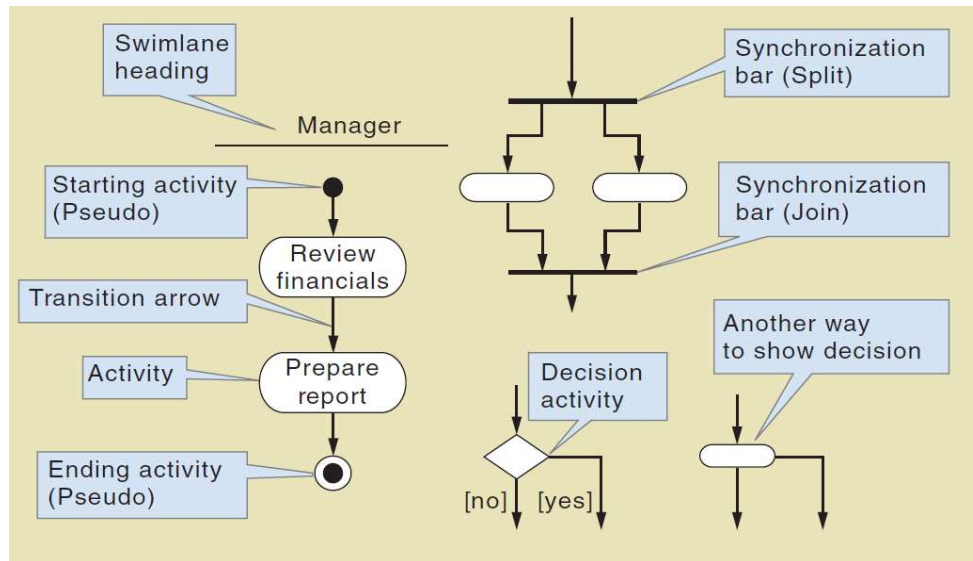
Merupakan area persegi dalam *activity diagram* yang menunjukkan aktivitas diselesaikan *single agent*.

F. *Synchronization bar*

Merupakan simbol yang digunakan untuk mengontrol pemisahan atau penyatuan dari jalur berurutan contoh nya *join* dan *split*.

G. *Decision Activity*

Merupakan simbol poin keputusan dalam alur proses mengikuti satu jalur atau jalur lainnya.

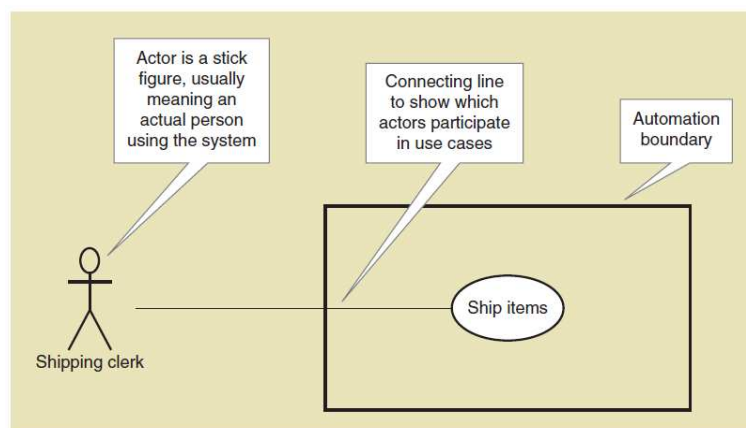


Gambar 2.1 Notasi Activity Diagram

(Sumber: Satzinger, Jackson, dan Burd 2012:58)

2.14.2 Use Case Diagram

Menurut Satzinger, Jackson, dan Burd (2005:78), “*Use case* adalah menggambarkan suatu hubungan kegiatan dalam UML model yang digunakan untuk menunjukkan interaksi antara penggunaan proses dengan aktor tertentu.



Gambar 2.2 Contoh Notasi Use Case Diagram

(Sumber: Satzinger, Jackson, dan Burd 2012:81)

Notasi yang terdapat dalam *use case diagram*:

1. *Actor*: Menggambarkan seseorang yang menggunakan *system* untuk melakukan proses tertentu.
2. *Connector*: Menunjukkan hubungan partisipasi actor dengan *use case*.
3. *Automation boundary*: Menggambarkan tempat terjadi nya hubungan proses antara *actor* dengan *use case*.

2.14.3 Use Case Description (fully developed)

Menurut Satzinger, Jackson, dan Burd (2012:121), “*Usecase description* adalah daftar yang berisi rincian pengolahan untuk kasus penggunaan yang memberikan gambaran berdasarkan *use case diagram* dimana berisi informasi rinci tentang setiap kasus penggunaan yang di gambarkan menggunakan deskripsi kasus bisnis. Contoh bagian yang terdapat dalam use case description:

1. Use case name.
2. Scenario.
3. Triggering event.
4. Brief Description.
5. Actors.
6. Related Use Case.
7. Stakeholders.
8. Preconditions.
9. Postconditions.
10. Flow of activities.
11. Exception conditions.

2 Fully developed use case description for Create customer account

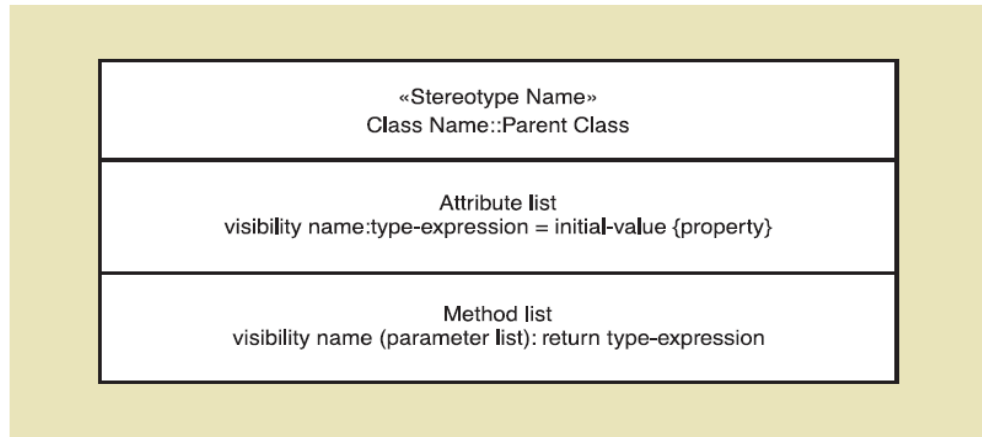
Use case name:	Create customer account.									
Scenario:	Create online customer account.									
Triggering event:	New customer wants to set up account online.									
Brief description:	Online customer creates customer account by entering basic information and then following up with one or more addresses and a credit or debit card.									
Actors:	Customer.									
Related use cases:	Might be invoked by the <i>Check out shopping cart</i> use case.									
Stakeholders:	Accounting, Marketing, Sales.									
Preconditions:	Customer account subsystem must be available. Credit/debit authorization services must be available.									
Postconditions:	Customer must be created and saved. One or more Addresses must be created and saved. Credit/debit card information must be validated. Account must be created and saved. Address and Account must be associated with Customer.									
Flow of activities:	<table><tr><th>Actor</th><th>System</th></tr><tr><td>1. Customer indicates desire to create customer account and enters basic customer information.</td><td>1.1 System creates a new customer. 1.2 System prompts for customer addresses.</td></tr><tr><td>2. Customer enters one or more addresses.</td><td>2.1 System creates addresses. 2.2 System prompts for credit/debit card.</td></tr><tr><td>3. Customer enters credit/debit card information.</td><td>3.1 System creates account. 3.2 System verifies authorization for credit/debit card. 3.3 System associates customer, address, and account. 3.4 System returns valid customer account details.</td></tr></table>	Actor	System	1. Customer indicates desire to create customer account and enters basic customer information.	1.1 System creates a new customer. 1.2 System prompts for customer addresses.	2. Customer enters one or more addresses.	2.1 System creates addresses. 2.2 System prompts for credit/debit card.	3. Customer enters credit/debit card information.	3.1 System creates account. 3.2 System verifies authorization for credit/debit card. 3.3 System associates customer, address, and account. 3.4 System returns valid customer account details.	
Actor	System									
1. Customer indicates desire to create customer account and enters basic customer information.	1.1 System creates a new customer. 1.2 System prompts for customer addresses.									
2. Customer enters one or more addresses.	2.1 System creates addresses. 2.2 System prompts for credit/debit card.									
3. Customer enters credit/debit card information.	3.1 System creates account. 3.2 System verifies authorization for credit/debit card. 3.3 System associates customer, address, and account. 3.4 System returns valid customer account details.									
Exception conditions:	1.1 Basic customer data are incomplete. 2.1 The address isn't valid. 3.2 Credit/debit information isn't valid.									

Gambar 2.3 Contoh Use Case Description (fully developed)

(Sumber: Satzinger, Jackson, dan Burd 2012:123)

2.14.4 Class Diagram

Menurut Satzinger, Jackson, dan Burd (2012:101), *Class Diagram* menggambarkan struktur dan deskripsi kelas, *package*, dan objek serta menunjukkan kelas objek untuk sistem.



Gambar 2.4 Notasi Class Diagram
(Sumber: Satzinger, Jackson, dan Burd 2012:310)

Format yang digunakan untuk menentukan masing-masing atribut :

A. Attribute visibility

Visibility menunjukkan apakah objek lain dapat mengakses atribut secara langsung atau tidak. Tanda + (plus) mengindikasikan atribut dapat terlihat atau publik, dan tanda – (minus) menandakan bahwa atribut tidak dapat terlihat atau *private*.

B. Attribute name

C. Type – expression

Dapat berupa karakter, string, integer, *number*, *currency*, atau *date*.

D. Initial value

E. Property

Ditempatkan dalam kurung kurawal. Contohnya, {key}.

Format yang digunakan dalam *method list* :

A. *Method visibility*

B. *Method name*

C. *Method parameter list* : *argument* yang masuk.

D. *Return type expression* (pengembalian parameter dari suatu method).

Menurut Satzinger, Jackson, dan Burd (2012:104), terdapat 2 hirarki dalam notasi *class diagram*, yaitu :

A. *Generalization/specialization notation*

Generalization/specialization didasarkan oleh pengelompokkan hal-hal berdasarkan persamaan dan perbedaan. *Generalization* adalah pengelompokkan hal-hal dengan jenis yang sama, contohnya ada banyak jenis kendaraan seperti mobil, motor, sepeda, pesawat, dan sebagainya. Sedangkan *specialization* adalah pengelompokkan jenis-jenis hal yang berbeda, sebagai contoh jenis khusus dari mobil adalah mobil *sport*, *sedan*, *jeep*, dan sebagainya. *Generalization/specialization hierarchy* digunakan untuk mengurutkan hal-hal umum menjadi lebih khusus. Hubungan antara *generalization* dan *specialization* digunakan untuk hal yang terstruktur atau peringkat yang lebih kepada hal umum ke khusus, disebut *superclass* jika setiap class memiliki class umum satu sama lain kemudian disebut *subclass* jika setiap class memiliki class yang lebih khusus.

B. *Whole-part hierarchy notation*

Whole-part hierarchies menggambarkan hubungan keterkaitan antara satu class dengan class lain nya yang merupakan bagian dari class tersebut.

Menurut Satzinger, Jackson, dan Burd (2012:106) Ada dua jenis *whole-part hierarchies*, yaitu *aggregation* dan *composition*. *Aggregation* digunakan untuk menggambarkan sebuah hubungan antara

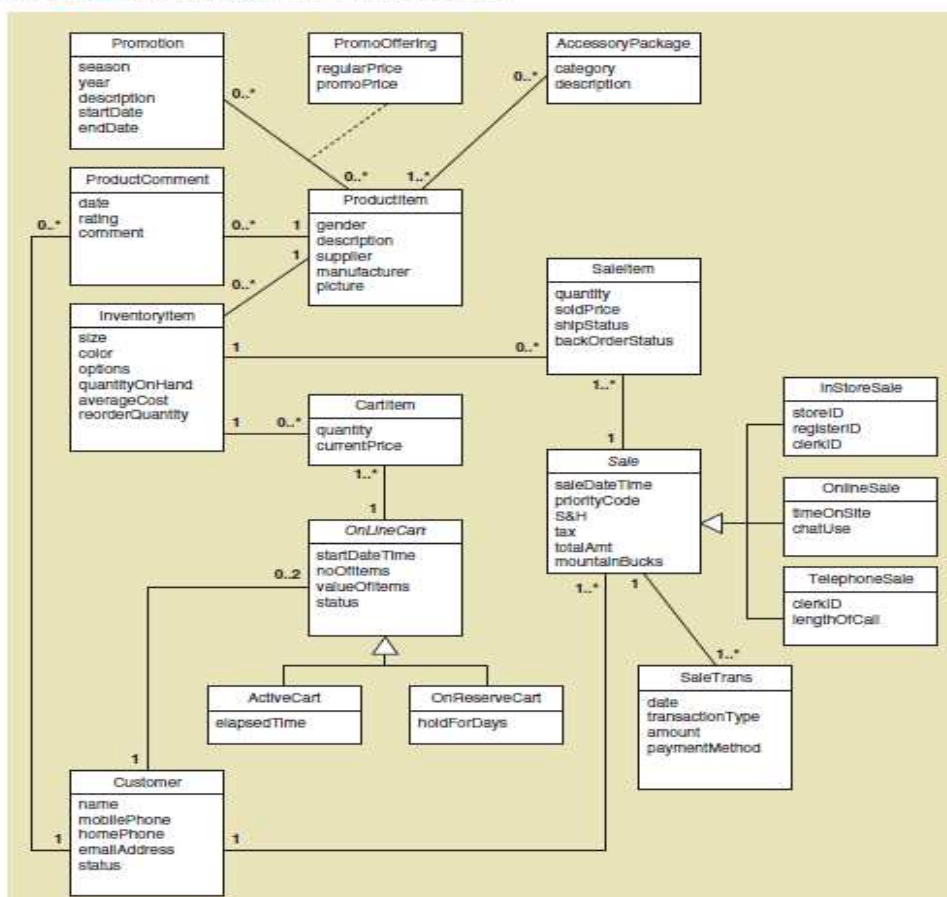
agregat (keseluruhan) dan komponennya (bagian-bagian) dimana bagian-bagian tersebut dapat berdiri sendiri secara terpisah, sedangkan composition digunakan untuk menggambarkan hubungan keterikatan yang lebih kuat, dimana tiap-tiap bagian tidak dapat berdiri sendiri secara terpisah.

Class Diagram memiliki 3 desain, yaitu:

A. Domain Mode Class Diagram

Menurut Satzinger, Jackson, dan Burd (2012:101) *Domain model class diagram* adalah suatu *class* yang menunjukkan hubungan pada masalah pengguna domain dimana *domain class diagram* terdiri dari nama *class* dan *attribute* yang menyatakan objek-objek *class* di dalam nya.

FIGURE 4-21 RMO Sales subsystem domain model class diagram

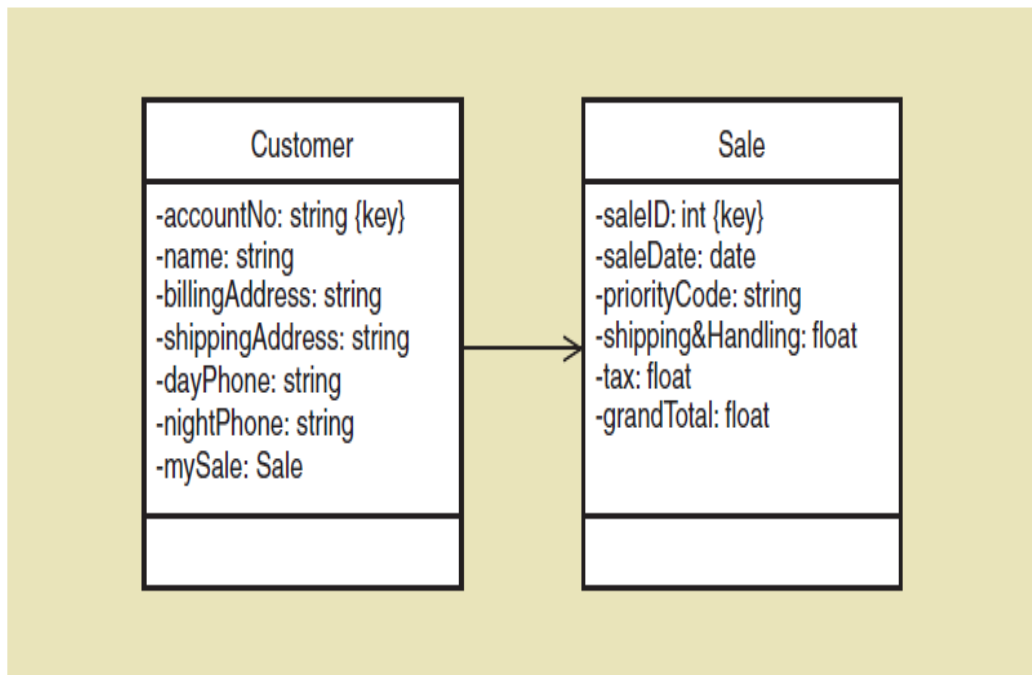


Gambar 2.5 Contoh Domain Class Diagram

(Sumber: Satzinger, Jackson, dan Burd 2012:109)

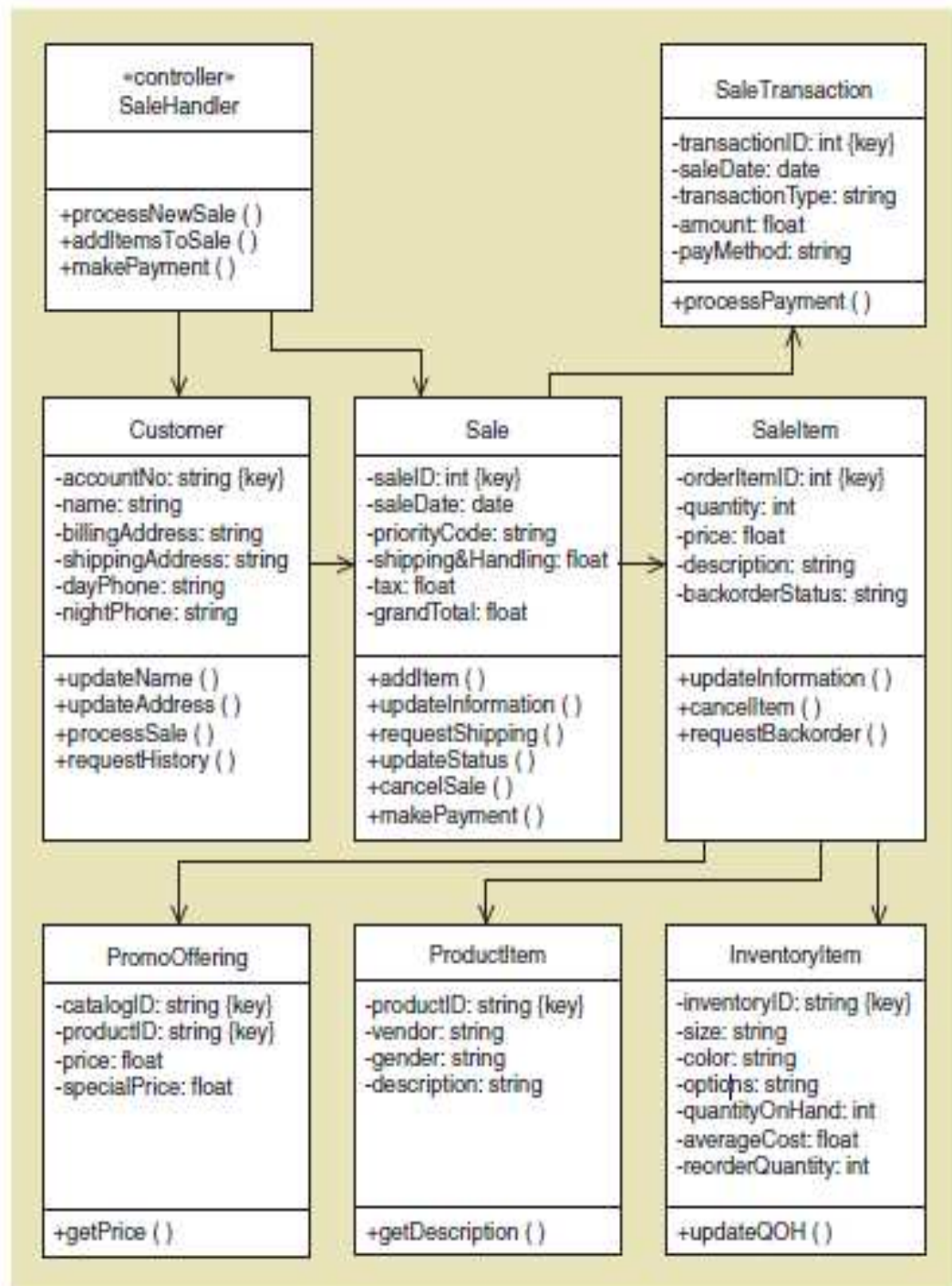
B. First Cut Class Diagram

Menurut Satzinger, Jackson, dan Burd (2012:312), “*First Cut Class Diagram* dikembangkan dengan memperluas *domain model class diagram*. Perluasan ini membutuhkan dua langkah: (1) melakukan elaborasi atribut dengan informasi *type and initial value* dan (2) menambahkan panah navigasi. Penggunaanya hanya fokus pada kasus yang terlibat dalam penggunaan *model class*.”



Gambar 2.6 Contoh *First Cut Class Diagram*
(Sumber: Satzinger, Jackson, dan Burd 2012:313)

C.Updated Design Class Diagram



Gambar 2.7 Contoh Updated Design Class Diagram

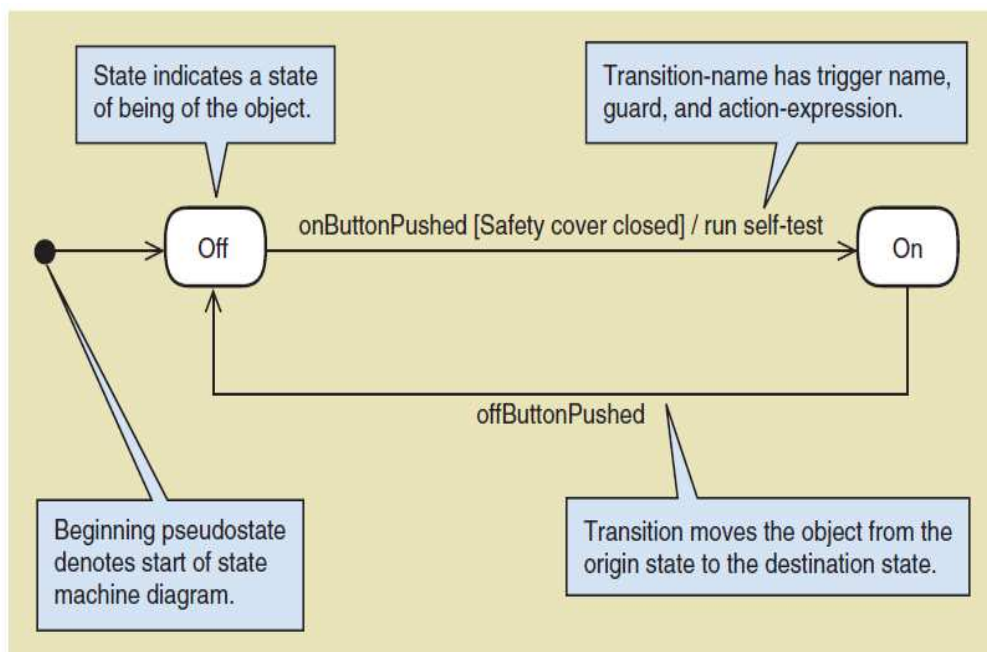
(Sumber: Satzinger, Jackson, dan Burd 2012:317)

2.14.5 State Machine Diagram

Menurut Satzinger, Jackson, dan Burd (2012:144) *state machine diagram* digunakan sebagai *model* objek dari suatu fungsi kepada transisi fungsi yang lain nya dalam *use case*.

Menurut Satzinger, Jackson, dan Burd (2012:134) Transition adalah suatu pergerakan objek dari suatu fungsi ke fungsi yang lain nya, proses transisi ialah mekanisme yang menyebabkan suatu objek untuk meninggalkan fungsi yang lama berpindah ke fungsi yang baru. Pengembangan transisi inilah yang di sebut dengan *state machine diagram*. *State machine diagram* dapat di kembangkan berdasarkan *problem domain class diagram* yang memiliki perilaku yang kompleks serta perlu di lacak setiap proses nya. Notasi dalam *state machine* dapat terbagi atas:

1. *Pseudostate*.
2. *State*.
3. *Transition*.



Gambar 2.8 Notasi State machine Diagram

(Sumber: Satzinger, Jackson, dan Burd 2012:134)

2.14.6 *System Sequence Diagram*

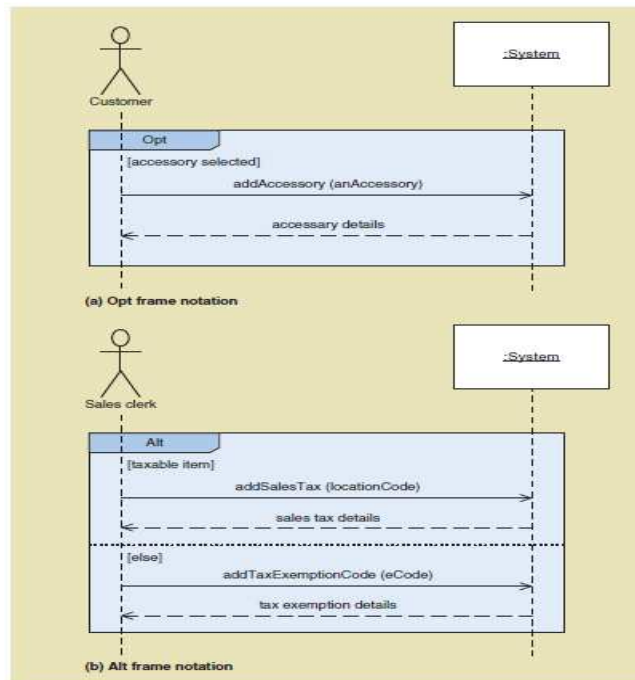
Menurut Satzinger, Jackson, dan Burd (2012:196) *System Sequence Diagram (SSD)* adalah suatu inisialisasi dokumentasi dialog.

Menurut Satzinger, Jackson, dan Burd (2012:211) SSD untuk mengidentifikasi suatu pesan masuk dan keluar untuk setiap kegiatan berdasarkan aktivitas proses yang ada.

Menurut Satzinger, Jackson, dan Burd (2012:126) SSD digunakan untuk menggambarkan aliran informasi *input* dan *output* dari sistem secara otomatis dimana *input* dan *output* mengidentifikasi interaksi antara *actor* dengan *system*.

Menurut Satzinger, Jackson, dan Burd (2012:127) Motasi yang berlabel :System adalah objek yang mewakili seluruh sistem secara otomatis. Di antara *actor* dan :system terdapat sebuah garis putus-putus vertikal yang di sebut *lifeline*, sebuah *lifeline* menghubungkan aktivitas bolak-balik antara *actor* dengan :system dan panah yang digunakan *actor* untuk mengirim pesan dimana setiap panah memiliki asal dan tujuan pesan tersebut dikirim kan.

Dapat disimpulkan bahwa diagram *sequence system* adalah sebuah diagram yang menunjukkan urutan pesan masukan dan keluaran pada aktor dan sistem selama sebuah *usecase* atau skenario tunggal.

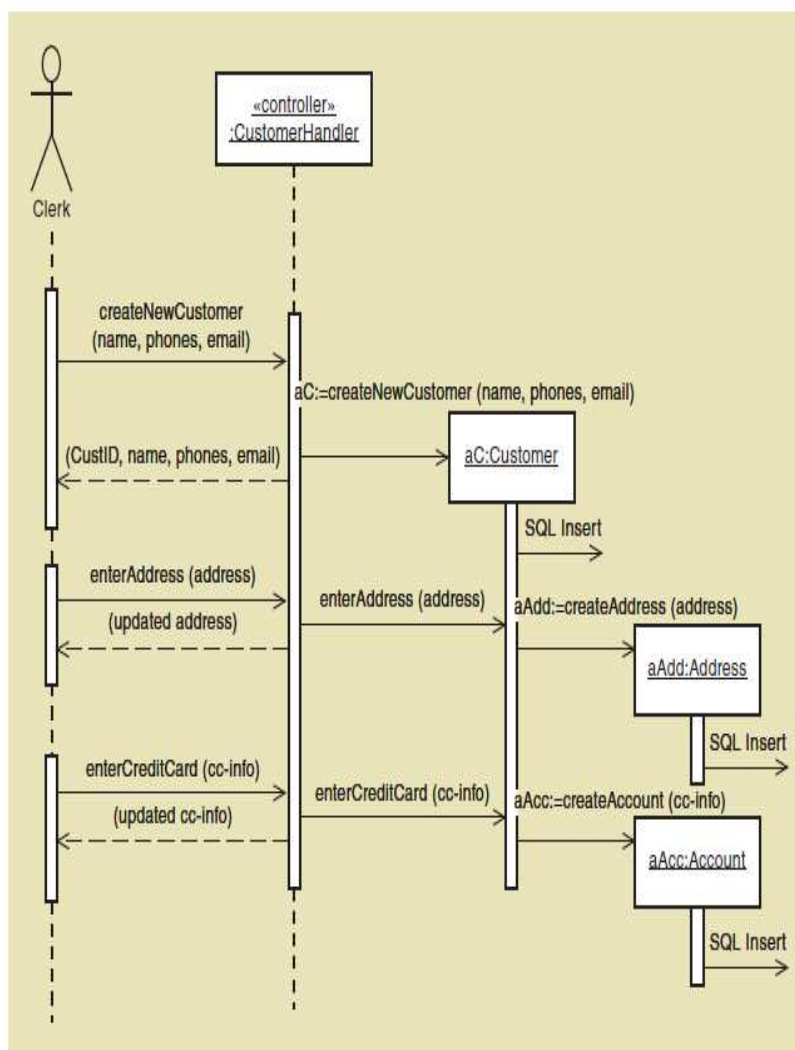


Gambar 2.9 Notasi *System Sequence Diagram*
 (Sumber: Satzinger, Jackson, dan Burd 2012:130)

2.14.7 *First Cut Sequence Diagram*

Menurut Satzinger, Jackson, dan Burd (2012:338 -339) *First Cut Sequence Diagram* digunakan untuk mengidentifikasi semua *class domain* dan diperlukan pesan *internal* antara *actor* dengan *class domain* kemudian ada penambahan pada *view layer* dan *data access layer*. Ketika mengidentifikasi dan menciptakan suatu pesan, pertama harus menentukan asal dan tujuan objek tersebut dimana setiap objek akan diperlukan untuk memulai pesan. Setiap pesan harus mencerminkan *request* yang dikirim.

FIGURE 11-8 First-cut sequence diagram for the Create customer account use case



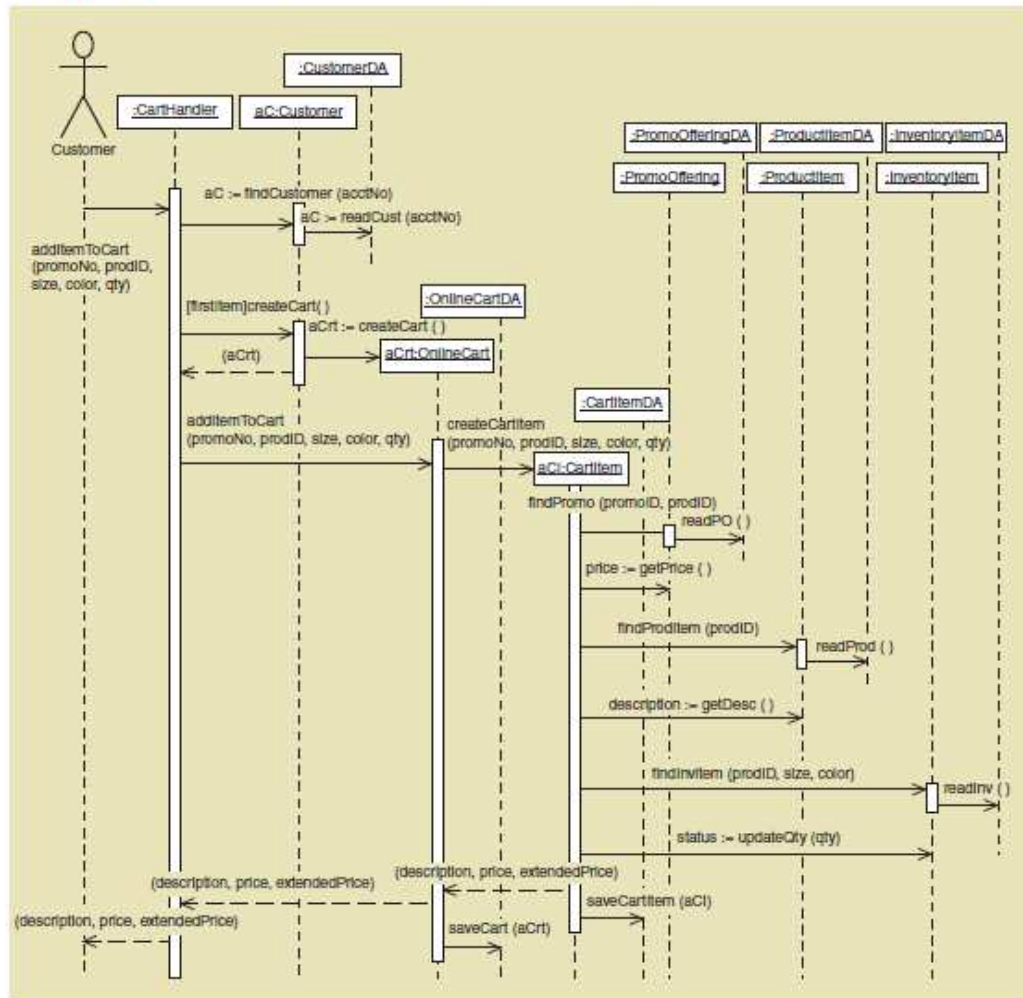
Gambar 2.10 Contoh *First Cut Sequence Diagram*
(Sumber: Satzinger, Jackson, dan Burd 2012:338)

2.14.8 Data Access Layer Diagram

Menurut Satzinger, Jackson, dan Burd (2012:345), Prinsip pemisahan tanggung jawab diberlakukan pada data *access layer*. Pada sistem yang besar atau kompleks desainer membuat desain *three layer* yang termasuk *class* yang bertanggung jawab dalam mengeksekusi *database SQL* dan dapat menampilkan hasil *query* serta memberikan informasi kepada lapisan *domain*. Dengan ada nya *multi layer design* dapat digunakan untuk mendukung pembangunan jaringan *database*

server dalam satu mesin dan *user interface* pada beberapa *desktop* dapat menciptakan sistem yang lebih kuat dan lebih fleksibel berdasarkan *data access layer*.

FIGURE 11-13 Sequence diagram for the Fill shopping cart use case with data access layer



Gambar 2.11 Contoh *Data Access Layer Sequence Diagram*

(Sumber: Satzinger, Jackson, dan Burd 2012:346)

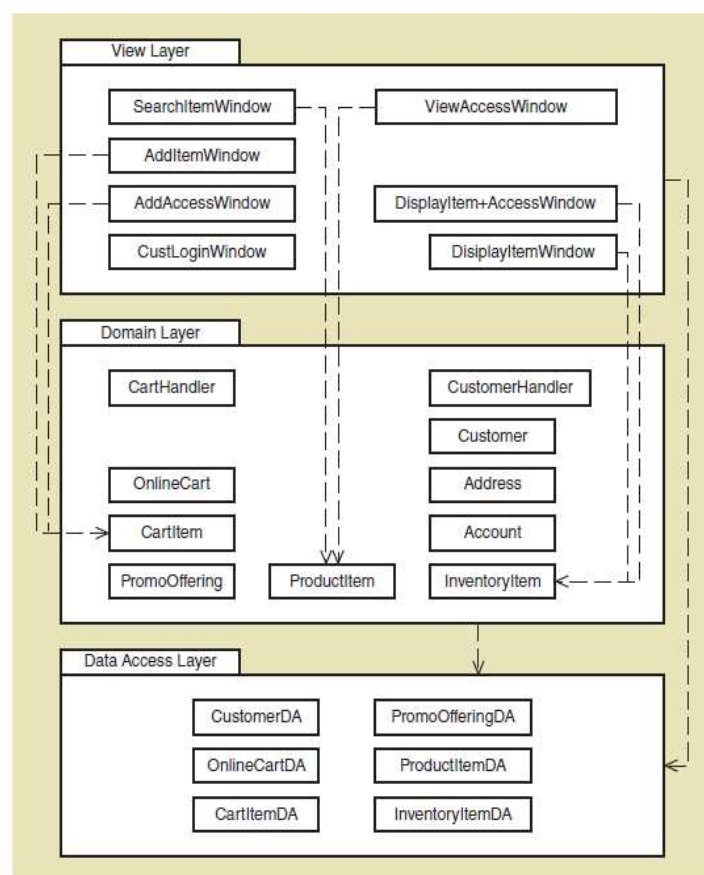
2.14.9 Package Diagram

Menurut Satzinger, Jackson, dan Burd (2012:353) Package Diagram merupakan desain tingkat tinggi yang memungkinkan desainer untuk mengasosiasikan class ke dalam kelompok lain dan class ditempatkan ke dalam setiap package yang dikembangkan berdasarkan

communication diagram. Simbol-simbol yang digunakan dalam package diagram adalah:

1. Dashed Arrow yaitu yang merepresentasikan hubungan dependensi/ketergantungan.
2. Arrow tail yaitu berupa panah yang terhubung antara package yang terlibat.
3. Arrow head yaitu berupa panah yang menghubungkan terhadap *independen package*.

FIGURE 11-19
Partial design of three-layer package diagram for RMO



Gambar 2.12 Contoh Package Diagram
(Sumber: Satzinger, Jackson, dan Burd 2012:353)

2.14.10 User Interface

Menurut Satzinger, Jackson, dan Burd (2012:189), *system interface* adalah suatu *input* dan *output* yang memerlukan campur tangan manusia dan memiliki input secara otomatis yang berasal dari *system* lain atau setiap transaksi yang diperoleh dari *system* lain. *System interface* mengirim pesan dan informasi kepada *system* lain. *User Interface* adalah suatu *input* dan *ouput* yang langsung melibatkan sistem pengguna (*user*) dimana *user interface* dapat mengenali karakteristik pengguna dan karateristik perangkat *interface* tertentu. Setiap *user interface* harus di rancang untuk memaksimalkan kemudahan pengguna (*user*) yang telah di optimalkan.

Di sebagian besar pengembangan proyek sistem, *analyst* memisahkan *system interface* dan *user interface* karena masing-masing komponen memiliki kemampuan dan teknologi nya sendiri. Tetapi tetap diperlukan koordinasi dan desain komponen sistem.



Gambar 2.13 Contoh User Interface

(Sumber: Satzinger, Jackson, dan Burd 2012:203)

2.14.11 Fase Siklus Pengembangan Sistem

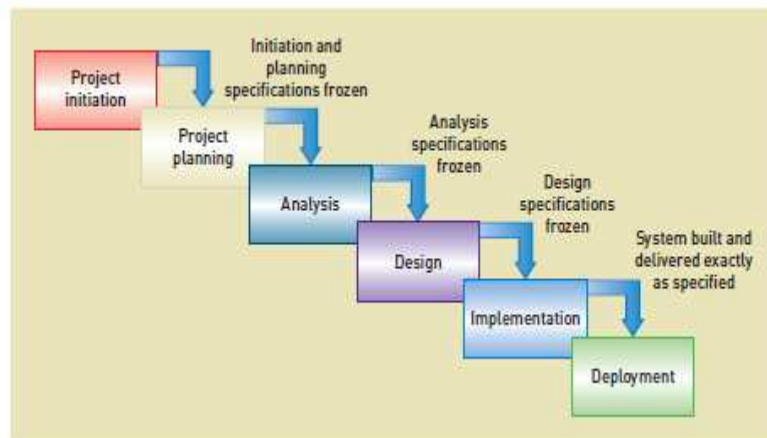
Menurut Satzinger, Jackson dan Burd (2012:06), *System Development Life Cycle* (SDLC) adalah suatu pengembangan sistem yang mengidentifikasi semua kegiatan yang diperlukan untuk membangun, meluncurkan dan memelihara sistem informasi. Biasanya SDLC mencakup semua kegiatan yang merupakan bagian dari analisis sistem, desain sistem, pemrograman, pengujian, dan pemeliharaan sistem serta proses manajemen proyek lainnya yang diperlukan untuk keberhasilan penyebaran sistem informasi. Terdapat 6 proses inti yang diperlukan untuk pengembangan aplikasi baru, yaitu:

1. Identifikasi masalah atau kebutuhan yang diperlukan dan mendapatkan persetujuan untuk beralih ke proses selanjutnya.
2. Merencanakan dan memonitor proyek apa saja yang harus dilakukan, bagaimana melakukannya dan siapa yang melakukannya.
3. Menemukan dan memahami detail dari masalah atau kebutuhan sistem.
4. Mendesain komponen sistem untuk memecahkan masalah dan menemukan solusi berdasarkan kebutuhan.
5. Membangun, mengetest dan mengintegrasikan komponen sistem.
6. Menyelesaikan test sistem dan menyebarkan atau menginformasikan solusi kepada user.

Menurut Satzinger, Jackson dan Burd (2012:228 - 229) bahwa pendekatan SDLC yang paling memungkinkan (*predictable*) adalah waterfall model/model air terjun dimana pada model ini pembangunan sistem dapat dikerjakan satu demi satu atau secara berurutan. Sistem dikembangkan melalui pemrograman, pengujian dan penginstalan. Dalam waterfall model memerlukan 6 tahapan, yaitu:

1. Inisiasi.
2. Perencanaan.
3. Analisis.
4. Desain.
5. Implementasi.
6. Aktivitas penyebaran sistem.

FIGURE 8-3 Waterfall model of the SDLC



Gambar 2.14 *Waterfall model of the SDLC*

(Sumber: Satzinger, Jackson, dan Burd 2012:229)

2.15 Database

2.15.1 Pengertian Database

Menurut Conolly, Begg (2015:52) *Database system* adalah suatu kumpulan program aplikasi yang saling berinteraksi dengan database lain bersamaan dengan DBMS (*Database Management System*) dan berinteraksi dengan *database* itu sendiri.

Menurut Conolly, Begg (2015:63) *Database* adalah suatu kumpulan/pengkoleksian data secara logis dalam waktu yang bersamaan dan dirancang untuk memenuhi kebutuhan informasi dari sebuah organisasi.

2.15.2 Pengertian *Database Management System (DBMS)*

Menurut Conolly, Begg (2015:64) DBMS (*Database Management System*) adalah suatu system perangkat lunak yang memungkinkan pengguna untuk mendefinisikan, membuat, memelihara dan mengontrol akses database. Tipe-tipe DBMS yaitu mempunyai fasilitas:

1. DDL (*Data Definition Language*) yaitu fasilitas DBMS yang memungkinkan pengguna untuk mendefinisikan database, seperti menentukan tipe data, struktur data dan kendala pada data yang akan disimpan dalam database.
2. DML (*Data manipulation Language*) yaitu fasilitas DBMS yang memungkinkan pengguna untuk melakukan *insert*, *update*, *delete* dan mengambil data dari database serta memiliki pusat repositori untuk semua data dan deskripsi data yang memungkinkan sistem DML untuk memberikan fasilitas manipulasi *database* dengan menggunakan bahasa *query* (*query language*) dan pada umumnya penggunaan bahasa *query* yang umum yaitu SQL (*Structure Query Language*).

2.16 *Unified Process*.

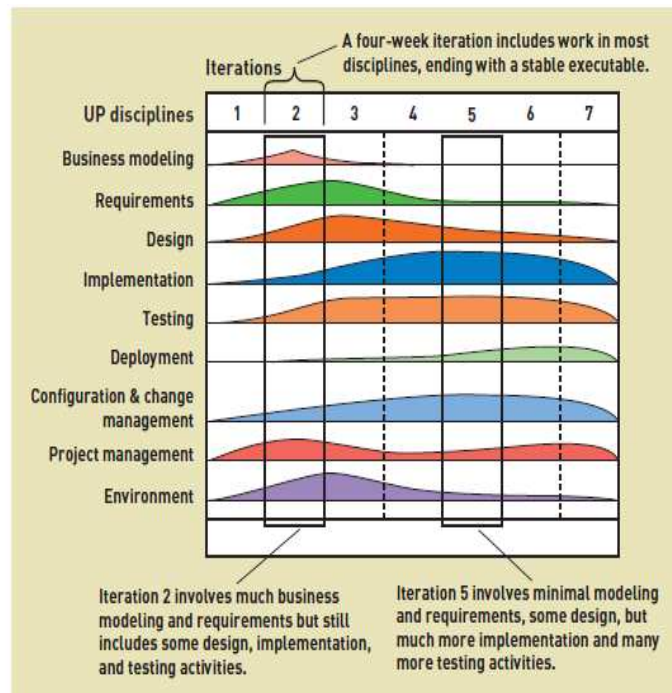
2.16.1 Pengertian *Unified Process (UP)*

Menurut Satzinger, Jackson dan Burd (2012:447) *Unified Process* (UP) adalah suatu disiplin ilmu yang mendefinisikan setiap iterasi secara berurutan berdasarkan 4 fase UP yaitu *inception*, *elaboration*, *construction* dan *transition*.

Setiap kegiatan UP merupakan aktivitas fungsional yang terkait dan saling memberikan kontribusi dimana salah satu aspek dalam suatu pembangunan *system* berdasarkan UP *dicipline* meliputi *business modelling*, *requirements*, *design*, *implementation*, *testing*, *deployment*, *configuration* dan *change management*. Setiap iterasi dalam proses UP melibatkan semua kegiatan dari disiplin ilmu perancangan.

FIGURE 14-3

UP disciplines used in varying amounts in each iteration



Gambar 2.15 Contoh *Unified Process Life Cycle Model*

(Sumber: Satzinger, Jackson, dan Burd 2012:448)

2.17 Kerangka Pikir

Kerangka pikir dalam penyusunan penelitian ini dimulai dari fase perencanaan penulisan yang meliputi penentuan awal yaitu latar belakang, rumusan masalah, ruang lingkup, tujuan dan manfaat, metodologi penelitian dan sistematika penulisan yang menjelaskan inisiasi penulisan dalam penentuan topik yang akan dibahas berupa pendahuluan untuk mengidentifikasi topik pada PT Tuffiadi Semesta. Tahap kedua adalah fase identifikasi dan pembahasan analisis proses bisnis yang berjalan dimana terdapat pengumpulan dokumen yang terkait dalam topik pembelian, hutang dan pengeluaran kas serta menjelaskan profil perusahaan secara detail dan produk yang terkait, membahas prosedur sistem yang sedang berjalan seperti prosedur pembelian, prosedur hutang, dan prosedur pengeluaran kas berdasarkan identifikasi masalah yang terjadi pada sistem yang sedang berjalan.

Setelah itu membuat rekomendasi solusi berupa analisis dan perancangan baru berdasarkan permasalahan yang telah ditemukan. Tahap ketiga yaitu tahap analisis Sistem Informasi Akuntansi pembelian, hutang dan

pengeluaran kas Setelah itu melakukan perancangan dengan menggunakan metode *Object Oriented Analysis and Design* pendekatan Satzinger dengan pendekatan Modelling and requirement Dicipline yang terdiri dari 3 tahap, yaitu tahap *inception*, tahap *elaboration* dan tahap *construction*. Pada tahap *inception* yaitu dengan mengidentifikasi dan menganalisis Sistem Informasi Akuntansi pembelian, hutang dan pengeluaran kas berupa penjelasan secara deskriptif dan terstruktur berdasarkan rekomendasi solusi perbaikan dari tata sistem yang berjalan.

Pada tahap *elaboration* yaitu menggunakan diagram perancangan dan pembuatan *UML (Unified Modelling Language)* dimana perancangan ini menggunakan *Activity Diagram* yang diusulkan, *Use Case Diagram*, *Use Case Description*, *Domain Class Diagram*, *First Cut Class Diagram*, *Updated Design Class Diagram*, *State Machine Diagram*, *System Sequence Diagram*, *Multilayer Sequence Diagram*, *Package Diagram* dan *User Interface*.

Setelah tahap *inception* dan *elaboration* selesai dilakukan maka selanjut nya adalah tahap ketiga yaitu tahap *construction* dimana pada tahap ini adalah perancangan dan pembuatan aplikasi menggunakan bahasa pemrograman *C#* dengan aplikasi *Microsoft Visual Studio 2010* dan *My SQL 2008* sebagai perancangan *database* dimana pada tahap ini akan dilakukan uji coba dan konversi sistem baru. Selanjut nya membuat rencana implementasi dan setelah itu pada penulisan terakhir maka terdapat saran dan simpulan untuk perusahaan mengenai sistem yang telah diperbaiki dan dibangun. Di bawah ini adalah gambar 2.16 yang menjelaskan kerangka pikir:

