

2. LANDASAN TEORI

2.1. Sistem Informasi Akuntansi

Sistem adalah rangkaian dua komponen atau lebih yang saling berhubungan dan berinteraksi untuk mencapai tujuan. Sistem Informasi adalah cara teratur untuk mengumpulkan, memproses, mengelola, dan melaporkan informasi agar organisasi dapat mencapai tujuan dan sasarnya. Sistem Informasi Akuntansi adalah sumber daya manusia dan modal dalam organisasi, yang bertanggung jawab terhadap persiapan informasi keuangan dan informasi tersebut diperoleh dari mengumpulkan dan memproses berbagai transaksi perusahaan (Romney & Steinbart, 2003).

Sistem Informasi Akuntansi memiliki 5 siklus sub sistem, yaitu : siklus pendapatan (*revenue cycle*), siklus pengeluaran (*expenditure cycle*), siklus penggajian (*payroll cycle*), siklus produksi (*production cycle*), siklus keuangan (*financing cycle*). Kelima siklus di atas memberikan data transaksi pada buku besar dan pelaporan (*general ledger* dan *reporting systems*) untuk pencatatan dan komunikasi yang berhubungan dengan penyiapan laporan keuangan dan laporan manajerial lainnya (Romney & Steinbart, 2003).

2.1.1. Siklus Pendapatan

Siklus pendapatan (*revenue cycle*) adalah rangkaian aktivitas bisnis dan kegiatan pemrosesan informasi terkait yang terus berulang dengan menyediakan barang dan jasa ke para pelanggan dan menagih kas sebagai pembayaran dari penjualan-penjualan tersebut (Romney & Steinbart, 2003). Pada siklus pendapatan, terdapat 4 aktivitas dasar bisnis yaitu :

a. Memasukkan pesanan penjualan (*sales order entry*)

Siklus pendapatan dimulai dengan penerimaan pesanan dari para pelanggan. Departemen bagian pesanan penjualan, melakukan proses memasukkan pesanan penjualan. Dokumen yang dibuat dalam proses memasukkan pesanan penjualan adalah pesanan penjualan (*sales order*). Proses memasukkan pesanan penjualan mencakup tiga tahap: mengambil

pesanan dari pelanggan, memeriksa dan menyetujui kredit pelanggan, serta memeriksa ketersediaan persediaan.

b. Mengirim pesanan (*shipping*)

Aktivitas dasar kedua dalam siklus pendapatan adalah memenuhi pesanan pelanggan dan mengirimkan barang dagangan yang diinginkan tersebut. Proses ini terdiri dari dua tahap: mengambil dan mengepak pesanan, dan mengirim pesanan tersebut beserta dokumen pengiriman (surat jalan). Departemen bagian penggudangan dan pengiriman melakukan aktivitas ini.

c. Penagihan dan piutang usaha (*billing and accounts receivable*)

Aktivitas dasar ketiga dalam siklus pendapatan melibatkan penagihan ke para pelanggan dan memelihara data piutang usaha. Dokumen yang dibuat dalam proses penagihan adalah faktur penjualan (*sales invoice*).

d. Menerima pembayaran / kas (*cash collection*)

Langkah terakhir dari siklus pendapatan adalah menerima pembayaran. Yang melakukan aktivitas ini adalah kasir.

2.1.2. Siklus Pengeluaran

Siklus pengeluaran (*expenditure cycle*) adalah rangkaian kegiatan bisnis dan operasional pemrosesan data terkait yang berhubungan dengan pembelian serta pembayaran barang dan jasa (Romney & Steinbart, 2003). Pada siklus pengeluaran, terdapat 3 aktivitas dasar bisnis yaitu :

a. Memesan barang, persediaan, dan jasa.

Aktivitas pertama dalam siklus pengeluaran adalah memesan perediaan atau perlengkapan. Keputusan penting yang dibuat dalam langkah ini adalah mengidentifikasi apa, kapan, dan berapa banyak yang dibeli, dan dari pemasok mana akan dibeli. Dokumen yang dibuat dalam proses pemesanan barang adalah pesanan pembelian (*purchase order*).

b. Menerima dan menyimpan barang, persediaan, dan jasa.

Aktivitas kedua dalam siklus pengeluaran adalah penerimaan dan penyimpanan barang yang dipesan. Bagian penerimaan bertanggung jawab untuk mengecek dan menerima kiriman dari para pemasok. Dokumen

yang dibuat dalam proses penerimaan barang adalah laporan penerimaan (*receiving report*).

- c. Membayar untuk barang, persediaan, dan jasa.

Aktivitas ketiga dalam siklus pengeluaran adalah menyetujui faktur penjualan dari vendor untuk pembayaran. Bagian utang usaha menyetujui faktur penjualan untuk dibayar dan kasir bertanggung jawab untuk melakukan pembayaran.

2.1.3. Siklus Penggajian

Siklus penggajian (*payroll cycle*) adalah rangkaian aktivitas bisnis berulang dan operasional pemrosesan data terkait yang berhubungan dengan cara yang efektif dalam mengelola pegawai (Romney & Steinbart, 2003). Pada siklus penggajian, terdapat 7 aktivitas dasar bisnis yaitu:

- a. Perbaharui file induk penggajian

Aktivitas pertama dalam siklus penggajian melibatkan pembaharuan file induk penggajian untuk mencerminkan berbagai jenis perubahan penggajian: memperkerjakan orang baru, pemberhentian, perubahan tingkat gaji.

- b. Perbaharui tarif dan pemotongan pajak

Aktivitas kedua dalam siklus penggajian adalah memperbaharui informasi mengenai tarif dan pemotongan pajak lainnya. File yang terlibat dalam proses perbaharui tarif dan pemotongan pajak adalah file tarif.

- c. Validasi data waktu dan kehadiran

Langkah ketiga dalam siklus penggajian adalah memvalidasi setiap data waktu dan kehadiran pegawai. Data waktu dan kehadiran didapat dari file absensi.

- d. Mempersiapkan penggajian

Langkah keempat dalam siklus penggajian adalah mempersiapkan penggajian. Informasi tingkat gaji didapat dari file induk penggajian.

- e. Membayar gaji

Langkah selanjutnya adalah pembayaran yang sesungguhnya atas cek gaji ke pegawai.

- f. Hitung kompensasi dan pajak yang dibayar perusahaan
Perusahaan membayar beberapa pajak penghasilan dan kompensasi pegawai secara langsung.
- g. Keluarkan pajak penghasilan dan potongan lain-lain
Aktivitas terakhir dalam proses penggajian adalah membayar kewajiban pajak penghasilan dan potongan sukarela lainnya dari setiap pegawai.

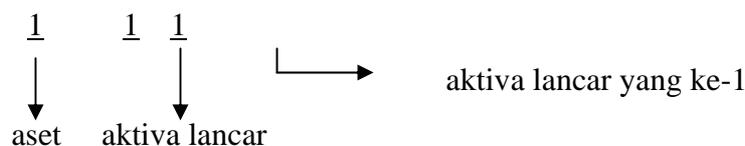
2.2. Sistem Akuntansi

Pada proses akuntansi terdapat 3 aktivitas, yaitu : mengidentifikasi, mencatat, dan mengkomunikasi kejadian ekonomi dari sebuah perusahaan. Proses pertama adalah identifikasi, yaitu kegiatan memilih kegiatan yang termasuk kegiatan ekonomi. Proses yang kedua adalah pencatatan, yaitu pencatatan semua kegiatan ekonomi yang terjadi pada perusahaan tersebut, tujuan dari pencatatan kegiatan ekonomi tersebut adalah untuk penyediaan sejarah dari kegiatan keuangan dari perusahaan. Proses yang ketiga adalah komunikasi, informasi yang telah didapat dari identifikasi dan pencatatan tidak akan berguna bila tidak dikomunikasikan. Informasi ini dikomunikasikan melalui persiapan dan distribusi dari laporan akuntansi, yang paling umum disebut sebagai laporan keuangan. (Weygandt, Kieso, & Kimmel, 1999)

2.2.1. Chart Of Account (COA)

Chart of Account adalah daftar rekening dan nomor rekening yang menunjukkan lokasi rekening tersebut dalam buku besar. Jumlah dan tipe rekening berbeda-beda untuk tiap perusahaan, tergantung pada besar, kompleksitas, dan tipe bisnis yang dilakukan. Umumnya rekening akuntansi dibedakan menjadi 5 kelompok besar, yaitu aset, hutang, modal, pendapatan, dan biaya.

Contoh penulisan nomor rekening : 111 → kas



2.2.2. Proses dan siklus akuntansi

Berikut ini merupakan urutan dari siklus akuntansi yang tidak pernah lepas dari Sistem Informasi Akuntansi (*Romney & Steinbart, 2003*), yaitu :

1. Analisis transaksi bisnis, seperti pengumpulan bukti-bukti transaksi yang terjadi.
2. Menjurnal transaksi-transaksi tersebut.
3. Mem-*posting* jurnal tersebut ke buku besar (*general ledger*).
4. Mempersiapkan neraca.
5. Menjurnal dan mem-*posting* penyesuaian (jurnal penyesuaian).
6. Mempersiapkan neraca penyesuaian
7. Mempersiapkan laporan keuangan, berupa laporan rugi laba, perubahan modal, dan neraca.
8. Menjurnal dan mem-*posting* penutup (jurnal penutup).
9. Mempersiapkan neraca penutup.

2.2.3. Jurnal

Jurnal dibedakan menjadi 2 macam, yaitu :

2.2.3.1. Jurnal Umum

Mencatat transaksi yang frekuensi terjadinya jarang seperti membayar pinjaman, penyesuaian di akhir periode, dan jurnal penutup. Jurnal Umum meliputi :

- Jurnal Penyesuaian

Jurnal penyesuaian digunakan untuk melakukan penyesuaian terhadap pendapatan dan biaya pada periode dimana seharusnya pendapatan dan biaya pada periode dimana seharusnya pendapatan itu didapatkan dan biaya ditanggungkan.

Contoh : Gaji pegawai selalu dibayarkan pada hari sabtu sebesar 70000, dan kebetulan pada akhir periode atau pada tanggal 31 Desember bukan merupakan hari sabtu tetapi hari selasa, maka jurnal penyesuaiannya adalah :

Beban gaji 20000
Utang gaji 20000

- Jurnal Penutup

Jurnal penutup digunakan untuk menutup rekening-rekening sementara yang didapatkan dalam suatu periode akuntansi. Jurnalnya adalah sebagai berikut :

Pendapatan

 Ikhtisar laba rugi

(digunakan untuk menutup *sales* atau penjualan)

Ikhtisar laba rugi

 Beban

 Beban

(digunakan untuk menutup beban-beban yang ada)

Ikhtisar laba rugi

 <Nama> Capital

(digunakan untuk menutup laba rugi jika perusahaan mengalami laba)

<Nama> Capital

 Iktisar laba rugi

(digunakan untuk menutup laba rugi jika perusahaan mengalami rugi)

<Nama> Capital

 <Nama> Prive

(digunakan untuk menutup Prive)

2.2.3.2. Jurnal Khusus

Mencatat transaksi yang frekuensi terjadinya sering, jurnal khusus menyederhanakan proses pencatatan transaksi yang terjadi berulang dalam jumlah yang besar. Jurnal khusus meliputi :

- Jurnal Penjualan (*Sales Journal*)

Jurnal penjualan digunakan untuk mencatat transaksi penjualan yang dilakukan secara kredit yang terjadi. Contoh jurnal penjualan dapat dilihat pada Gambar 2.1.

| Karn Wholesale Supply Sales Journal | | | | | |
|--|-----------------|-------------|------|------------------------------------|---|
| Date | Account Debited | Invoice No. | Ref. | Accts. Receivable Dr. Sales Cr. | Cost of Goods Sold Dr. Merchandise Inventory Cr. |
| 2005 | | | | | |
| May 3 | Abbot Sisters | 101 | | 10,600 | 6,360 |
| 7 | Babson Co. | 102 | | 11,350 | 7,370 |
| 14 | Carson Bros. | 103 | | 7,800 | 5,070 |
| 19 | Deli Co | 104 | | 9,300 | 6,510 |
| 21 | Abbot Sisters | 105 | | 15,400 | 10,780 |
| 24 | Deli Co. | 106 | | 21,210 | 15,900 |
| 27 | Babson Co. | 107 | | 14,570 | 10,200 |
| | | | | 90,230 | 62,190 |

Gambar 2.1. Contoh Jurnal Penjualan

Sumber: Weigandt, Kieso, Kimmel (2005, p. 285)

- Jurnal Pembelian (*Purchase Journal*)

Jurnal pembelian digunakan untuk mencatat transaksi pembelian yang dilakukan secara kredit yang terjadi. Contoh jurnal pembelian dapat dilihat pada Gambar 2.2.

| Karns Wholesale Supply PURCHASE JOURNAL | | | | |
|--|---------------------------|------------|-----|--|
| Date | Account Credited | Term | Ref | Merchandise Inventory Dr. Account Payable Cr. |
| 2005 | | | | |
| May 6 | Jesper Manufacturing Inc. | 2/10, n/30 | | 11,000 |
| 10 | Eaton Hower Inc. | 3/10, n/30 | | 7,200 |
| 14 | Fabor and Son | 1/10, n/30 | | 6,900 |
| 19 | Jesper Manufacturing Inc. | 2/10, n/30 | | 17,500 |

Gambar 2.2. Contoh Jurnal Pembelian

Sumber: Weigandt, Kieso, Kimmel (2005, p. 290)

- Jurnal Penerimaan Kas (*Cash Receipts Journal*)

Jurnal penerimaan kas digunakan untuk mencatat transaksi penjualan yang dilakukan secara tunai yang terjadi. Selain itu, jurnal penerimaan kas juga dapat dilakukan jika perusahaan mendapatkan pembayaran piutang usaha, pinjaman dari bank atau pihak tertentu dan jika terjadi transfer kas. Contoh jurnal penerimaan kas dapat dilihat pada Gambar 2.3.

| Karns Wholesale Supply CASH RECEIPTS JOURNAL | | | | | | | | |
|---|----------------------|------|----------|---------------------|------------------------|-----------|-------------------|---------------------------------------|
| Date | Account Credited | Ref. | Cash Dr. | Sales Discounts Dr. | Account Receivable Cr. | Sales Cr. | Other Account Cr. | Cost Of Goods Sold Dr. Mdse. Inv. Cr. |
| 2005 | | | | | | | | |
| May 1 | D. A. Karns. Capital | 301 | 5,000 | | | | 5,000 | |
| 7 | | | 1,900 | | | 1,900 | | 1,240 |
| 10 | Abbot Sisters | | 10,388 | 212 | 10,600 | | | |
| 12 | | | 2,600 | | | 2,600 | | 1,690 |
| 17 | Babson Co. | | 11,123 | 227 | 11,350 | | | |

Gambar 2.3. Contoh Jurnal Penerimaan Kas

Sumber: Weigandt, Kieso, Kimmel (2005, p. 287)

- Jurnal Pengeluaran Kas (*Cash Payments Journal*)

Jurnal pengeluaran kas digunakan untuk mencatat transaksi pembelian yang dilakukan secara tunai yang terjadi. Selain itu, jurnal pengeluaran kas juga dapat dilakukan jika perusahaan membayar utang usaha kepada *supplier* atau pihak tertentu dan jika terjadi pembayaran beban. Contoh jurnal pengeluaran kas dapat dilihat pada Gambar 2.4.

| Karns Wholesale Supply CASH PAYMENTS JOURNAL | | | | | | | |
|---|---------|-------------------|-----|--------------------|--------------------|---------------------------|----------|
| Date | Ck. No. | Account Debited | Ref | Other Accounts Dr. | Account Payable Dr | Merchandise Inventory Cr. | Cash Cr. |
| 2005 | | | | | | | |
| May 1 | 101 | Prepaid Insurance | 130 | 1,200 | | | 1,200 |
| 3 | 102 | Msde. Inventory | 120 | 100 | | | 100 |
| 8 | 103 | Msde. Inventory | 120 | 4,400 | | | 4,400 |
| 10 | 104 | Jasper Manuf. Inc | | | 11,000 | 220 | 10,780 |

Gambar 2.4. Contoh Jurnal Pengeluaran Kas

Sumber: Weigandt, Kieso, Kimmel (2005, p. 292)

2.2.3.3. Buku Besar (*General Ledger*)

Setiap jurnal yang sudah dibuat dipindahkan kedalam buku besar sesuai dengan kelompok rekeningnya. Contoh buku besar dapat dilihat pada Gambar 2.5.

| <i>Kas</i> | | | | | <i>No 101</i> |
|----------------|-------------------|------------|---------------|---------------|----------------|
| <i>Tanggal</i> | <i>Keterangan</i> | <i>Ref</i> | <i>Debit</i> | <i>Kredit</i> | <i>Balance</i> |
| <i>Juni 1</i> | | <i>J1</i> | <i>500000</i> | | <i>500000</i> |
| <i>31</i> | | | | <i>150000</i> | <i>350000</i> |

Gambar 2.5. Contoh Buku Besar

Sumber: Weigandt, Kieso, Kimmel (2005, p. 292)

2.2.3.4. Laporan Keuangan

Tahapan terakhir dari proses akuntansi adalah untuk mempersiapkan laporan keuangan. Macam-macam laporan keuangan :

1. Laporan laba rugi (*income statement*)

Laba rugi menyajikan pendapatan dan biaya serta hasil bersih pemasukan atau kerugian bersih dari toko untuk periode waktu tertentu.

2. Laporan perubahan modal (*statement of equity*)

Merangkum perubahan modal pemilik dalam periode waktu tertentu.

3. Neraca (*balance sheet*)

Neraca menyajikan asset, hutang, dan modal pemilik terhadap bisnisnya pada tanggal tertentu.

2.3. Sistem Inventory

Pada sistem *inventory* ini dapat diketahui apa saja yang tersedia untuk dijual dan apa yang telah terjual. Terdapat dua buah sistem *inventory*, yaitu:

2.3.1. Perpetual System

Dalam *perpetual system*, detail *record* biaya dari tiap-tiap *inventory* yang dibeli dan dijual dicatat. Sistem ini secara berkala menampilkan *inventory* yang seharusnya dimiliki untuk setiap *item*. Setiap ada penjualan dilakukan perhitungan pada *Cost of Good Sold* (COGS) / Harga Pokok Penjualan (HPP)

2.3.2. Periodik System

Dalam *periodik system* detail *record inventory* barang yang dimiliki tidak dicatat sebelum periode berakhir. Biaya dari barang yang terjual ditetapkan hanya

pada akhir dari periode akuntansi. Pada saat itu penghitungan *inventory* telah diambil untuk menetapkan biaya barang yang dimiliki. Setiap ada penjualan tidak dilakukan perhitungan pada HPP tetapi perhitungan HPP dilakukan pada akhir periode akun.

2.4. Metode Penghitungan Alur Biaya

Metode Penghitungan alur biaya terdiri dari 3 macam, yaitu : *First In First Out*, *Last In First Out*, dan *Average*. Penjelasan untuk masing – masing metode dalam dijelaskan pada sub bab berikut :

2.4.1. *First In First Out* (FIFO)

Metode FIFO mengasumsikan bahwa barang yang dibeli lebih awal adalah barang pertama yang dijual. Dengan metode FIFO biaya untuk pembelian barang yang dibeli lebih dahulu ditetapkan disebut sebagai harga pokok penjualan. (Catatan: penjelasan di atas tidak berarti barang yang pertama kali dibeli harus dijual pertama). Contoh FIFO dapat dilihat pada Gambar 2.6.

| Pool of Cost Cost of Goods Available for Sale | | | | |
|--|---------------------|-------|-----------|------------|
| Date | Explanation | Units | Unit Cost | Total Cost |
| 01/01 | Beginning Inventory | 100 | \$10 | \$1,000 |
| 04/15 | Purchase | 200 | 11 | 2,200 |
| 08/24 | Purchase | 300 | 12 | 3,600 |
| 11/27 | Purchase | 400 | 13 | 5,200 |
| | Total | 1,000 | | 12,000 |

| Step 1 Ending Inventory | | | | Step 2 Cost of Goods Sold | |
|----------------------------|------|-----------|------------|----------------------------------|--------|
| Date | Unit | Unit Cost | Total Cost | | |
| 11/27 | 400 | \$13 | \$5,200 | Cost of goods available for sale | 12,000 |
| 08/24 | 50 | 12 | 600 | Less: Ending Inventory | 5,800 |
| Total | 450 | | 5,600 | Cost of Goods Sold | 6200 |

Gambar 2.6. Contoh Metode FIFO

Sumber: Sumber: Weigandt, Kieso, Kimmel (2005, p. 236)

2.4.2. *Last In First Out* (LIFO)

Metode LIFO mengasumsikan bahwa barang yang dibeli paling akhir adalah barang pertama yang dijual. Dengan metode LIFO biaya untuk pembelian

barang yang dibeli paling akhir ditetapkan disebut sebagai harga pokok penjualan. (Catatan: penjelasan di atas tidak berarti barang yang terakhir dibeli harus dijual pertama). Contoh LIFO dapat dilihat pada Gambar 2.7.

| Pool of Cost Cost of Goods Available for Sale | | | | |
|--|---------------------|-------|-----------|------------|
| Date | Explanation | Units | Unit Cost | Total Cost |
| 1/1 | Beginning Inventory | 100 | 10 | 1000 |
| 4/15 | Purchase | 200 | 11 | 2200 |
| 8/24 | Purchase | 300 | 12 | 3600 |
| 11/27 | Purchase | 400 | 13 | 5200 |
| | Total | 1000 | | 12000 |

| Step 1 Ending Inventory | | | | Step 2 Cost of Goods Sold | |
|----------------------------|-------|-----------|------------|----------------------------------|------|
| Date | Units | Unit Cost | Total Cost | Cost of goods available for sale | |
| | | | | 12000 | |
| | | | | Less: Ending inventory | 5000 |
| | | | | Cost of Goods Sold | 7000 |
| 1/1 | 100 | 10 | 1000 | | |
| 4/15 | 200 | 11 | 2200 | | |
| 8/24 | 150 | 12 | 1800 | | |
| | 450 | | 5000 | | |

Gambar 2.7. Contoh Metode LIFO

Sumber: Weigandt, Kieso, Kimmel (2005, p. 237)

2.4.3. Average

Metode *average cost* mengasumsikan bahwa barang yang tersedia untuk dijual memiliki harga yang sama setiap unitnya. Dengan metode *average cost*, Harga Pokok Penjualan dihitung dari rata-rata harga pembelian. Contoh average dapat dilihat pada Gambar 2.8.

| | | | | |
|----------------------------------|---|-------------------------------|---|----------------------------|
| Cost of goods available for sale | + | Total Unit Available for Sale | = | Weighted-Average Unit Cost |
| 12000 | + | 1000 | = | 12 |

Gambar 2.8. Contoh Metode Average

Sumber: Weigandt, Kieso, Kimmel (2005, p. 238)

2.5. Sistem Flowchart

Berikut ini akan dijelaskan tentang pengertian dan macam – macam simbol yang digunakan di dalam *flowchart*.

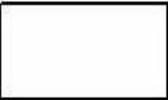
2.5.1. Definisi *Flowchart*

Flowchart adalah representasi grafik dari langkah-langkah yang harus diikuti dalam menyelesaikan suatu permasalahan yang terdiri atas sekumpulan simbol, dimana masing-masing simbol merepresentasikan suatu kegiatan tertentu. *Flowchart* diawali dengan penerimaan input, pemrosesan input, dan diakhiri dengan penampilan output.

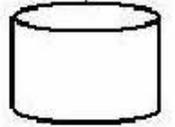
2.5.2. Simbol-Simbol *Flowchart*

Beberapa simbol yang digunakan di dalam *flowchart* yang berfungsi sebagai alat bantu dalam menjelaskan proses di dalam program dapat dilihat pada Tabel 2.1.

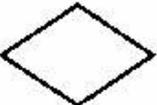
Tabel 2.1. Simbol-simbol *flowchart*

| Simbol | Nama Simbol | Keterangan |
|---|-------------------------|---|
|  | Dokumen | Dokumen atau laporan, dapat ditulis tangan atau dicetak dengan komputer. |
|  | Dokumen dengan tembusan | Dokumen yang sama dicetak dalam jumlah tertentu untuk kepentingan tertentu. |
|  | Input/output | Data yang dimasukkan ke proses atau data yang dihasilkan. |
|  | <i>Display</i> | Informasi yang ditampilkan pada sebuah alat seperti monitor atau layar. |
|  | Pengetikan online | Data yang dimasukkan melalui alat seperti keyboard dan barcode. |
|  | Proses komputer | Proses yang dilakukan dengan atau oleh komputer |

Tabel 2.1. Simbol-simbol *flowchart* (lanjutan)

| Simbol | Nama Simbol | Keterangan |
|---|--------------------------|---|
|  | Proses manual | Proses yang dilakukan secara manual |
|  | <i>Magnetic disk</i> | Data disimpan secara permanen di dalam <i>magnetic disk</i> , biasanya digunakan untuk file <i>master</i> . |
|  | Penyimpanan online | Data disimpan di dalam file online temporer melalui media yang dapat diakses secara |
|  | File | File dokumen secara manual disimpan dan diambil kembali. Huruf yang ada di dalam simbol menunjukkan urutan pengaturan file. |
|  | Arus dokumen atau proses | Arah pemrosesan atau arus dokumen. |
|  | On-page connector | Menghubungkan arus pemrosesan di satu halaman yang sama. Ini bertujuan untuk menghindari garis-garis saling bersilangan. |
|  | Off-page connector | Suatu penanda masuk dari, atau keluar ke, halaman yang lain. |
|  | Arus barang | Menunjukkan perpindahan barang secara fisik. |

Tabel 2.1. Simbol-simbol *flowchart* (lanjutan)

| Simbol | Nama Simbol | Keterangan |
|---|-------------|--|
|  | Terminal | Titik awal, akhir, atau pemberhentian dalam suatu proses atau program. |
|  | Keputusan | Langkah pengambilan keputusan (<i>if</i>). |

Sumber : Romney & Steinbart (2003, p. 198)

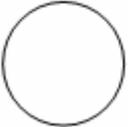
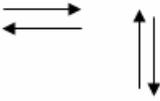
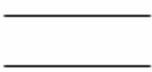
2.6. Data Flow Diagram (DFD)

Data Flow Diagram (DFD) merupakan suatu *graphic* yang menunjukkan hubungan antar proses dalam suatu *data flow*. *Data Flow Diagram* (DFD) menjelaskan aliran informasi dan transformasi data yang bergerak dari pemasukan data hingga keluar data. *Data Flow Diagram* (DFD) merupakan salah satu alat bantu yang digunakan dalam perancangan sistem. Tabel 2.2 menjelaskan simbol-simbol yang digunakan di dalam pembuatan DFD.

Tabel 2.2. Simbol-simbol DFD

| Notasi | Penjelasan |
|---|--|
|  | <i>External entity</i> adalah seseorang, sekelompok orang, sebuah departemen di dalam maupun di luar organisasi, atau sebuah sistem yang lain yang memberikan <i>input</i> untuk sistem yang ada atau menerima <i>output</i> dari sistem yang ada. <i>External entity</i> juga disebut <i>terminator</i> , karena merupakan batas dari sebuah sistem |

Tabel 2.2. Simbol-simbol DFD (lanjutan)

| Notasi | Penjelasan |
|---|--|
|  | <p>Proses adalah simbol yang mengubah suatu data dari suatu bentuk menjadi bentuk yang lain. Atau dengan kata lain, proses menerima <i>input</i> data dan mengeluarkan <i>output</i> data yang lain yang telah diproses. Dalam simbol DFD ini perlu diberikan keterangan tentang nama suatu proses yang ada didalamnya.</p> |
|  | <p>Arus data adalah aliran yang menunjukkan perpindahan data dari satu bagian ke bagian yang lain dalam sebuah sistem. Adapun data yang mengalir ini dapat berupa info. <i>Data flow</i> dalam DFD disimbolkan dengan tanda panah dan diberi nama atau keterangan di sampingnya yang menunjukkan data apa yang mengalir, yang hanya dapat dialirkan pada tanda panah adalah data atau informasi.</p> |
|  | <p>Penyimpan data adalah tempat penyimpanan data dalam suatu sistem, baik secara manual maupun secara elektronik. <i>Data store</i> digunakan jika suatu proses perlu menggunakan data tersebut lagi kemudian.</p> |

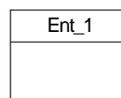
Sumber : McLeod, Raymond & Schell (2004, p. 430)

2.7. Entity Relationship Diagram (ERD)

Menurut Romney & Steinbart (2003), definisi *Entity Relationship Diagram* (ERD) adalah “A graphical technique for portraying a database schema” (p.115). Komponen – komponen utama dari ERD adalah :

- *Entity*

Merupakan suatu objek yang unik yang dapat dibedakan antara satu dengan yang lainnya. *Entity* juga mempunyai arti sebagai sebuah kumpulan konseptual data dari suatu data *field* setiap *entity* dilambangkan dengan suatu kotak dimana di dalam kotak tersebut tertulis nama dari *entity* tersebut. Ada dua macam *entity*, yaitu *Weak Entity* yaitu *entity* yang tidak memiliki *key attribute*, yaitu suatu *field* yang berisi kode-kode *unique* sebagai kata kunci, serta *Strong Entity* yaitu *entity* yang memiliki *key attribute*.



Gambar 2.9. Contoh gambar Entity

Sumber: Kroenke (2004, p. 37)

- *Attribute*

Merupakan bagian dari suatu *entity*, yang berisi keterangan mengenai *field-field* apa saja yang dimiliki oleh suatu *entity*, dimana setiap *attribute* biasanya berada pada kotak entity beserta penjelasan ukuran *size field* tersebut.



Gambar 2.10. Contoh Gambar Attribute

Sumber: Kroenke (2004, p. 37)

- *Key*

Merupakan suatu pengenal yang unik dari suatu *entity* antara satu dengan yang lainnya berbeda, biasanya ditulis dengan digaris bawahhi dalam suatu ERD. Ada beberapa macam *key*, yaitu : *Primary key*, *Foreign key*, *Partial key*, dan *Composite key*.

- *Relationship*

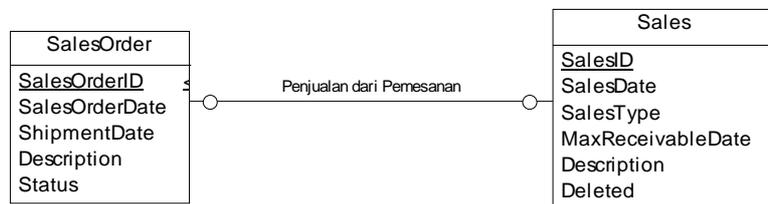
Merupakan suatu hubungan antar *entity* yang satu dengan yang lainnya atau dengan beberapa *entity* . Di dalam *relationship*, dikenal 3 istilah, yaitu :

1. *Cardinality*

Jumlah maksimum atau minimum dari elemen yang diizinkan pada setiap sisi dari suatu *relationship*. Terdapat 3 macam *cardinality* :

- *One-to-one*

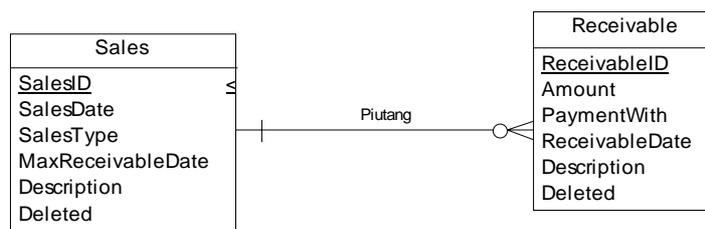
Entity 1 mempunyai hubungan minimal 0 dan maksimal 1 terhadap *entity* 2 begitu pula sebaliknya.



Gambar 2.11. Contoh gambar relasi One-to-one

- *One-to-many*

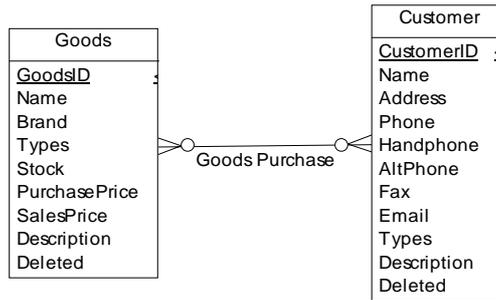
Entity 1 mempunyai hubungan minimal 0 dan maksimal banyak terhadap *entity* 2, *entity* 2 mempunyai hubungan minimal 0 dan maksimal 1 terhadap *entity* 1.



Gambar 2.12. Contoh gambar relasi One-to-many

- *Many-to-many*

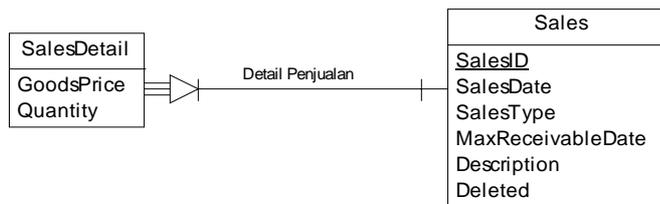
Entity 1 mempunyai hubungan minimal 0 dan maksimal banyak terhadap *entity* 2 begitu pula sebaliknya.



Gambar 2.13. Contoh gambar relasi Many-to-many

2. Dependency

Ketergantungan sebuah *entity* dengan *entity* induknya. Sehingga jika *entity* induknya dihapus, maka *entity* anaknya akan ikut terhapus secara otomatis.



Gambar 2.14. Contoh gambar Dependency

3. Mandatory

Menandai apakah semua *record* dari sebuah *entity* harus berelasi dengan *record* dari *entity* yang lain.

2.8. Pengertian SQL

Structured Query Language (SQL) adalah bahasa *American National Standard Input (ANSI)* yang digunakan untuk melakukan *query* data pada *database*

2.8.1. Kegunaan SQL

Secara umum *SQL* digunakan untuk :

- Memproses data lebih sebagai kumpulan daripada sebagai unit individual
- Menyediakan fungsi navigasi dari satu data ke dalam data lain.

- Menggunakan *statement* atau perintah yang dapat digabung atau berjalan sendiri-sendiri.

Secara mendasar *SQL* dapat mengoperasikan data pada level logika. *SQL* digunakan untuk melakukan implementasi detail seperti misalnya manipulasi data, sebagai contoh misalnya, untuk mengambil sekumpulan data berupa kumpulan baris dari suatu tabel, tinggal mendefinisikan kriteria yang dibutuhkan.

Semua perintah *SQL* menggunakan *optimizer*, yaitu suatu bagian dalam bahasa yang berfungsi untuk menetapkan cara yang paling efisien ketika mengakses data (McEwan, Bennet & Solomon, 2000). Secara umum bahasa *SQL* menyediakan fasilitas untuk melakukan *setting* agar *optimizer* dapat melakukannya dengan baik. Perintah-perintah yang ada dalam *SQL*:

- *Query* data.
- *Inserting*, *Updating* dan *deleting* baris data dalam tabel.
- Untuk modifikasi objek juga tersedia *creating*, *replacing*, *altering*, dan *dropping* objek.
- Perintah untuk pengontrolan akses ke *database* dan objeknya.
- Perintah untuk menjamin konsistensi dan integritas *database*.

2.8.2. Komponen SQL

Komponen *SQL* terdiri dari tiga bagian yaitu komponen untuk pendefinisian data (*DDL*), komponen manipulasi data (*DML*) dan komponen pengontrolan data (*DCL*).

2.8.2.1. DDL - Data Definition Language

Pada *DDL* hal terpenting adalah pembentukan *database* dengan objek utama tabel, indeks, membuat *domain*. *Domain* dalam hal ini adalah kemampuan *database* untuk membentuk suatu tipe baru yang pada dasarnya mengacu pada tipe dasar.

Beberapa hal yang dapat dilakukan *DDL* yaitu:

- *Creating tables* : untuk membuat tabel.
- *Creating indices* : untuk membuat indeks.
- *Altering tables* : untuk mengubah tabel.
- *Dropping tables* : untuk menghapus tabel.
- *Creating domain* : untuk membuat tipe data baru.

2.8.2.2. DML - Data Manipulation Language

Pada *DML* yang terpenting adalah melakukan operasi atau manipulasi terhadap *database*

Beberapa hal yang dilakukan *DML* yaitu:

- *Inserting* : penyisipan data.
- *Updating* : pengubahan data.
- *Deletion* : penghapusan data.
- *Creating view* : membuat tabel maya.

2.8.2.3. DCL – Data Control Language

Pada *DCL* hal terpenting adalah memberi otoritas, konfirmasi penyimpanan *database*, menunda perubahan, dan menghapus otoritas yang telah diberikan. Pada beberapa *database* juga dapat diatur *database space per user*.

2.9. SQL Server

SQL Server adalah sistem manajemen *database* relational (*RDBMS*) yang dirancang untuk aplikasi dengan arsitektur *client/server* (Marcus, Teddy, Priyono, A&Widhiadi, 2004), istilah *client*, *server* dan *client/server* dapat digunakan untuk merujuk kepada konsep yang sangat umum atau hal spesifik dari perangkat keras atau perangkat lunak. Pada level yang sangat umum, sebuah *client* adalah setiap komponen dari sebuah sistem yang meminta layanan atau sumber daya (*resources*) dari komponen sistem lainnya. Sedangkan sebuah *server* adalah setiap komponen sistem yang menyediakan layanan atau sumber daya ke komponen lainnya. *SQL Server* juga dapat diartikan sebagai *database server* yang fungsional yang menyediakan layanan *database*. *Database* dalam *SQL Server* tidak hanya

mempunyai data mentah, tetapi juga mempunyai struktur *database*, *index*, keamanan *database*, dan objek lain seperti *view* (tampilan) dan prosedur yang disimpan pada *database* yang penting.

2.10. Pengertian tabel pada *database*

Tabel adalah alat bantu untuk mengatur atau mengelompokkan data mengenai subyek yang sama dan mengandung informasi dari kolom dan baris. Tabel- tabel saling berhubungan dengan *database* pada saat dibutuhkan, tabel juga mempunyai tipe – tipe data dan data yang aktual. Beberapa hal penting yang harus dimiliki oleh suatu tabel :

- Nama tabel. (harus unik, sehingga dapat membedakan antara 1 tabel dengan tabel lainnya)
- Deskripsi kolom (*field*), nama kolom, *domain* kolom(jenis / *type*), panjang, dan *range*
- *Referential Integrity Constraint* :
 - Definisi apakah kolom tersebut termasuk *primary key*.
 - Hubungan *foreign key* pada tabel dengan *primary key* dari tabel lain.