

Bab 2

Tinjauan Pustaka

2.1 Sistem

Sistem adalah sekelompok komponen terkait yang berkolaborasi untuk mencapai tujuan dengan menerima *input* dan produksi *output* dalam transformasi yang reguler [4, p. 49]. Konsep umum yang sering digunakan di berbagai bidang dalam mendefinisikan sistem adalah proses mengubah *input* menjadi *output*. Konsep ini adalah konsep sederhana pada sistem yang digunakan oleh banyak bidang dalam mendefinisikan sistem. Hal ini berarti bahwa sistem terdiri dari tiga komponen dasar yaitu *input*, proses (transformasi), dan *output*. Pada pengembangannya atau dalam kata lain definisi secara khusus makna sistem disesuaikan atau diperluas sesuai cakupan kebutuhan tiap bidang. Contohnya yaitu suatu sistem yang ditambahkan analisis dan evaluasi setelah menghasilkan *output*. [5, p. 12]

2.2 Informasi

Informasi dan data merupakan hal yang berbeda meskipun keduanya sering digunakan secara berkaitan. Informasi merupakan data-data berguna dan informatif yang sudah diolah sehingga mampu dijadikan sebagai acuan atau dasar untuk mengambil suatu keputusan yang tepat. Informasi ini sangat penting bagi organisasi. Informasi merupakan salah satu sumber daya bagi organisasi atau bisnis (perusahaan) sumber daya lainnya seperti peralatan, material, sumber daya manusia, finansial, dan energi. Pada sistem informasi akuntansi, Kesuksesan dari sebuah sistem sangat dipengaruhi oleh kualitas dari informasi yang dikumpulkan (didapatkan). Seluruh sistem organisasional dapat mencapai tujuannya secara konseptual dengan melalui suatu proses alokasi sumber daya, yang kemudian diwujudkan dengan suatu proses pengambilan keputusan manajerial. Informasi juga memiliki nilai ekonomi saat informasi ini dapat mendukung suatu keputusan alokasi dari sumber daya, sehingga dapat dikatakan bahwa informasi tersebut mendukung sistem untuk mencapai tujuannya. [6, pp. 5–6]

Suatu informasi yang berkualitas dapat diklasifikasikan berdasarkan 4 dimensi informasi yaitu, relevansi, akurasi, ketepatan waktu, dan kelengkapan dari informasi tersebut. [7, p. 65] Selain itu, informasi adalah syarat utama untuk meningkatkan kemakmuran suatu perusahaan atau organisasi. Menurut penjelasan di atas, dapat diketahui bahwa alasan utama untuk mengolah data menjadi sebuah informasi adalah sebagai berikut [8, p. 15]:

1. Membantu proses atau prosedur pengambilan keputusan
2. Diklasifikasikan sebagai elemen terkuat dalam suatu perusahaan atau organisasi
3. Memberikan nilai kompetisi
4. Meningkatkan ilustrasi struktural
5. Meminimalkan resiko kesalahan dalam suatu perusahaan atau organisasi
6. Memperbarui dan mengembangkan tingkat pengetahuan

2.3 Akuntansi

Akuntansi merupakan mengelola semua informasi keuangan yang ada, menciptakan lembar untuk perencanaan strategis, dan mendistribusikan laporan keuangan untuk membantu manajer dalam menentukan keputusan khususnya dalam hal keuangan. Fungsi keuangan bertanggung jawab atas pengelolaan aset keuangan perusahaan, seperti uang tunai, saham, dan investasi lainnya. Hal ini berarti bahwa akuntansi bertanggung jawab untuk memelihara dan mengelola keuangan perusahaan seperti kwitansi, penggajian, dan sebagainya. [9, p. 1284]

Tujuan akuntansi adalah untuk menyediakan informasi tentang entitas yang bermanfaat bagi pemegang saham dan kreditor dalam melakukan investasi dan keputusan kredit. Bidang akuntansi mencakup akuntansi keuangan, akuntansi manajerial, dan perpajakan. Jika dirincikan bidang-bidang itu seperti penggajian, piutang dagang, utang dagang, inventaris, dan penganggaran [10, p. 2]. Memang sulit untuk memikirkan tugas akuntansi yang tidak terintegrasi, dalam beberapa cara, dengan sistem informasi akuntansi. Tantangan bagi akuntan adalah menentukan cara terbaik untuk memberikan informasi yang diperlukan untuk mendukung bisnis dan proses pemerintah.

Peranan akuntansi khususnya akuntansi biaya dapat membantu manajemen suatu usaha dalam beberapa hal yaitu: [11, pp. 195–196]

1. Menyusun serta merencanakan anggaran dana untuk setiap aktivitas operasional perusahaan
2. Memperoleh metode dari perhitungan biaya yang memungkinkan mengatur dan mengendalikan aktivitas serta meminimasi biaya sehingga dapat mengendalikan kualitas
3. Mengendalikan seluruh persediaan bahan baku maupun inventaris, serta menentukan harga dari setiap produk yang dihasilkan
4. Menghitung serta menentukan biaya dan laba yang diperoleh perusahaan dalam suatu periode akuntansi
5. Menentukan alternatif dalam suatu jangka pendek maupun panjang yang dapat mempengaruhi pendapatan dan biaya.

2.4 Sistem Informasi Akuntansi

Setiap perusahaan selalu ingin tumbuh dan bergerak maju. Tetapi pertumbuhan perusahaan akan lebih baik jika didukung oleh sistem informasi yang dapat mendukung kompleksitas kegiatan perusahaan. Sistem informasi adalah lingkungan yang terintegrasi dari perangkat keras, perangkat lunak, orang-orang yang terutama berfungsi untuk tujuan mengumpulkan dan mengolah data menjadi berharga informasi dengan menerapkan daftar prosedur pada data pengumpulan, dengan demikian informasi berasal dari data oleh prosedur SI, dan membersihkan perbedaan antara data dan informasi [8, p. 15].

Sedangkan sistem informasi akuntansi umumnya merupakan teknik berbasis komputer untuk mencari perkembangan akuntansi dan kombinasinya dengan sumber daya lainnya. sistem informasi akuntansi bertanggung jawab atas pengumpulan, pemrosesan, penyimpanan, dan penyampaian informasi akuntansi atau keuangan yang digunakan dalam pengambilan keputusan administrasi yang terdiri dari transaksi secara non finansial secara khusus mempengaruhi transaksi finansial [12, p. 16].

Sistem informasi akuntansi memiliki fungsi penting dalam organisasi seperti dari pengumpulan dan pencatatan data informasi hingga pada pengelolaan, pemrosesan, dan mengkomunikasikan informasinya kepada pihak-pihak terkait baik eksternal maupun internal. Selain itu sistem informasi akuntansi yang efisien juga memiliki beberapa karakteristik dimana yang paling menonjol adalah kesederhanaan penggunaan, ketepatangunaan, keandalan, serta fleksibilitas sistem. Seiring perkembangan teknologi informasi, proses dan prosedur akuntansi juga akan berubah ketika data informasinya tersedia secara elektronik sehingga metode akuntansi yang akan digunakan juga akan berbeda. [1, p. 202]

Sistem informasi akuntansi tidak hanya mendukung proses bisnis akuntansi dan keuangan. Sistem ini juga sering membuat informasi berguna bagi pekerja non akuntan misalnya, individu yang bekerja dalam pemasaran, produksi, atau hubungan manusia. Agar informasi ini efektif, sebaiknya individu yang bekerja dalam sistem ini harus dapat membantu pengembangan sistem informasi akuntansi untuk mengidentifikasi Informasi apa saja yang mereka butuhkan dalam hal perencanaan, pengambilan keputusan, dan kontrol mereka. [6, p. 5]

2.4.1 Tujuan Sistem Informasi Akuntansi

Sistem informasi akuntansi bertujuan untuk memberi informasi akuntansi yang menguntungkan bagi penggunanya juga mencapai tujuan umum yang terdiri dari beberapa hal yang penting seperti [3, p. 147]:

1. Mengukur peristiwa-peristiwa ekonomi yang terjadi di dalam instansi melalui proses pengumpulan, pencatatan, dan meringkasnya dalam daftar akuntansi
2. Memberi informasi akuntansi melalui berbagai rangkaian dokumen serta laporan-an-nas ada pihak-pihak yang dapat memperoleh manfaatnya yang diantaranya merupakan administrasi instansi yang menggunakan informasi ini untuk evaluasi kinerja serta membuat keputusan yang tepat.

3. Mencapai Pengendalian internal atas semua elemen atau komponen material yang ada dalam institusi

Selain itu, berikut ini merupakan keunggulan dari menggunakan sistem informasi akuntansi [13, p. 19]:

1. Perusahaan yang lebih baik, hal ini dikarenakan pengguna dapat menentukan dan memastikan operasi khususnya dalam akuntansi perusahaan berjalan normal dimana kegiatan perusahaan dibagi sesuai fungsi dan implementasinya.
2. dapat memenuhi kebutuhan multifungsi dimana penggunaan objek akuntansi menjadi ekspansi setingkat level manajemen Secara umum, sistem informasi akuntansi dapat menghasilkan laporan keuangan ke bagian eksekutif keuangan dan beberapa informasi lainnya
3. Dapat mengontrol sistem secara lebih mudah baik sistem yang bersifat manual maupun sistem informasi akuntansi yang telah berbasis komputer.

Sistem informasi akuntansi dapat didefinisikan sebagai sistem saraf dari suatu organisasi. Jika mereka tidak berfungsi dengan baik, organisasi akan lumpuh dan secara bertahap bubar. Hal ini didukung oleh studi yang membahas dampak sistem informasi pada perusahaan jasa di Yordania dengan hasil berikut: [3, p. 156]

1. Sistem informasi akuntansi yang digunakan di perusahaan jasa Yordania cocok untuk mencapai kualitas data keuangan serta mengkonfirmasi keakuratan dan bahwa sektor jasa di Yordania sedang berkembang.
2. Kualitas data keuangan dicapai dengan menjaga keamanan sistem informasi akuntansi yang digunakan untuk menyimpan data, karena untuk lingkungan sistem informasi yang lebih terjamin, semakin tinggi tingkat kualitas data keuangan, karena melindungi dari kehilangan atau perubahan.

3. Tingkat kualitas data keuangan yang diandalkan oleh perusahaan jasa Yordania tinggi.
4. Ada perbedaan signifikan secara statistik pada tingkat ($\alpha \leq 0,05$) di antara perusahaan jasa Yordania dalam hal sifat sistem informasi akuntansi, *input* pada sistem informasi akuntansi dan keamanan sistem informasi akuntansi yang dikaitkan dengan sektor dimana perusahaan berada. Perbedaan-perbedaan ini cenderung menguntungkan sektor (hotel dan pariwisata). Ini mungkin menunjukkan perbedaan dalam sistem informasi akuntansi yang diterapkan di setiap sektor jasa.

2.4.2 Komponen Sistem Informasi Akuntansi

Sistem informasi akuntansi sama seperti sistem apapun yang terdiri dari berbagai komponen untuk mencapai tujuannya. Pada umumnya sistem informasi akuntansi terdiri dari komponen-komponen berikut [14, pp. 1432–1434]

1. Manusia (Unsur Manusia)

Orang-orang dalam sistem informasi akuntansi adalah pengguna sistem. Profesional yang mungkin memerlukan penggunaan sistem informasi akuntansi organisasi melibatkan akuntan, analis bisnis, konsultan, manajer, auditor, dan kepala petugas keuangan. Sistem informasi akuntansi juga memfasilitasi memperoleh informasi kepada pengguna luar organisasi bila diinginkan. Untuk Misalnya, konsultan dapat menggunakan informasi dalam sistem informasi akuntansi untuk menganalisis kinerja efektif dari struktur harga perusahaan dengan melihat data penjualan, data biaya, dan pendapatan. sistem informasi akuntansi harus dirancang untuk memenuhi persyaratan orang-orang yang akan memanfaatkannya. Selanjutnya, sistem harus halus dan mudah digunakan dan harus meningkatkan efisiensi dan tidak menghalangi itu.

2. Prosedur dan Instruksi

Prosedur dan instruksi sistem informasi akuntansi adalah teknik dan metode yang digunakan untuk mengumpulkan, menyimpan dan menyimpan, mengambil dan memproses data. Teknik-teknik otomatis dan manual ini akan menjadi keduanya, dan data dapat berasal dari sumber dalam (misalkan karyawan) dan sumber luar (misalkan pesanan pelanggan online). Persyaratan prosedur dan instruksi dipatuhi secara konsisten agar memadai untuk memenuhi permintaan dan kebutuhan pengguna yang berbeda dan berbagai jenis informasi.

3. Data

Data yang termasuk dalam sistem informasi akuntansi adalah semua informasi keuangan yang terkait dengan praktik bisnis organisasi. Apapun data bisnis yang mempengaruhi keuangan perusahaan itu harus masuk ke dalam sistem informasi akuntansi. Data termasuk dalam informasi akuntansi sistem berdasarkan sifat bisnis. Data ini kemudian dapat digunakan untuk menyiapkan laporan dan laporan akuntansi seperti tabel penyusutan atau amortisasi, umur piutang, neraca saldo percobaan, laba rugi, dan sebagainya. Keberadaan semua data ini dalam satu posisi dalam sistem informasi akuntansi membantu dan mendukung pencatatan, pelaporan, analisis, audit, dan kegiatan pengambilan keputusan, dari bisnis. Agar data bermanfaat, harus lengkap, benar, dan relevan.

4. Perangkat lunak

Komponen perangkat lunak pada sistem informasi akuntansi adalah komputer perangkat lunak yang digunakan untuk menyimpan, mengambil, menyimpan, memproses, dan menganalisis data keuangan perusahaan. Sebelum komputer ada, sistem informasi akuntansi dioperasikan dengan tangan, sistem berbasis kertas, tetapi, sebagian besar perusahaan menerapkan perangkat lunak komputer sebagai dasar pada sistem informasi. Keandalan, kualitas, dan keamanan adalah komponen mendasar dari perangkat lunak yang berguna untuk sistem informasi akuntansi. Manajer mengandalkan informasi yang dihasilkan oleh sistem untuk membuat resolusi perusahaan yang sesuai, dan mereka menginginkan informasi berkualitas tinggi untuk melakukan keputusan yang tepat ini.

5. Infrastruktur Teknologi Informasi

Komponen ini hanya nama yang elegan untuk peralatan dan perangkat keras yang digunakan mengoperasikan sistem informasi akuntansi. Sebagian besar perangkat keras ini dan item peralatan adalah alat yang dibutuhkan bisnis untuk memilikinya; karenanya mereka termasuk komputer pribadi, perangkat penyimpanan, printer, server, router, dan mungkin daya cadangan. Selain biaya komponen ini harus dipertimbangkan apakah mereka termasuk kombinasi faktor termasuk kapasitas penyimpanan, kecepatan, dan ukuran memori dan apakah mereka scalable dan dapat ditingkatkan. Mungkin sebagian besar penting, perangkat keras yang dipilih untuk sistem informasi akuntansi harus kompatibel dengan perangkat lunak yang dimaksud. Selain itu, sistem informasi akuntansi yang sempurna harus mencakup: skema untuk menjaga, melayani, meningkatkan dan mengganti komponen sistem perangkat keras, serta rencana untuk menyingkirkan tidak efisien dan perangkat keras yang usang sehingga data sensitif sepenuhnya hancur.

6. Kontrol Internal

komponen pada sistem informasi akuntansi ini adalah perlindungan dan keamanan prosedur serta tindakan yang termasuk untuk melindungi data penting. Ini bisa jadi rumit seperti identifikasi biometrik atau semudah kata sandi. Jadi, sistem informasi akuntansi harus terdiri dari kontrol internal untuk melindungi terhadap ilegal dan akses ilegal komputer dan untuk membatasi akses ke pengguna yang berwenang dalam organisasi. Demikian juga, itu juga harus mencegah akses file yang tidak sah oleh pengguna yang berwenang untuk masuk hanya ke bagian spesifik pada sistem. Suatu sistem informasi akuntansi melibatkan informasi rahasia yang tidak hanya berkaitan dengan perusahaan tetapi lebih jauh dengan pelanggan dan karyawan tersebut. Data ini dapat mencakup nomor kartu kredit, nomor ID Nasional, nomor jaminan sosial, dan sebagainya. Seluruh data tersebut harus dikriptografis, dan perlu diakses ke sistem harus login dan dimonitor. Juga, aktivitas sistem harus jelas dan dapat dilacak.

Komponen-komponen sistem informasi diatas memiliki fungsi tertentu seperti fungsi-fungsi berikut [3, pp. 146–147]:

1. Pengumpulan data

Unit ini mengumpulkan data-data yang diperoleh dari lingkungan institusi yang terkait. data ini dapat diwakili dalam peristiwa dan fakta bahwa para akuntan menganggapnya penting dan harus disimpan dan dicatat. Sifat dari keluarannya (*output*) memiliki dampak besar terhadap jenis-jenis data yang akan dikumpulkan dan dicatat ke dalam sistem.

2. Pengoperasian atau pemrosesan data

Data yang telah dikumpulkan dalam sistem informasi akuntansi tersebut dapat segera digunakan jika bermanfaat bagi pembuat keputusan ketika data tersebut dikumpulkan. Tetapi dalam kebanyakan kasus, data-data ini perlu diperhatikan atau diproses agar menjadi informasi yang dapat berguna dalam proses pengambilan keputusan sehingga nantinya data ini dapat dikirim ke Unit penyimpanan dan pengambilan data dalam sistem informasi akuntansi.

3. penyimpanan dan pengambilan data

Unit ini bertanggung jawab dalam menyimpan data Seandainya data tersebut tidak langsung digunakan untuk diproses atau dioperasikan. Hal ini berarti bahwa data tersebut memungkinkan untuk digunakan di masa depan atau untuk diproses sebelum dikirimkan kepada pengambil keputusan.

4. Pengiriman informasi

Unit ini merupakan sarana untuk mengirimkan atau memberikan data dan informasi dari satu unit ke Unit lain dalam sistem informasi akuntansi sehingga pada akhirnya menuju pembuat keputusan administratif. Saluran komunikasi sebagai pengiriman informasi ini saluran yang otomatis atau manual sesuai dengan sarana yang tersedia pada institusi terkait.

Saat memilih komponen dari perangkat keras maupun perangkat lunak untuk sistem akuntansi, biaya dan pengeluaran komponen yang dikeluarkan harus lebih kecil daripada manfaat yang dihasilkan dari menggunakan sistem tersebut sehingga anggota audit harus memiliki pengetahuan mengenai an-nur angkat lunak pendukung, alat yang digunakan, an-nasr ta hal lain yang terdapat dalam sistem informasi akuntansi tersebut. Tim audit harus dapat terbiasa sa dengan otomatisasi dimana pengendalian otomatis ini juga memiliki dampak terhadap pekerja pekerja akuntan di masa depan baik dalam hal metode kerja, pencatatan data, sistem dan jaringan baru, serta metode audit yang digunakan nantinya.

2.4.3 Peran Sistem Informasi Akuntansi dalam Suatu Organisasi

Pada awalnya, akuntansi dalam sistem informasi yaitu proses komunikatif yang mengumpulkan, menyimpan, memproses, dan mendistribusikan informasi kepada mereka yang membutuhkannya. Tetapi pengguna informasi akuntansi terkadang mengkritik sistem informasi akuntansi karena hanya menangkap dan melaporkan transaksi keuangan. Mereka mengklaim bahwa laporan keuangan sering mengabaikan beberapa kegiatan paling penting yang mempengaruhi entitas bisnis. Akan tetapi, sekarang ini sistem informasi akuntansi memusatkan perhatian pada data dan informasi non-keuangan serta keuangan. Hal ini sesuai dengan perspektif kontemporer bahwa sistem akuntansi bukan hanya sistem keuangan. [6, p. 9]

Tugas mendasar dari sebuah sistem informasi akuntansi adalah untuk mengotomatisasi tugas rutin memasukkan dan memposting transaksi akuntansi. Informasi ini diatur dalam format elektronik sehingga menghasilkan laporan keuangan dan bisa diakses segera untuk membantu dalam manajemen perusahaan. Manajemen sistem menyediakan informasi keuangan untuk semua manajer yang terkait dalam suatu organisasi. Keputusan biasanya didasarkan pada informasi yang dihasilkan pada sistem akuntansi. [9, p. 1284]

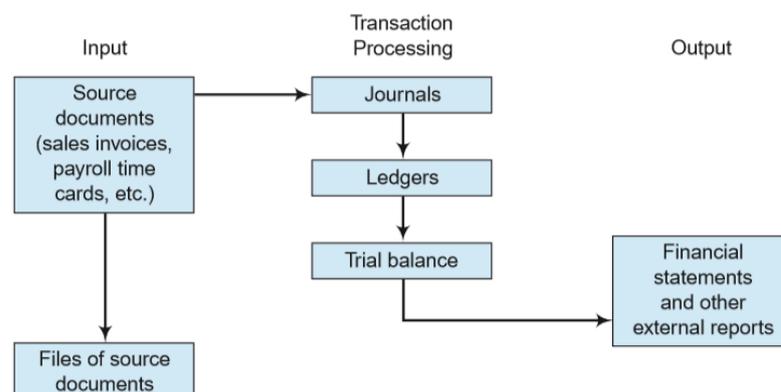
Sistem informasi akuntansi dapat menjadi salah satu sistem yang paling kritis untuk mengumpulkan, menyimpan, memproses, dan mendistribusikan informasi. maka dari itu sistem informasi akuntansi harus fokus pada faktor-faktor penting tersebut

untuk meminimalkan kesalahan yang dapat terjadi. Pada sistem informasi akuntansi, terdapat beberapa peran dari para pemangku kepentingan yang harus diidentifikasi diantaranya [13, p. 18]:

1. Prosedur data adalah mereka yang mengumpulkan atau membuat data untuk sistem informasi akuntansi
2. Pengelola data merupakan mereka yang mendesain mengembangkan serta mengoperasikan sistem informasi akuntansi
3. Konsumen data adalah mereka yang menggunakan informasi yang dihasilkan pada sistem dalam kegiatan kerja mereka
4. Manajer data adalah mereka yang bertanggung jawab untuk mengelola seluruh kualitas data yang ada dalam sistem informasi akuntansi

2.5 Sistem Informasi Akuntansi Keuangan

Objek utama pada sistem informasi akuntansi adalah untuk memberi info yang relevan pada individu dan kelompok baik internal maupun eksternal organisasi seperti investor, perpajakan, kreditor, dan sebagainya. Para akuntan mencapai informasi ini dengan menyiapkan berbagai laporan keuangan seperti laporan laba rugi, laporan penjualan, laporan arus kas, neraca saldo, dan sebagainya. Pada pihak internal biasanya digunakan oleh manajer untuk melakukan perencanaan, pengambilan keputusan, dan kegiatan pengendalian. Kegiatan pada sistem informasi akuntansi keuangan biasanya dimulai dengan transaksi dan berakhir dengan laporan keuangan periodiknya seperti pada gambar 2.1. [6, p. 15]



Gambar 2.1 Kegiatan sistem informasi akuntansi keuangan

Input dasar dan keluaran pada sistem akuntansi keuangan tradisional biasanya diekspresikan di unit-unit moneter. Disket dapat mengalami kesulitan jika sistem informasi akuntansi mengabaikan informasi non-moneter yang juga penting bagi pengguna. Sebagai contoh, seorang investor mungkin ingin tahu apa prospek untuk penjualan masa depan sebuah perusahaan, tetapi banyak sistem informasi akuntansi keuangan tidak mencatat informasi seperti penjualan pelanggan yang tidak diisi karena penjualan seperti itu bukan peristiwa keuangan yang dapat dikenali meskipun itu adalah peristiwa keuangan yang penting. [6, p. 15]

Dampak dari perkembangan teknologi informasi pada akuntansi keuangan menyangkut waktu *input*, pemrosesan, dan *output*. Laporan keuangan bersifat berkala dan sebagian besar perusahaan besar menerbitkannya secara triwulanan, dengan laporan komprehensif dihasilkan setiap tahun. Berdasarkan perkembangan teknologi informasi yang memungkinkan transaksi dilakukan dengan segera, akuntan dan bahkan sistem informasi akuntansi sendiri dapat menghasilkan laporan keuangan hampir secara *real-time*. [6, p. 16] Berikut ini merupakan langkah-langkah dari siklus transaksi akuntansi dan pencatatannya dalam jurnal: [6, p. 220]

1. Catat transaksi dalam jurnal.
2. Posting entri jurnal ke buku besar.
3. Siapkan saldo percobaan yang belum disesuaikan.
4. Rekam dan posting entri jurnal penyesuaian.
5. Mempersiapkan saldo percobaan yang disesuaikan.
6. Menyiapkan laporan keuangan.
7. Rekam dan posting entri jurnal penutup.
8. Siapkan saldo uji coba pasca-penutupan.

2.5.1 Pembukuan

Berdasarkan pada persiapan dokumen sumber, sistem informasi akuntansi mencatat setiap transaksi atau peristiwa bisnis yang mempengaruhi kondisi keuangan organisasi dalam pembukuan (jurnal, buku besar, dan neraca saldo). Berikut ini merupakan jenis-jenis pembukuan yang biasa digunakan dalam akuntansi [6, pp. 218–219]:

1. Jurnal umum.

Jurnal adalah catatan kronologis peristiwa bisnis berdasarkan persamaan akuntansi. Jurnal dapat berupa jurnal umum atau jurnal khusus. Jurnal khusus merupakan jurnal yang mencatat semua kegiatan transaksi akuntansi perusahaan sedangkan jurnal khusus menangkap jenis transaksi tertentu. Mereka biasanya dicadangkan untuk transaksi yang sering terjadi dalam suatu organisasi. Perusahaan dapat membuat jurnal khusus untuk hampir semua jenis transaksi. yang umum adalah jurnal penjualan, jurnal pembelian, jurnal penerimaan kas, dan jurnal pengeluaran kas.

2. Buku besar.

Entri jurnal menunjukkan semua aspek transaksi tertentu. Setiap entri menunjukkan debit dan kredit yang dihitung, tanggal transaksi, akun yang terpengaruh, dan keterangan tentang acara tersebut. Setelah sistem informasi akuntansi tersebut merekam entri jurnal, selanjutnya entri entri dalam buku besar umum. Pada sistem informasi akuntansi, buku besar adalah kumpulan informasi moneter terperinci tentang berbagai aset, liabilitas, ekuitas pemilik, pendapatan, dan pengeluaran.

3. Neraca saldo

Setelah sistem informasi akuntansi mempostingnya ke buku besar, selanjutnya dapat dibuat neraca saldo. Neraca saldo adalah daftar semua akun yang dikategorikan kepada 2 jenis utama yaitu aktiva dan pasiva. Pada neraca saldo, setiap rekening akan dikelompokkan pada aktiva yang berupa aktiva lancar dan aktiva tetap serta pasiva yang berupa utang-piutang dan modal. Kemudian setiap saldo dari rekening tersebut dijumlahkan dan diperiksa apakah total saldo pada kategori aktiva sama dengan total saldo pada kategori pasiva atau tidak. Setelah jumlah dalam saldo percobaan ini sama, maka neraca saldo telah dikatakan *balance* sedangkan jika tidak, terdapat suatu kesalahan atau transaksi yang tidak sesuai.

4. Laporan keuangan

Setelah saldo dalam neraca saldo ini *balance*, sistem informasi akuntansi siap untuk menghasilkan laporan keuangan. Laporan keuangan adalah *output* utama pada sistem akuntansi keuangan. Siklus akuntansi tidak berakhir ketika sistem informasi akuntansi menghasilkan laporan keuangan.

Sistem informasi akuntansi menggunakan no rekening untuk membuat bagan akun perusahaan. No rekening adalah kode berurutan dimana blok angka tertentu disediakan. Tabel 2.1 adalah no rekening umum yang biasa digunakan: [6, p. 220]

Tabel 2.1 No rekening yang digunakan untuk bagan akun perusahaan

Akun utama
100-199 Aset lancar
200-299 Aset tidak lancar
300-399 Kewajiban lancar
400-499 Kewajiban jangka panjang
500-599 Ekuitas pemilik
600-699 Pendapatan
700-799 Harga pokok penjualan
800-899 Biaya operasi
900-999 Penghasilan & biaya yang tidak beroperasi

Dimana aset lancar umumnya digunakan no rekening pada tabel 2.2: [6, p. 220]

Tabel 2.2 No rekening yang digunakan untuk akses lancar

Aset lancar
100 Kas
110 Surat berharga
120 Saham biasa
121 Saham pilihan
122 Obligasi
123 Sertifikat pasar uang
124 Sertifikat bank
125 Piutang usaha
130 Biaya dibayar dimuka
140 Persediaan
150 Wesel tagih

2.5.2 Laporan Informasi Akuntansi

Sebagian besar data akuntansi dikumpulkan oleh suatu organisasi sehingga muncul pada beberapa jenis laporan internal dan atau eksternal. Maka dari itu, desain sistem informasi akuntansi yang efektif biasanya dimulai dengan laporan yang diharapkan pengguna sistem. *Output* pada sistem informasi akuntansi adalah: [6, p. 221]

1. laporan kepada manajemen,
2. laporan kepada investor dan kreditor,
3. data lengkap mengenai transaksi transkrip data, dan
4. uraian lengkap tentang akun rekening (misalnya, catatan inventaris).

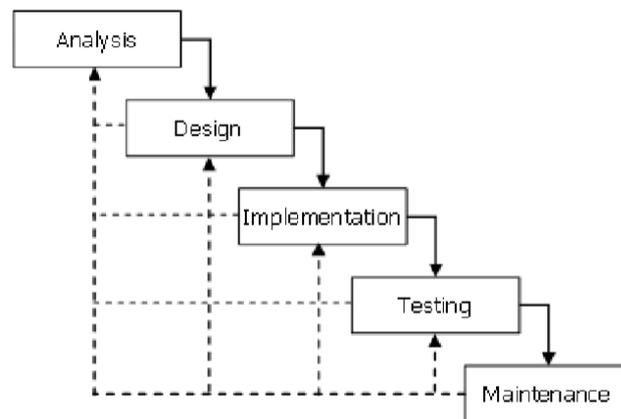
Hal yang paling penting dari keluaran ini adalah laporan kepada manajemen karena laporan ini membantu kegiatan pengambilan keputusan. Jika seorang manajer menanyakan sistem *database*, monitor menunjukkan data yang diminta dan sistem dapat menghasilkan laporan cetak hanya berdasarkan permintaan. Kadang-kadang format yang digunakan adalah grafis, seperti diagram lingkaran Format grafis lainnya termasuk grafik batang dan garis tren. Adanya grafik juga dapat meningkatkan laporan dalam bentuk apa pun. Laporan keluaran yang baik memiliki karakteristik yang sama terlepas dari jenisnya, seperti: [6, p. 221]

1. berguna
2. format yang mudah digunakan
3. mudah diidentifikasi
4. konsisten.

Laporan yang baik selalu mengandung identifikasi mendasar, termasuk judul, nomor halaman, dan tanggal. Laporan sistem informasi akuntansi harus konsisten dari waktu ke waktu, lintas level departemen atau divisi, dan dengan praktik akuntansi umum. Konsistensi dari waktu ke waktu memungkinkan para manajer untuk membandingkan informasi dari periode waktu yang berbeda. Demikian pula, laporan harus konsisten di seluruh departemen sehingga pengawas dapat membandingkan kinerja departemen serta format laporan harus konsisten dengan praktik akuntansi umum sehingga manajer dan investor dapat memahami dan menggunakan laporan ini. [6, p. 22]

2.6 System Development Life Cycle

System Development Life Cycle (SDLC) dalam sistem teknik, sistem informasi dan rekayasa perangkat lunak adalah proses menciptakan atau mengubah sistem, model, dan metodologi yang digunakan untuk mengembangkan sistem ini. Metode ini membangun suatu kerangka kerja perencanaan dan pengendalian berdasar proses pengembangan suatu sistem informasi. [15, p. 776] Salah satu model pada SDLC adalah *waterfall model*. *Waterfall model* adalah proses pengembangan perangkat lunak berurutan dimana kemajuan dianggap semakin menurun melalui fase yang harus dijalankan agar berhasil membangun sistem. Model ini mendefinisikan beberapa fase yang harus diselesaikan dan pindah ke fase berikutnya hanya ketika fase sebelumnya sepenuhnya dilakukan serta bersifat rekursif karena setiap fase dapat diulang hingga disempurnakan seperti pada gambar 2.2: [16, p. 2]



Gambar 2.2 SDLC model *waterfall*

Pada dasarnya, model *waterfall* terdiri dari lima fase yaitu: [16, p. 2]

1. *Analysis*

Fase ini adalah deskripsi lengkap dan komprehensif tentang sistem yang akan dikembangkan serta melibatkan analisis sistem dan bisnis untuk menentukan persyaratan fungsional dan non-fungsional. Biasanya, persyaratan fungsional ditentukan dengan menggunakan case yang menggambarkan interaksi pengguna dengan sistem. Sebaliknya, persyaratan non-fungsional merujuk pada berbagai kriteria, kendala, batasan, dan persyaratan yang dikenakan pada desain dan pengoperasian sistem daripada pada perilaku tertentu.

2. *Design*

Fase ini adalah proses perencanaan dan pemecahan masalah untuk solusi sistem. Fase ini melibatkan pengembang dan perancang sistem untuk menentukan rencana solusi yang meliputi desain algoritma, desain arsitektur sistem, skema konseptual *database* dan desain diagram logis, desain konsep, desain antarmuka pengguna grafis, dan definisi struktur data.

3. *Implementation*

Fase ini mengacu pada realisasi persyaratan bisnis dan spesifikasi desain menjadi program yang dapat dieksekusi, *database*, situs web, atau komponen sistem melalui pemrograman dan penyebaran. Fase ini adalah tempat kode asli ditulis dan dikompilasi menjadi aplikasi operasional, dan dimana basis data dan file teks dibuat sehingga dengan kata lain, adalah proses mengubah seluruh persyaratan dan cetak biru menjadi lingkungan produksi.

4. *Testing*

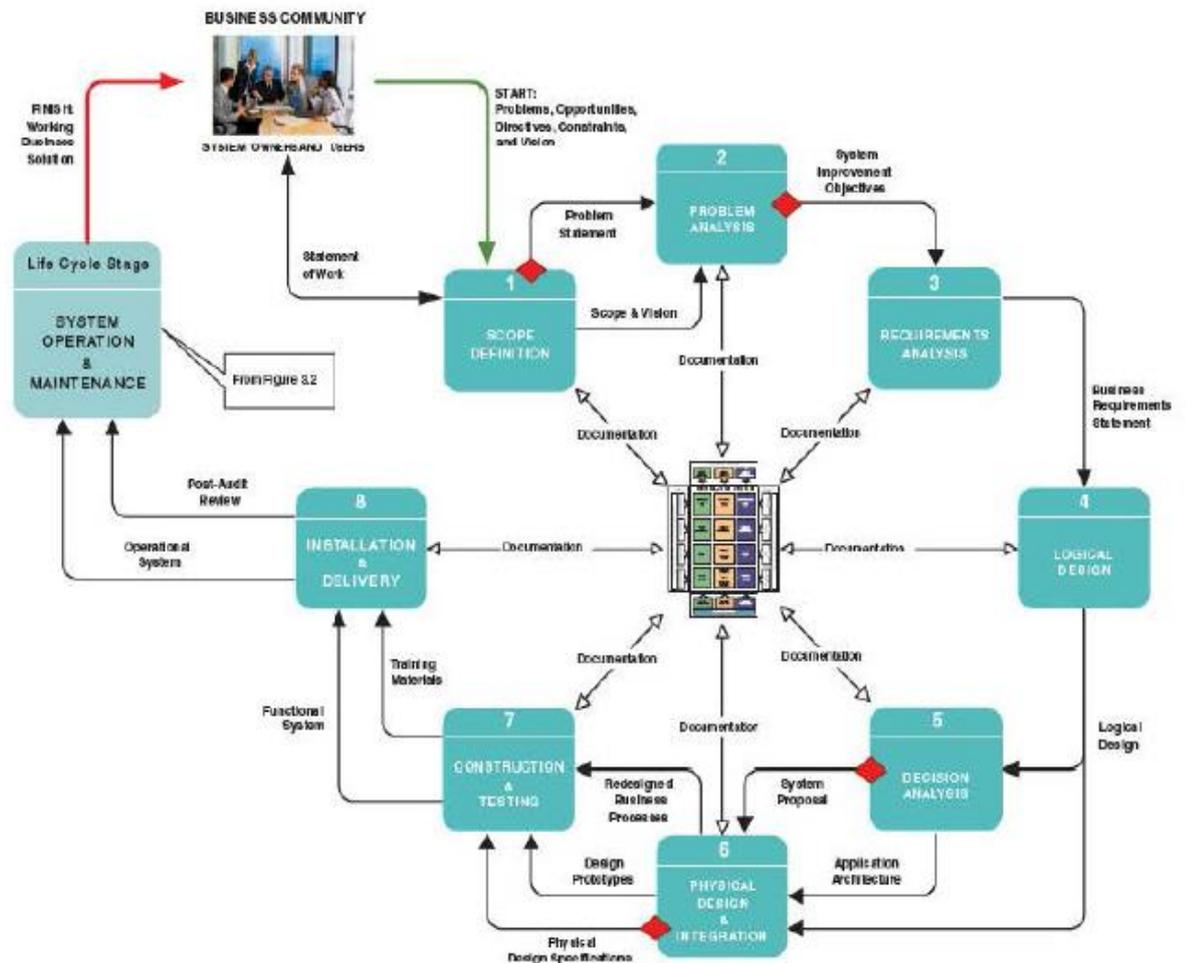
Fase ini juga dikenal sebagai verifikasi dan validasi yang merupakan proses untuk memeriksa apakah suatu solusi sistem memenuhi persyaratan dan spesifikasi asli dan bahwa itu mencapai tujuan yang dimaksudkan. Selain itu, fase *testing* adalah jalan keluar untuk melakukan *debugging* dimana *bug* dan gangguan sistem ditemukan, diperbaiki, dan disempurnakan.

5. *Maintenance*

Fase ini adalah proses memodifikasi solusi sistem setelah pengiriman dan penyebaran untuk memperbaiki *output*, memperbaiki kesalahan, dan meningkatkan kinerja dan kualitas. Kegiatan pemeliharaan tambahan dapat dilakukan dalam fase ini termasuk mengadaptasi sistem dengan lingkungannya, mengakomodasi kebutuhan pengguna baru, dan meningkatkan keandalan sistem.

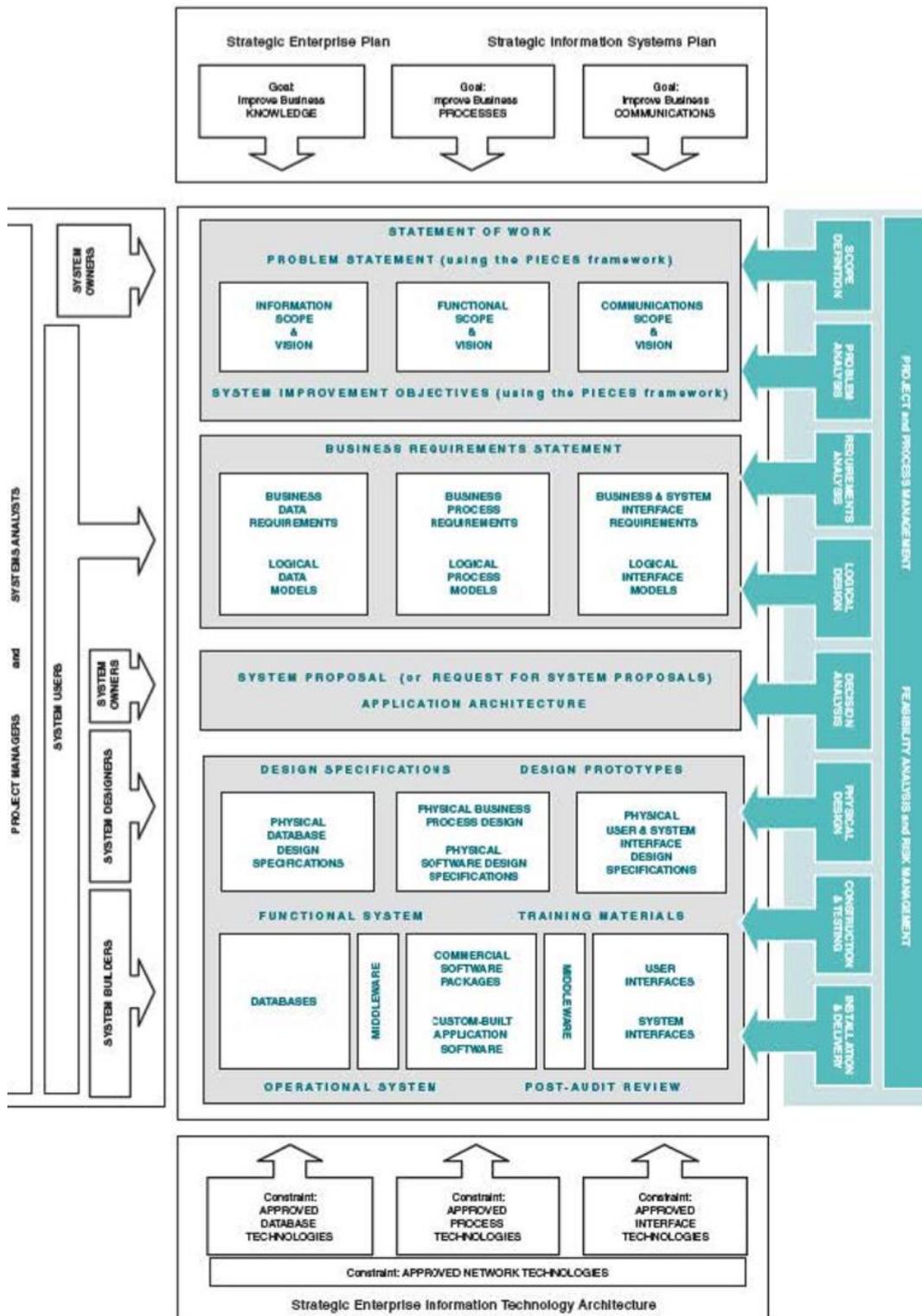
2.7 Framework the Application of System Thinking (FAST)

FAST adalah kerangka kerja cerdas yang cukup fleksibel untuk menyediakan berbagai jenis proyek serta strategi dan praktik-praktik gabungan berisi penggunaan metode pengembangan sistem yang dapat ditemukan dalam banyak metode komersial dan BGG [17, p. 126]. Pada proses pengembangan sistem informasi, FAST terdiri dari fase-fase yang dapat dilihat pada gambar 2.3: [18, pp. 77–80]



Gambar 2.3 Process of view system development

Gambar 2.4 mengilustrasikan mengenai pandangan pengembangan sistem dalam bentuk diagram blok: [18, p. 81]



Gambar 2.4 Building blocks view of system development

Terdapat 8 fase yang dilalui dalam pengembangan sistem berdasarkan metode *FAST*, yaitu [18, pp. 77–88]:

1. *Scope definition*

Tahap ini mempunyai tujuan sebagai berikut

- a. Mengidentifikasi masalah, peluang, serta tujuan dari pengembangan sistem
- b. mengidentifikasi ruang lingkup yang nantinya dikerjakan
- c. mengidentifikasi kelayakan dari perencanaan proyek tersebut

2. *Problem analysis*

Tahap ini memiliki kegiatan untuk mengidentifikasi berbagai penyebab masalah dan menganalisis serta memahami sistem yang sedang berjalan. Fase ini ditujukan untuk menganalisis setiap masalah yang muncul pada sistem sebelumnya agar mampu lebih memahami mengenai masalah tersebut serta membatasi ruang lingkup dari pengembangan sistem yang dilakukan. [19, p. 70]

3. *Requirement analysis*

Tahap ini memiliki kegiatan untuk mengidentifikasi berbagai kebutuhan dari pengguna dan menganalisis kebutuhan pada sistem

4. *Logical design*

Pada tahap ini, perancang menerjemahkan *business process* yang telah ditentukan ke dalam model sistem pada bentuk gambar (berupa diagram) untuk memvalidasi kebutuhan-kebutuhan berdasarkan kecukupan dan konsistensi. *Logical design* sebaiknya diinterpretasikan sebagai teknologi yang independen. Maksudnya setiap gambar mengilustrasikan sistem independen guna memberikan alternatif solusi.

5. *Decision analysis*

Tahap ini memiliki kegiatan mengidentifikasi berbagai alternatif sistem, menganalisis kelayakan dari analisis sistem tersebut, dan pemilihan alternatif sistem yang dikatakan paling layak

6. *Physic design and testing*

Kegiatan yang dilakukan pada tahap ini yaitu mentransformasikan ke dalam bentuk fisik yang nantinya akan digunakan sebagai bentuk basis data dalam merancang sistem. Selain itu, pada tahap ini juga dilakukan perancangan dari *input output* dan juga perancangan *database* dan *interface* untuk digunakan ke dalam fase berikutnya.

7. *Construction and testing*

Tahap ini perancang membangun serta menguji pada sistem yang telah dirancang sesuai kebutuhan dan spesifikasi rancangan tersebut serta mengimplementasikan antara sistem lama dengan sistem yang baru

8. *Installation*

Pada tahap ini dilakukan penerapan sistem yang baru serta pelatihan dan didokumentasikan. Saat ini dilakukan dengan tujuan mendukung sistem yang dirancang agar dapat beroperasi dengan baik serta dilakukan pemeliharaan sistem secara berkala.

Fase-Fase tersebut dikategorikan kepada beberapa fase klasik (standar) seperti yang tercantum pada tabel 2.3 [18, p. 79]

Tabel 2.3 Kategori setiap fase pada FAST

Fase FAST	Fase klasik			
	<i>Project initiation</i>	<i>System analysis</i>	<i>System design</i>	<i>System implementation</i>
<i>Scope definition</i>	X			
<i>Problem analysis</i>	X	X		
<i>Requirement analysis</i>		X		
<i>Logical design</i>		X		
<i>Decision analysis</i>	Fase transisi sistem			

Tabel 2.3 Kategori setiap fase pada *FAST* (lanjutan)

Fase <i>FAST</i>	Fase klasik			
	<i>Project initiation</i>	<i>System analysis</i>	<i>System design</i>	<i>System implementation</i>
<i>Physical design and integration</i>			X	
<i>Construction and testing</i>			X	X
<i>Installation and delivery</i>				X

Berdasarkan tabel tersebut, perencanaan dan analisis sistem dilakukan untuk mengidentifikasi dan menganalisis ruang lingkup pengembangan, permasalahan dan kebutuhan menunjang objek dan fungsi penelitian. Kemudian, perancangan sistem melakukan desain mengenai rancangan sistem berdasarkan kebutuhan yang ada. Sedangkan implementasi direncanakan dan dilaksanakan dengan membangun sistem mengacu pada rancangan yang ada guna mencapai tujuan. [20, p. 2]

2.8 Database

Database adalah kumpulan besar data terkait yang biasanya disimpan dalam file yang terkomputerisasi, ditautkan, dan dimanipulasi oleh paket perangkat lunak khusus yang disebut sistem manajemen *database* (DBMS). [6, p. 116]

2.8.1 Peranan *Database* dalam Sistem Informasi Akuntansi

Berikut adalah alasan lain pentingnya *database* dalam sistem: [6, pp. 117–118]

1. *Valuable information*

Informasi yang disimpan dalam *database* organisasi terkadang merupakan aset terpentingnya.

2. *Volume*

Pengelolaan *database* dengan ukuran sangat besar adalah tugas yang sangat besar dan seringkali menakutkan.

3. *Complexity*

Database dari beberapa organisasi dipusatkan (mis., Disimpan di satu lokasi di kantor pusat perusahaan atau dipelihara pada server file tunggal di jaringan area lokal). Namun, banyak *database* lainnya didistribusikan. Tetapi mendistribusikan informasi menjadikannya sulit untuk memastikan keakuratan, konsistensi, dan kelengkapan, mengamankan informasi dari akses yang tidak sah, dan membuat kembali file dengan cadangan jika terjadi kegagalan sistem.

4. *Privacy*

Database sering mengandung informasi sensitif misalnya, tarif gaji karyawan atau nomor kartu kredit pelanggan. Informasi ini harus dilindungi dari mereka yang tidak berwenang untuk memilikinya. Beberapa prosedur kontrol yang paling penting untuk sistem informasi akuntansi adalah prosedur yang melindungi *database* dari akses yang tidak beralasan.

5. *Irreplaceable Data*

Informasi dari sebagian besar sistem informasi akuntansi tentu unik untuk organisasi yang membuatnya dan, karenanya, seringkali tak ternilai harganya. Inilah sebabnya mengapa dimensi khusus dari manajemen *database* lagi-lagi keamanan file.

6. *Need for accuracy.*

Data yang disimpan dalam *database* komersial harus lengkap, komprehensif, dan akurat mati rasa. Ini terutama berlaku untuk *database* pemerintah. Penting juga bahwa sistem seperti itu mudah digunakan dan melayani misi strategis mereka

7. *Internet uses*

Database adalah komponen penting untuk sistem web perusahaan internal dan eksternal. *Database* ini menyimpan hal-hal seperti informasi produk untuk penjualan katalog online, email, data registrasi produk, dan data perusahaan terkini tentang peluang kerja, harga saham, dan kantor eksekutif. Aplikasi

Internet sering juga menyimpan data yang dimasukkan pelanggan seperti pesanan produk online, nomor kartu kredit, informasi berlangganan, pemesanan penerbangan, dan data pendaftaran mahasiswa-universitas.

2.8.2 Record dan Primary Key

Field data tertentu di setiap *record* tabel *database* adalah bagian dari apa yang disebut struktur *record*. Pada aplikasi akuntansi, baik jumlah *field* data atau ukuran *field* data yang diberikan di setiap *record* mungkin sama atau bervariasi. Maka dari itu dibutuhkan suatu *field* yang unik. *Primary key* adalah *field* data di setiap *record* yang secara unik membedakan satu *record* dari yang lain dalam tabel *database*. *Primary key* umumnya berupa penomoran atau kode unik pada suatu record seperti pada gambar 2.5: [6, pp. 118–119]

Department code (primary key)	Manager	Number of employees	Location	Secretary phone	Other info.
A	B. Wright	45	Bldg. 23	x8734	...

Gambar 2.5 Contoh sebuah record

Para profesional IT memperkirakan bahwa biayanya sekitar sepuluh kali lipat untuk mengoreksi informasi yang sudah ada dalam *database* seperti halnya memasukkannya dengan benar pada awalnya. Bahkan kesalahan sederhana dalam *database* dapat menyebabkan kesalahan mahal, keputusan buruk, atau bencana. Oleh karena itu, perangkat lunak yang digunakan untuk membuat *database* juga harus mencakup tes *edit* yang menjaga *database* dari entri data yang salah. Kontrol integritas data ini dirancang oleh pengembang *database* dan disesuaikan untuk aplikasi yang ada. [6, pp. 121–122] Berikut adalah langkah-langkah untuk membantu membuat tabel dan *record*: [6, p. 139].

1. Desain pertama; buat tabel dan catatan terakhir. Beberapa orang tidak punya waktu untuk melakukan hal yang benar hanya waktu untuk melakukan sesuatu. Jangan menjadi salah satu dari mereka. Definisi yang cermat dari entitas *database* dan hubungan mereka dapat menghindari masalah di kemudian hari.

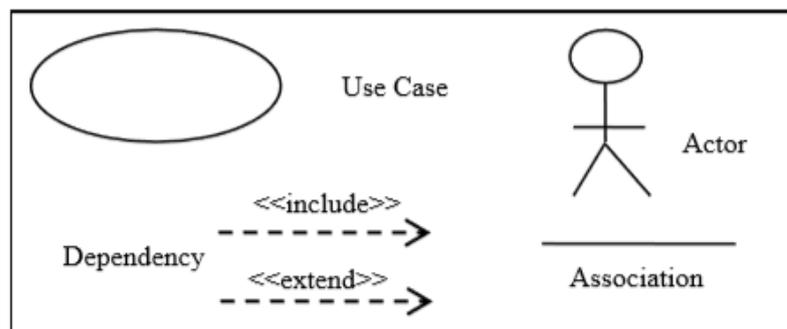
2. Beri nama tabel secara sistematis dan gunakan konvensional tbl refix. Bahkan *database* kecil mengandung banyak tabel, kueri, formulir, dan laporan. Menggunakan prefiks konvensional seperti tbl untuk tabel dan qry untuk kueri memungkinkan perancang *database* untuk membedakan di antara mereka. Anda juga dapat menemukan nama untuk tabel terkait secara sistematis .
4. Gunakan nama mnemonik untuk *field* data. Setiap *field* data dalam catatan harus memiliki nama, dan nama kecil membantu Anda mengingat apa yang masing-masing lapangan maksud. Misalnya, nama "Negara" lebih baik daripada "Jalur Alamat 3" untuk mewakili *field* data untuk negara pelanggan. Demikian pula, nama " Singkatan Negara " atau " Kode Negara " bahkan mungkin lebih baik jika Anda mengalokasikan hanya 2 digit untuk *field* ini.
5. Tetapkan tipe data yang benar untuk *field* data. Jika Anda berencana untuk memanipulasi *field* data secara matematis, Anda harus mendefinisikan *field* ini sebagai angka bukan *field* teks. Sebagai alternatif, Anda harus menggunakan tipe data teks untuk *field* seperti Jaminan Sosial, kartu kredit, atau nomor telepon. Angka-angka ini adalah kode yang disimpan dengan mudah sebagai nilai teks.
6. *Field* data yang menautkan tabel bersama harus memiliki tipe data yang sama. Jika Anda menggunakan *field* data dari tabel terpisah untuk menghubungkan dua tabel bersama, *field* ini harus dari tipe data yang sama persis. Dengan demikian, Anda tidak dapat menautkan tabel bersama-sama jika kunci asing di satu tabel adalah *field* teks dan yang lainnya adalah *field* tanggal. Seperti disebutkan sebelumnya, saat menggunakan *field* data *number* , tipe nomor juga harus cocok.
7. Batasi ukuran *field* data hingga panjang yang wajar. Access menetapkan ukuran default 255 karakter ke *field* teks. Jika kode negara hanya dua digit, maka ubah ukuran standar menjadi dua digit. Ini akan membatasi pengguna untuk memasukkan tidak lebih dari dua digit. Pedoman serupa berlaku untuk nomor jaminan sosial, nomor telepon, nomor produk, dan nilai-nilai serupa yang telah ditentukan, panjang tetap.

8. Gunakan *input mask*, yaitu templat yang menguraikan nilai yang diharapkan untuk *field* data. Contohnya nomor telepon adalah (999) 000-0000, yang membatasi nilai dalam *field* nomor telepon hingga 10 digit angka. *Input mask* membantu memastikan *input* yang akurat dan membantu mengurangi kesalahan.

2.9 Use case diagram

Use case diagram menggambarkan interaksi antara sistem dan sistem eksternal dan pengguna. Hal ini berarti bahwa *use case diagram* menggambarkan siapa yang akan menggunakan sistem dan dengan cara apa pengguna ingin berinteraksi dengan sistem. [21, p. 2] Selain itu, *use case diagram* secara tekstual untuk menggambarkan setiap urutan langkah dari setiap interaksi. [4, p. 49]

Use case diagram adalah diagram yang dibuat selama fase awal pengembangan perangkat lunak yang signifikan untuk merumuskan, menguatkan dan mendokumentasikan perilaku sistem. Persyaratan fungsional sistem yang sedang dipertimbangkan digambarkan dengan bantuan *use case diagram* untuk mengidentifikasi prosedur dan penggunaan sistem yang dirancang. Gambar 2.4 merupakan simbol yang biasa digunakan dalam *use case diagram* [22, p. 37]:



Gambar 2.6 Simbol-simbol dalam *use case diagram*

Simbol-simbol tersebut diartikan dengan [22, p. 37]:

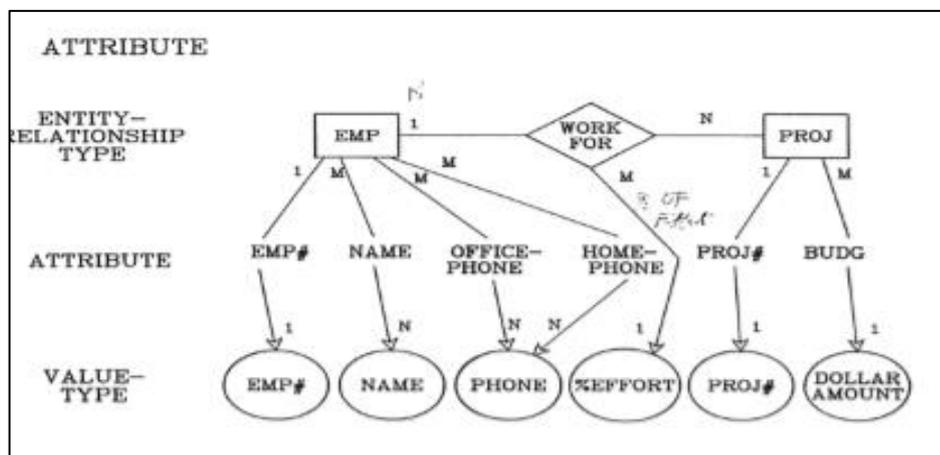
1. *Use case*: Mengilustrasikan persyaratan fungsional yang diberikan oleh semua pemangku kepentingan.

2. *Actor*: Pengguna sistem disebut aktor dan dapat berupa orang, perangkat, atau sistem lainnya.
3. *Dependency*: Dua kasus penggunaan dapat secara semantik terkait dan implementasi Kasus Penggunaan mungkin saling tergantung. Secara luas ada dua jenis dependensi yang ada dalam *Use case diagram* yang disebut *include dependency* dan *extended dependency*.
4. *Include dependency* adalah antara dua *use case* dimana perilaku *use case* dependen secara eksplisit termasuk di dalam basis *use case*
5. *Extend dependency* adalah antara dua *use case* dimana perilaku *use case* dependen secara implisit dimasukkan dalam basis *use case* dan *use case* dependen tidak dapat berdiri sendiri.
6. *Association relationship*: Seorang aktor dan *use case* terhubung menggunakan garis penghubung lurus yang disebut asosiasi antara keduanya.
7. *Generalization* adalah hubungan dimana aktor yang berbeda atau kasus penggunaan digeneralisasi masing-masing sebagai aktor. Konsep pewarisan ditunjukkan melalui generalisasi dalam *use case diagram*.

2.10 Entity relationship diagram

Entity relationship diagram (ERD) telah banyak digunakan dalam analisis terstruktur dan pemodelan konseptual. ERD merupakan suatu langkah penting dalam desain sistem informasi dan rekayasa perangkat lunak. Model ER memiliki pendekatan berlapis untuk mengatur informasi, dalam komponen dasar model ER: entitas, atribut, dan hubungan hanya dapat digabungkan dengan cara tertentu. Salah satu kunci dalam pemodelan ER adalah untuk mendokumentasikan entitas dan tipe hubungan dalam bentuk grafis yaitu *Entity-Relationship diagram*. [23, p. 4]

Berbagai notasi ERD telah dikembangkan untuk mewakili konsep di atas. Beberapa dari mereka mengizinkan hubungan n-ary sementara yang lain tidak. Beberapa dari mereka mewakili kardinalitas dan batasan partisipasi secara terpisah, sementara yang lain menggunakan notasi minimum dengan menggabungkan kardinalitas dan batasan partisipasi. Beberapa dari mereka menentukan batasan kardinalitas di seluruh hubungan sementara yang lain di dekat entitas. Salah satu notasi yang sering digunakan dalam pembuatan ERD adalah notasi Chen. Gambar 2.7 adalah bentuk dan simbol-simbol diagram ER dalam notasi Chen [23, p. 4].



Gambar 2.7 Simbol-simbol dalam diagram ER notasi Chen

Tipe entitas seperti EMP dan PROJ pada gambar disimbolkan sebagai kotak persegi panjang. Kemudian, tipe hubungan seperti WORK-FOR pada gambar disimbolkan sebagai belah ketupat. Selain itu, set nilai (domain) seperti EMP #, NAME, dan PHONE pada gambar disimbolkan sebagai lingkaran. Informasi kardinalitas hubungan juga diungkapkan misalnya dengan "1" atau "n" pada garis antara jenis entitas. Sedangkan jenis hubungan menunjukkan batas atas entitas dari jenis entitas yang berpartisipasi dalam hubungan itu [23, p. 4]. Selain itu, berikut ini merupakan makna dari istilah-istilah yang terdapat dalam ERD: [24, pp. 2–3]

1. Entity type

Entity type mewakili tipe objek yang dapat dibedakan. Dalam pemodelan dunia nyata, tipe entitas adalah objek bisnis penting yang berisi lebih dari satu properti. Kami hanya akan memanggil entitas, bukan tipe entitas, seperti dalam banyak

latihan. Entitas yang lemah adalah tipe khusus dari entitas yang keberadaannya bergantung pada entitas lain yang disebut entitas pemilik. Ketergantungan ini disebut ketergantungan keberadaan. Dengan demikian entitas yang lemah tidak memiliki pengidentifikasi sendiri. Oleh karena itu, pengidentifikasi entitas yang lemah adalah kombinasi dari pengidentifikasi entitas pemilik dan kunci parsial dari entitas yang lemah.

2. *Relationship type*

Relationship type mewakili hubungan antara atau di antara beberapa entitas. Dalam pemodelan dunia nyata, suatu hubungan mewakili suatu asosiasi yang perlu diingat oleh sistem *database*. Kami hanya akan memanggil hubungan, bukan tipe hubungan. Tipe hubungan bisa unary, binary, atau n-ary, tergantung pada apakah jumlah entitas yang terlibat dalam hubungan tersebut adalah 1, 2 atau lebih dari 2.

2. *Attribute*

Attribute adalah properti yang digunakan untuk menggambarkan entitas atau hubungan. Perhatikan bahwa beberapa metode tidak memungkinkan atribut dalam suatu hubungan. Atribut yang merupakan kunci utama dari relasi lain disebut kunci asing.

3. Batasan kardinalitas

Batasan kardinalitas menentukan jumlah instance hubungan tempat entitas dapat berpartisipasi. Secara teoritis, segala jenis hubungan dapat digunakan dalam mengidentifikasi entitas. Agar lebih sederhana, jenis hubungan yang akan digunakan dalam ER dibatasi dengan hanya menggunakan dua jenis hubungan biner dengan pemetaan 1:1, 1:n dan m:n dimana keberadaan entitas m dan n hubungannya bergantung pada keberadaan suatu entitas di sisi lain sisi hubungan. Misalnya pada pemetaan 1:n, 1 karyawan mungkin akan memiliki n (0,1,2,3,...) kandungan keberadaan tanggungan pada karyawan yang sesuai. Metode identifikasi entitas dan hubungannya dengan entitas lain dapat diterapkan secara rekursif sampai entitas tersebut dapat diidentifikasi dengan

nilai atribut mereka sendiri. Misalnya, primary key suatu departemen di suatu perusahaan dapat terdiri dari nomor departemen serta memiliki atribut nomor divisi dimana nomor divisi tersebut merupakan primary key pada suatu divisi di perusahaan tersebut.[25, p. 18]

4. Batasan partisipasi

Batasan partisipasi menentukan apakah *instance* dari entitas dapat ada tanpa berpartisipasi dalam hubungan dengan entitas lain. Batasan ini sesuai dengan batasan minimum dalam beberapa notasi. Partisipasi total (atau wajib) dan parsial (atau opsional) adalah dua jenis partisipasi. Partisipasi total ada ketika *instance* entitas tidak dapat ada tanpa berpartisipasi dalam hubungan dengan *instance* entitas lainnya. Partisipasi parsial ada ketika *instance* entitas dapat ada tanpa berpartisipasi dalam hubungan dengan *instance* entitas lainnya. Beberapa metode menggabungkan kardinalitas dan batasan partisipasi dan mewakilinya menggunakan batas minimum dan maksimum dalam bentuk notasi (min, maks).

5. *Generalization/specialization*

Generalization/specialization menetapkan hubungan superclass dan subclass antara tipe entitas. Dalam hierarki generalisasi / spesialisasi, ada dua kendala - *disjoint dan complete* . Batasan disjoint menentukan apakah suatu entitas dapat muncul di lebih dari satu entitas *subclass* (tumpang tindih) atau tidak (*disjoint*). Spesialisasi dikatakan memungkinkan tumpang tindih jika satu *instance* entitas di kelas super dapat muncul di beberapa entitas subclass. Kalau tidak, *subclass*-nya terpisah. Kendala kedua adalah kendala kelengkapan. Ini menentukan apakah *instance* entitas kelas super dapat ada tanpa memiliki setidaknya satu entitas *subclass* (spesialisasi sebagian) atau tidak (total spesialisasi).

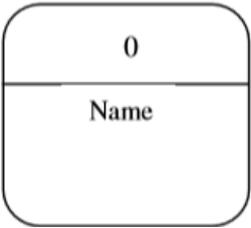
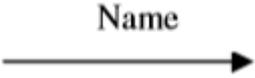
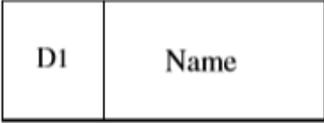
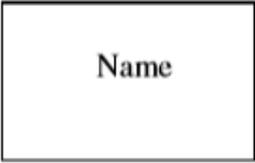
2.11 *Data flow diagram*

Model proses dapat digunakan untuk mewakili proses atau kegiatan yang dilakukan dalam suatu sistem dan menunjukkan cara perpindahan data di antara proses. Untuk membuat diagram model proses, diperlukan *data flow diagram*. *Data flow diagram* (DFD) didefinisikan sebagai alat grafis yang memungkinkan analisis dan pengguna

sistem untuk menggambarkan aliran data dalam suatu sistem informasi. DFD digunakan terutama dalam proses pengembangan sistem karena aliran data yang terdokumentasi penting untuk memahami sistem informasi akuntansi, banyak sumber menggunakan DFD untuk menggambarkan aliran data dalam sistem informasi akuntansi yang sedang dibahas. [6, p. 88]

Biasanya, sistem dapat berbasis fisik atau logis, manual atau komputer. Simbol diagram aliran data terdiri dari empat simbol yaitu proses, aliran data, penyimpanan data dan entitas eksternal. Set simbol standar yang biasa digunakan, dirancang oleh simbol Gane dan Sarson yang dapat dilihat pada tabel 2.3 [26, p. 61].

Tabel 2.4 Simbol dan nama elemen DFD

Simbol	Nama elemen
	Proses
	Aliran data
	Gudang data
	Entitas eksternal

Gambar-gambar diatas menggambarkan empat simbol dasar yang digunakan dalam DFD. Kotak atau kotak yaitu entitas eksternal mewakili pihak eksternal atau tujuan data. Misalnya, pelanggan. Garis aliran data adalah garis dengan panah yang menunjukkan arah aliran data. Dengan demikian, garis aliran data menunjukkan

jalur yang diikuti, keluar dari, atau melalui sistem yang sedang dipelajari karena setiap simbol sumber data akan memiliki satu atau lebih jalur aliran data yang menjauh darinya, dan setiap simbol tujuan data akan memiliki satu atau lebih jalur aliran data yang mengarah ke sana. Kotak dengan sudut lengkung dalam DFD menunjukkan entitas sistem atau proses yang mengubah atau mentransformasikan data. Sedangkan persegi panjang dengan kotak digambarkan gudang data atau tempat dimana data disimpan.

Pada DFD, tampilan tingkat tertinggi pada sistem dikenal sebagai diagram konteks. Diagram konteks adalah diagram alir data dari 10 ruang lingkup sistem organisasi yang menunjukkan batas-batas sistem, entitas eksternal yang berinteraksi dengan sistem dan arus informasi utama antara entitas dan sistem. Diagram konteks menunjukkan *input* dan *output* aplikasi sumber data dan tujuan di permukaan aplikasi. Hal ini menunjukkan bahwa diagram konteks merupakan DFD yang paling sederhana tapi bersifat umum. [6, p. 89]

Level berikutnya dari diagram aliran data disebut diagram aliran data level 0 yang mewakili proses utama sistem, aliran data, dan penyimpanan data pada tingkat detail yang tinggi. Setiap proses dalam diagram aliran data level n-1 akan didekomposisi menjadi diagram aliran data level bawah yang merupakan diagram aliran data level n. Prinsip utama dalam diagram aliran data adalah untuk memastikan keseimbangan yang berarti bahwa diagram aliran data pada satu tingkat secara akurat diwakili dalam diagram aliran data tingkat berikutnya ketika mengembangkan proyek. Level dekomposisi yang ideal adalah mendekomposisi sistem sampai analis dan pengguna sistem dapat memberikan deskripsi terperinci proses dimana deskripsi proses tidak lebih dari satu halaman. [27, pp. 96–97]

2.11.1 Syntax dan Aturan Semantik DFD

DFD diilustrasikan pergerakan data antara entitas eksternal dan proses dan penyimpanan data dalam suatu sistem. Teknik DFD efektif untuk mengekspresikan persyaratan fungsional untuk sistem kompleks besar. Definisi 1 memberikan definisi DFD dimana ada empat simbol dalam DFD yaitu: [27, p. 98]

Definisi 1: DFD terdiri dari:

1. Proses
2. Aliran Data
3. Penyimpan Data
4. Entitas Eksternal

Dimana:

- Suatu proses adalah kegiatan atau fungsi yang dilakukan untuk beberapa alasan bisnis tertentu
- Aliran data adalah satu bagian data atau kumpulan logis beberapa informasi
- Penyimpanan data adalah kumpulan data yang disimpan dalam beberapa cara
- Entitas eksternal adalah orang, organisasi, atau sistem yang berada di luar sistem tetapi berinteraksi dengannya.

Seperti yang telah dipaparkan sebelumnya, DFD tingkat tertinggi dikenal sebagai diagram konteks. Menyimpan data biasanya tidak termasuk dalam diagram konteks. Oleh karena itu diagram konteks diuraikan menjadi diagram tingkat yang lebih rendah yaitu DFD level 0. Setelah itu digunakan diagram dekomposisi dari diagram konteks. [28, p. 7] Faktanya, setiap proses pada DFD level 0 dapat didekomposisi menjadi DFD yang lebih eksplisit, yang disebut diagram level 1 dan selanjutnya dapat didekomposisi menjadi diagram level-rendah berikutnya. [27, p. 98]

Secara umum, ada dua jenis masalah yang berbeda secara fundamental yang dapat terjadi dalam DFD yang merupakan kesalahan sintaks dan kesalahan semantik. Sintaks dari DFD adalah bagaimana komponen saling berhubungan melalui aliran data dan komponen apa yang merupakan subsistem yang dimodelkan. Di sisi lain, semantik DFD adalah bagaimana aliran data saling terkait dalam hal transformasi data. Kesalahan sintaksis lebih mudah ditemukan dan diperbaiki daripada kesalahan semantik karena ada aturan yang jelas yang dapat digunakan untuk mengidentifikasi mereka. Ada seperangkat aturan yang harus diikuti oleh analis ketika menggambar DFD untuk mengevaluasi DFD untuk kebenaran. Definisi 2 hingga Definisi 8 menyatakan aturan ini. [27, pp. 98–99]

Definisi 2: Aturan diagram alir data:

1. Setidaknya satu *input* atau *output data flow* untuk entitas eksternal
2. Setidaknya satu *input data flow* dan / atau satu *flow data output* suatu proses
3. *Output data flow* biasanya memiliki nama yang berbeda dari *input* aliran data untuk suatu proses
4. Data mengalir hanya dalam satu arah • Setiap aliran data terhubung ke setidaknya satu proses

Definisi 3: Nama unik dalam DFD:

1. Nama unik (fase kata kerja), angka dan deskripsi untuk suatu proses
2. Nama unik yang merupakan kata benda dan deskripsi untuk aliran data
3. Nama unik yang merupakan kata benda dan deskripsi untuk penyimpanan data
4. Nama unik yang merupakan kata benda dan deskripsi untuk entitas eksternal

Definisi 4: Konsistensi:

1. Setiap rangkaian diagram alir data harus memiliki satu diagram konteks.

Definisi 5: Sudut Pandang Konsistensi:

1. Ada sudut pandang konsistensi untuk seluruh rangkaian DFD.

Definisi 6: Dekomposisi:

1. Setiap proses sepenuhnya dan sepenuhnya dijelaskan oleh proses pada DFD anak-anaknya.

Definisi 7: Menyeimbangkan:

1. Setiap aliran data, penyimpanan data, dan entitas eksternal pada DFD level yang lebih tinggi ditampilkan pada DFD level bawah yang menguraikannya.

Definisi 8: Penyimpanan Data:

1. Untuk setiap penyimpanan data, data tidak dapat berpindah langsung dari satu penyimpanan data ke penyimpanan data lainnya.
2. Data harus dipindahkan oleh suatu proses.

Definisi 2 hingga 8 menjelaskan aturan dasar DFD. Konsistensi antara diagram konteks dan DFD sangat penting dan aturan untuk konsistensi ini ditangkap dalam Definisi 4 dan 5. Mengikuti masalah konsistensi, Definisi 6 membahas aspek dekomposisi proses ke tingkat DFD yang lebih rendah dan Definisi 7 membahas aspek penyeimbangan elemen DFD ke level DFD yang lebih rendah. Aturan sintaksis digunakan untuk memverifikasi kesalahan sintaksis dalam DFD. Aturan sintaks didefinisikan dalam Definisi 9. [27, p. 99]

Definisi 9: Sintaks aturan DFD:

1. Setidaknya satu *input data flow* untuk suatu proses
2. Setidaknya satu *output data flow* untuk suatu proses
3. Proses dari entitas eksternal tidak dapat pindah langsung ke entitas eksternal lain
4. Setidaknya satu *input data flow* untuk penyimpanan data
5. Setidaknya satu aliran data *output* untuk penyimpanan data
6. Data suatu penyimpanan tidak dapat langsung dipindahkan ke penyimpanan lain

Berdasarkan Definisi 9, enam aturan sintaks digunakan untuk memverifikasi kebenaran diagram konteks dan DFD level 0. Namun, aturan sintaks penyimpanan data hanya diterapkan pada DFD level 0. Aturan semantik digunakan untuk memverifikasi kesalahan semantik dari diagram konteks ke DFD level 0. Aturan semantik didefinisikan dalam Definisi 10. [27, p. 100]

Definisi 10: Aturan semantik DFD:

1. Total dan nama entitas eksternal diagram konteks sama dengan level 0 DFD
2. Jumlah total dan nama aliran data antara proses dan entitas eksternal dalam diagram konteks sama dengan level 0 DFD
3. Total dan nama entitas eksternal level 0 DFD sama dengan diagram konteks
4. Jumlah total dan nama aliran data antara proses dan entitas eksternal di level 0 DFD sama seperti pada diagram konteks

Aturan semantik yang didefinisikan dalam Definisi 10 digunakan untuk melakukan pemeriksaan konsistensi dari diagram konteks ke DFD level 0.