

BAB II

TINJAUAN PUSTAKA

2.1 Pengertian Sistem Informasi

Sistem Informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan [1].

Suatu sistem informasi adalah suatu kegiatan dari prosedur-prosedur yang diorganisasikan, jika dieksekusi akan menyediakan informasi untuk mendukung pengambilan keputusan dan pengendalian di dalam organisasi [2].

Suatu sistem informasi adalah suatu kombinasi dari orang-orang, fasilitas, teknologi, media, prosedur-prosedur dan pengendalian yang ditujukan untuk mendapatkan jalur komunikasi penting, memproses tipe transaksi rutin tertentu, memberi sinyal kepada manajemen dan yang lainnya terhadap kejadian-kejadian internal dan eksternal yang penting dan menyediakan suatu dasar untuk pengambilan keputusan [2].

Sistem informasi adalah sebuah rangkaian prosedur formal dimana data dikumpulkan, diproses menjadi informasi dan didistribusikan kepada pemakai [2].

Setiap organisasi harus menyesuaikan informasinya dengan kebutuhan pemakaiannya. Secara umum tujuan tujuan sistem informasi pada perusahaan adalah sebagai berikut [2]:

1. Untuk mendukung fungsi kepengurusan manajemen. Kepengurusan merujuk ke tanggung jawab manajemen untuk mengatur sumber daya perusahaan secara benar. Sistem informasi menyediakan informasi tentang kegunaan sumber daya ke pemakai eksternal melalui laporan keuangan tradisional dan laporan-laporan yang diminta.
2. Untuk mendukung pengambilan keputusan manajemen. Sistem informasi memberikan para manajer informasi yang mereka perlukan untuk melakukan tanggung jawab pengambilan keputusan.

3. Untuk mendukung kegiatan operasi perusahaan harian, sistem informasi menyediakan informasi bagi personal operasi untuk membantu mereka melakukan tugas mereka setiap hari dengan efisien dan efektif.

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*building block*), yang terdiri dari blok masukan, blok model, blok keluaran, blok teknologi, blok basis data dan blok kendali. Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran [1].

1. Blok Masukan (*input block*).

Input mewakili data yang dimasukkan kedalam sistem informasi. *Input* yang dimaksud adalah metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen-dokumen dasar.

2. Blok Model (*model block*).

Blok ini terdiri dari kombinasi prosedur, logika dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan

3. Blok Keluaran (*output block*).

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi (*technology block*).

Teknologi merupakan “*toolbox*” dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 3 (tiga) bagian utama yaitu teknisi (*brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

5. Blok Basis Data (*database block*).

Basis data (*database*) merupakan kumpulan daya yang saling berkaitan dan berhubungan satu sama lain, tersimpan di perangkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang

dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut *DBMS (Database Management Management System)*.

6. Blok Kendali (*control block*).

Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, ketidakefisienan, sabotase dan lain sebagainya, beberapa pengendalian perlu dirancang dan diterapkan untuk menyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

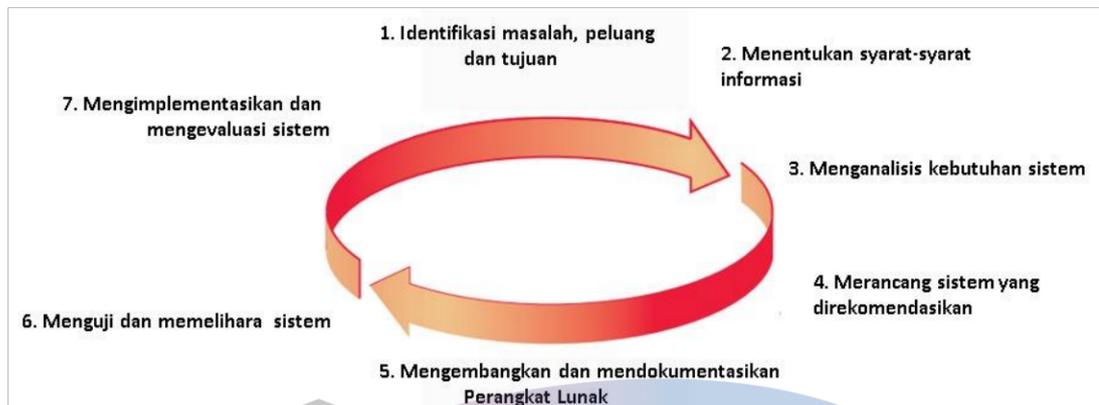
2.2 Penjualan Pembelian dan Persediaan

Pembelian adalah sebagai salah satu fungsi dari pembelanjaan atau merupakan kegiatan dari pembelanjaan. Pembelian sama pentingnya dengan penjualan, yaitu, untuk memenuhi kebutuhan setiap perusahaan, seperti kebutuhan peralatan kantor, gedung, peralatan produksi dan lain sebagainya [7].

Penjualan merupakan sebuah proses dimana kebutuhan pembeli dan kebutuhan penjualan dipenuhi melalui antar pertukaran informasi dan kepentingan [7]. Persediaan merupakan aktiva lancar yang ada dalam suatu perusahaan, apabila perusahaan tersebut perusahaan dagang maka persediaan diartikan sebagai barang yang disimpan untuk dijual dalam operasi normal perusahaan, sedangkan apabila perusahaan merupakan pengusaha manufaktur maka persediaan diartikan sebagai bahan baku yang terdapat dalam proses produksi atau yang disimpan untuk tujuan proses produksi [7].

2.3 System Development Life Cycle (SDLC)

Siklus hidup pengembangan sistem adalah pendekatan melalui beberapa tahap untuk menganalisis dan merancang sistem yang dimana sistem tersebut telah dikembangkan dengan sangat baik melalui penggunaan siklus kegiatan menganalisis dan pemakai secara spesifik [3].



Gambar 2.1 Tahapan Siklus Hidup Pengembangan Sistem

Siklus hidup pengembangan sistem dibagi ke dalam tujuh tahap, yang dilakukan secara simultan berulang dan saling tumpang tindih, yaitu [3].

1. Mengidentifikasi masalah, peluang dan tujuan.

Di tahap pertama dari siklus hidup pengembangan sistem ini, penganalisis mengidentifikasi masalah, peluang dan tujuan yang hendak dicapai. Tahap ini sangat penting bagi keberhasilan proyek, karena tidak seorangpun yang ingin membuang-buang waktu kalau tujuan masalah yang keliru.

2. Menentukan syarat-syarat.

Dalam tahap berikutnya penganalisa memasukan apa saja yang menentukan syarat-syarat informasi untuk para pemakai yang terlibat. Diantara perangkat-perangkat yang digunakan untuk menetapkan syarat-syarat informasi di dalam bisnis diantaranya ialah menentukan sampel dan memeriksa data mentah, wawancara, mengamati perilaku pembuat keputusan dan lingkungan kantor, dan *prototyping*.

3. Menganalisis kebutuhan-kebutuhan sistem.

Tahap berikutnya adalah menganalisis kebutuhan-kebutuhan sistem. Sekali lagi, perangkat dan teknik-teknik tertentu akan membantu menganalisis penentuan kebutuhan, perangkat yang dimaksud ialah penggunaan diagram alir data untuk menyusun daftar *input*, *process* dan *output* fungsi bisnis dalam bentuk grafik terstruktur, dari diagram aliran data, dikembangkan suatu kamus data.

4. Merancang sistem yang direkomendasikan.

Dalam tahap desain dari siklus hidup pengembangan sistem, penganalisis sistem menggunakan informasi-informasi yang terkumpul sebelumnya untuk mencapai

desain sistem informasi yang logis. Penganalisis merancang prosedur data-entry sedemikian rupa sehingga data yang dimasukkan ke dalam sistem informasi benar-benar akurat. Selain itu, penganalisis menggunakan teknik-teknik bentuk dan perancangan layar tertentu untuk menjamin keefektifan *input* sistem informasi.

5. Mengembangkan dan mendokumentasikan perangkat lunak.

Dalam tahap kelima dari siklus hidup pengembangan sistem, penganalisis bekerja bersama-sama dengan penrogram untuk mengembangkan suatu perangkat lunak awal yang diperlukan, beberapa teknik terstruktur untuk merancang dan mendokumentasikan perangkat lunak meliputi rencana struktur, *Nassi-Shneiderman charts*, dan *pseudocode*. Penganalisis sistem menggunakan salah satu dari semua perangkat ini untuk memprogram apa yang perlu deprogram.

6. Menguji dan mempertahankan sistem.

Sebelum sistem informasi dapat digunakan maka harus dilakukan pengujian terlebih dahulu. Akan bisa menghemat biaya bila dapat menangkap adanya masalah sebelum sistem tersebut ditetapkan. Sebagian pengujian dilakukan oleh pemrogram sendiri, dan lainnya dilakukan oleh penganalisis sistem.

7. Mengimplementasikan dan mengevaluasi sistem.

Di tahap terakhir dari pengembangan sistem, penganalisis membantu untuk mengimplementasikan sistem informasi. Tahap ini melibatkan pelatihan bagi pemakai untuk mengendalikan sistem, selain itu penganalisis perlu merencanakan konversi perlahan dari sistem lama ke sistem baru.

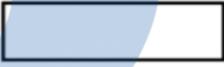
2.4 Data Flow Diagram (DFD)

DFD yang digunakan dalam bahasa Indonesia disebut sebagai DAD (Diagram Arus Data atau Diagram Aliran Data) memperlihatkan gambaran tentang masukan-proses-keluaran dari suatu sistem atau perangkat lunak, yaitu objek-objek data mengalir ke dalam perangkat lunak, kemudian ditransformasi oleh elemen-elemen pemrosesan, dan objek-objek data hasilnya akan mengalir keluar dari sistem atau perangkat lunak. Objek-objek data dalam penggambaran DFD biasanya direpresentasikan menggunakan tanda panah berlabel, dan transformasi-transformasi biasanya direpresentasikan menggunakan lingkaran-lingkaran yang sering disebut

sebagai gelembung-gelembung. DFD pada dasarnya digambarkan dalam bentuk hirarki, yang pertama sering disebut sebagai DFD *level 0* yang menggambarkan sistem secara keseluruhan sedangkan DFD-DFD berikutnya merupakan penghalusan dari DFD sebelumnya [4].

DFD memiliki representasi *node* dan aliran data. *Node* dapat berupa *data store*, *auxiliary store*, proses, atau terminator. Panah berhubungan dengan aliran data, *Node* dan aliran data harus diberi label yang menandakan *node* ataupun aliran data tersebut [5]. Simbol-simbol DFD tersebut dapat dilihat pada tabel 2.1 berikut ini: [6].

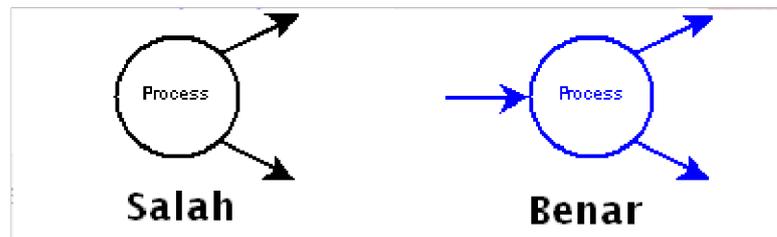
Tabel 2.1 Simbol-Simbol DFD

Nama	Yourdan & DeMarco	Gane & Sarson
Terminator/Entitas Luar		
Proses		
Alur Data		
Data Store		

2.5. Aturan Penggambaran DFD

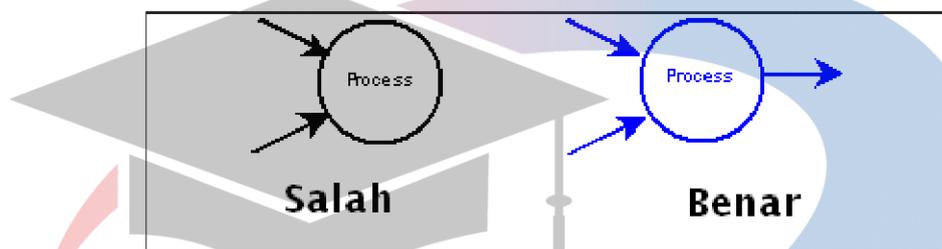
Terdapat beberapa aturan dalam penggambaran DFD yaitu [8]:

1. Semua proses harus memiliki nama yang unik. Jika dua arus data (atau *datastores*) memiliki label atau nama yang sama, keduanya harus mengacu pada arus data (atau *datastore*) yang sama.
2. *Input* ke suatu proses harus berbeda dengan *output* dari proses.
3. Suatu DFD sebaiknya tidak lebih dari tujuh proses.
4. Tidak ada proses yang hanya memiliki *output*. (Ini berarti proses menghasilkan informasi dari sesuatu yang tidak ada.). Jika suatu objek hanya memiliki *outputs*, maka itu haruslah sumber (*source*).



Gambar 2.2 Sumber dari suatu proses

5. Tidak ada proses yang hanya memiliki *input*. Jika suatu objek hanya memiliki *input*, maka itu haruslah tujuan (*sink*).

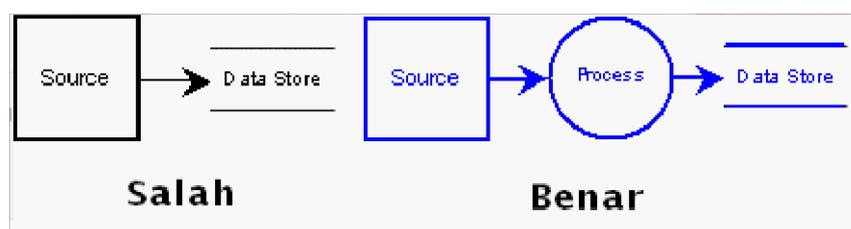


Gambar 2.3 Tujuan dari suatu proses

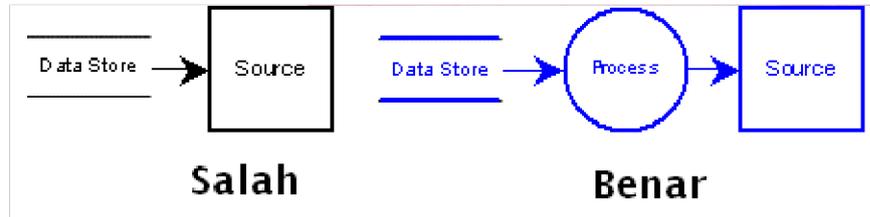
6. Suatu proses memiliki label / nama berupa kata kerja (*verb phrase*).
7. Data tidak bisa mengalir secara langsung dari satu data store ke data store lainnya. Data harus dialirkan oleh suatu proses.

Gambar 2.4 Aliran *datastore* ke proses dan ke *datastore*

8. Data tidak bisa mengalir secara langsung dari sumber luar / *outside source* ke suatu *datastore*. Data harus dialirkan oleh suatu proses yang menerima data dari *source* dan menemukannya pada *datastore*.

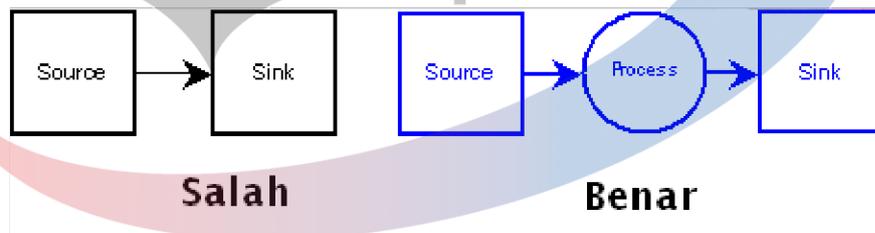
Gambar 2.5 Aliran *source* ke proses dan ke *datastore*

9. Data tidak bisa mengalir secara langsung ke tujuan / *outside sink* dari suatu *datastore*. Data harus dialirkan oleh suatu proses.



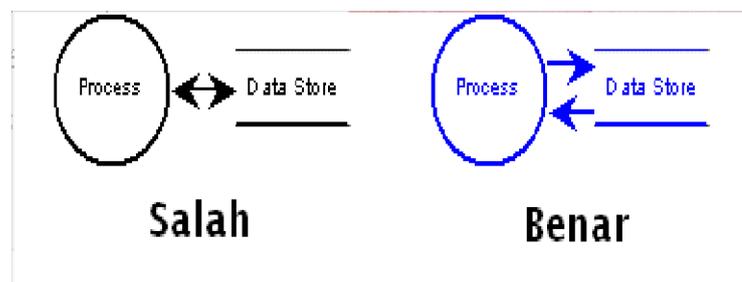
Gambar 2.6 Aliran *datastore* ke proses dan ke *source*

10. Suatu *datastore* memiliki nama atau label berupa kata benda.
11. Data tidak dapat mengalir secara langsung dari sumber (*source*) ke tujuan (*sink*). Data harus dialirkan oleh proses. Jika data dialirkan secara langsung dari *source* ke *sink* (dan tidak melibatkan pemrosesan) maka itu diluar lingkup sistem dan tidak ditunjukkan pada DFD.



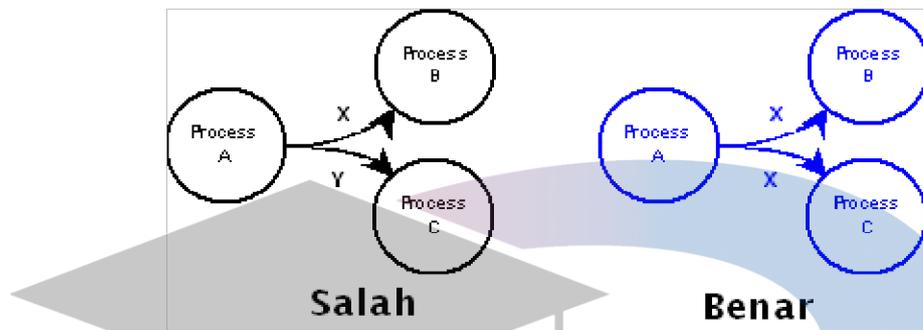
Gambar 2.7 Aliran *source* ke proses dan ke *sink*

12. Sumber (*source*) dan tujuan (*sink*) memiliki nama atau label berupa kata benda.
13. Suatu aliran data hanya memiliki satu arah. Bisa saja memiliki dua arah di antara suatu proses dan sebuah *datastore* untuk menunjukkan pembacaan sebelum pembaharuan (*update*). Untuk menunjukkan secara efektif pembacaan sebelum *update*, gambarkan dua aliran data yang terpisah sebab kedua langkah tersebut (*read dan update*) terjadi pada waktu yang berbeda.



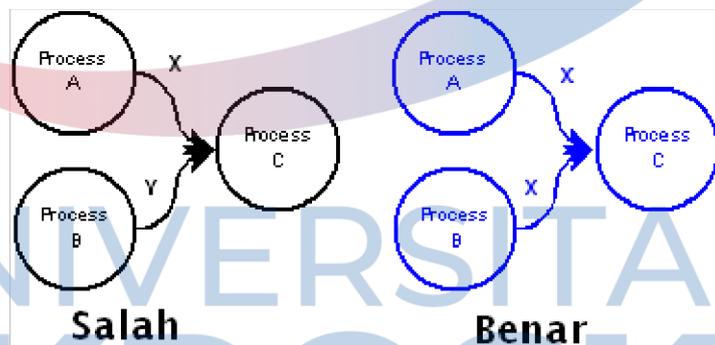
Gambar 2.8 Dua arah aliran antara proses dan *datastore*

14. Pencabangan pada aliran data memiliki makna data yang sama dari suatu lokasi ke dua atau lebih proses, *datastores*, sumber (*sources*) atau tujuan (*sink*) yang berbeda. Ini biasanya menunjukkan salinan data yang sama ke lokasi yang berbeda.



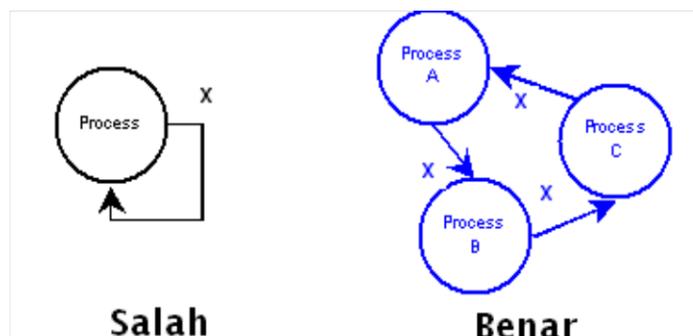
Gambar 2.9 Pencabangan aliran data

15. Gabungan suatu aliran data memiliki makna data yang sama datang dari dua atau lebih proses, *datastore*, sumber (*sources*) atau tujuan (*sinks*) yang berbeda ke suatu lokasi.



Gambar 2.10 Gabungan aliran data

16. Suatu aliran data tidak dapat langsung kembali ke proses yang sama. Setidaknya harus ada satu proses lain yang menangani aliran data, menghasilkan beberapa aliran data lain dan kembali ke proses semula,



Gambar 2. 11 Aliran data tidak langsung kembali ke proses yang sama

17. Suatu aliran data ke suatu *datastore* memiliki makna *update* (bisa *delete*, *add*, atau *change*).
18. Suatu aliran data dari suatu data store memiliki makna mengambil atau menggunakan.
19. Suatu aliran data memiliki nama atau label berupa kata benda (*noun phrase*). Lebih dari satu nama data dapat dinyatakan dengan satu simbol aliran data, sepanjang data pada aliran data dialirkan bersama-sama sebagai satu kesatuan data (satu paket).

2.6. Kamus Data

Kamus data adalah dokumen yang mengumpulkan dan mengkoordinasikan istilah data tertentu, dan menegaskan arti setiap istilah bagi orang yang berbeda dalam organisasi. Analis sistem menyusun kamus data untuk memandu mereka melalui analisis dan desain [3].

Selain menyediakan dokumentasi dan menghilangkan redundansi, kamus data dapat digunakan untuk [3]:

1. Validasi diagram aliran data untuk kelengkapan dan keakuratannya.
2. Berikan titik awal untuk mengembangkan layar dan laporan.
3. Tentukan isi data yang disimpan dalam *file*.
4. Kembangkan logika untuk proses diagram aliran data.
5. Buat XML (*Extensible Markup Language*).

2.7. Struktur Data

Struktur data biasanya dijelaskan menggunakan notasi aljabar. Metode ini memungkinkan seorang analis untuk menghasilkan tampilan elemen-elemen yang membentuk struktur data, bersama dengan informasi tentang elemen-elemen tersebut. Misalnya, analis akan menunjukkan apakah ada banyak elemen yang sama dalam struktur data (grup berulang) atau apakah dua elemen mungkin ada saling eksklusif dari

satu sama lain. Notasi aljabar menggunakan simbol-simbol berikut [10]:

1. Tanda sama dengan (=) berarti “terdiri dari.”
2. Tanda plus (+) berarti “dan.”

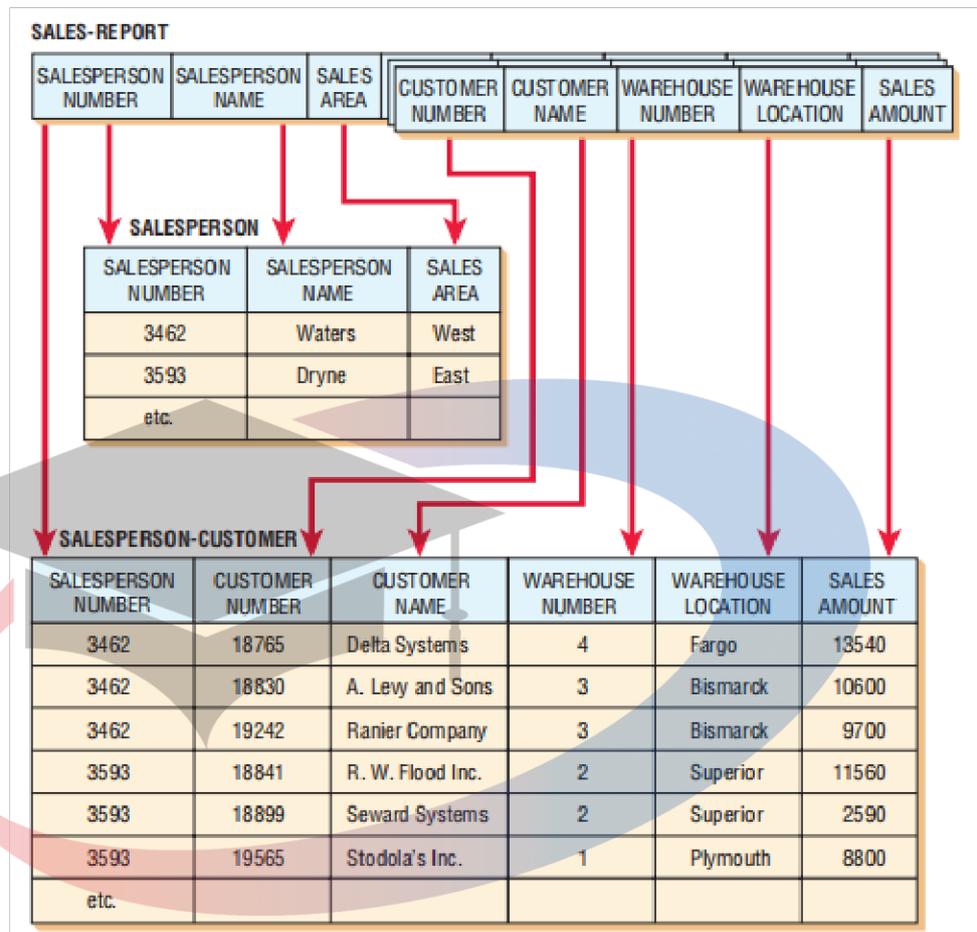
3. Tanda kurung kurawal { } menunjukkan elemen berulang, juga disebut grup atau tabel berulang. Mungkin ada satu elemen berulang atau beberapa dalam satu grup. Grup berulang mungkin memiliki kondisi, seperti jumlah pengulangan yang tetap, atau batas atas dan bawah untuk jumlah pengulangan.
4. Tanda kurung [] digunakan untuk situasi ini/atau. Salah satu elemen atau elemen lain mungkin ada, tetapi tidak keduanya. Elemen-elemen yang tercantum di antara tanda kurung adalah saling eksklusif.
5. Tanda kurung () digunakan untuk elemen opsional. Elemen opsional dapat dikosongkan pada layar entri dan mungkin berisi spasi atau nol untuk bidang numerik dalam struktur file.

Proses normalisasi merupakan proses pengelompokan elemen data menjadi tabel-tabel yang menunjukkan entitas dan relasinya. Proses ini selalu diuji pada beberapa kondisi, apakah ada kesulitan pada saat menambah data, menghapus, mengubah atau membaca data pada suatu *database*. Relasi dapat dipecah menjadi beberapa relasi lagi bila ada kesulitan pada proses membaca data [2].

2.8. Langkah- langkah Normalisasi

Dalam proses normalisasi terdapat tiga langkah penting. Setiap langkah melibatkan prosedur penting yang menyederhanakan struktur data. Langkah-langkahnya adalah sebagai berikut [9]:

1. Bentuk normalisasi pertama 1NF.
Langkah pertama dalam menormalkan relasi adalah menghilangkan kelompok berulang. Dalam contoh kita, relasi *SALES-REPORT* yang tidak ternormalisasi akan dipecah menjadi dua relasi terpisah. Hubungan baru ini akan diberi nama *SALESPERSON* dan *SALESPERSON-CUSTOMER*.

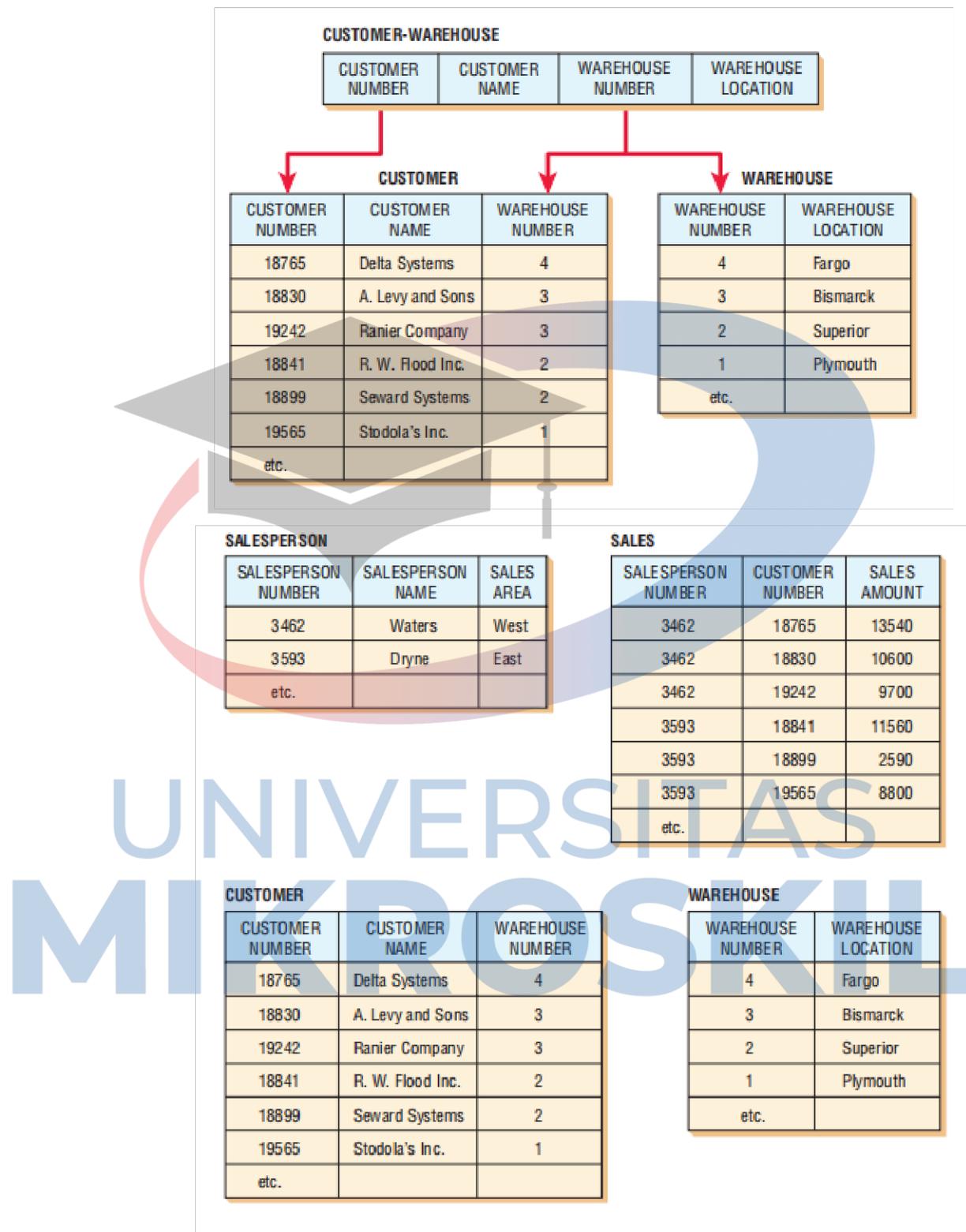


Gambar 2.12 Contoh Normalisasi 1NF

2. Bentuk normalisasi 2NF.

Dalam bentuk normal kedua, semua atribut akan bergantung secara fungsional pada kunci utama. Oleh karena itu, langkah selanjutnya adalah menghapus semua atribut yang bergantung sebagian dan menempatkannya di relasi lain. Pada gambar dibawah ini menunjukkan bagaimana hubungan

SALESPERSON-CUSTOMER dibagi menjadi dua hubungan baru: *SALES* dan *CUSTOMER-WAREHOUSE*.



Gambar 2.14 Contoh Normalisasi 3NF