

BAB II

TINJAUAN PUSTAKA

2.1 Konsep Sistem Informasi

Sistem – sistem mempunyai *input* (masukan) dan *output* (keluaran) mereka bergantung kepada beberapa himpunan masukan yang diproses untuk mencapai sasaran sistem. Semua sistem menghasilkan beberapa *output* yang dibutuhkan oleh sistem – sistem [1].

2.1.1 Pengertian Sistem

Sistem berasal dari bahasa Latin *Systema* dan bahasa Yunani *Sustema* adalah suatu kesatuan yang terdiri atas komponen dan elemen yang saling berkaitan untuk memudahkan aliran data dan informasi untuk mencapai suatu tujuan. Sistem merupakan kesatuan (*entity*) yang terdiri dari bagian – bagian yang disebut subsistem yang saling berkaitan dengan tujuan untuk mencapai tujuan – tujuan tertentu [1].

Sistem merupakan kumpulan elemen-elemen yang saling terkait dan bekerja sama untuk memproses masukan sampai menghasilkan keluaran / *output* yang diinginkan [2].

Jadi dapat disimpulkan bahwa sistem adalah sekumpulan data yang bekerja sama untuk membentuk suatu objek dengan tujuan agar menghasilkan sebuah hasil yang diinginkan dan adanya pencapaian suatu tujuan [1].

2.1.2 Pengertian Informasi

Informasi atau dalam bahasa Inggrisnya adalah *information*, berasal dari kata *informacion* bahasa Prancis. Kata tersebut diambil dari bahasa Latin, yaitu *informationem* yang artinya “konsep, ide, garis besar”. Informasi merupakan hasil pengolahan data dari satu sumber atau berbagai sumber, yang kemudian diolah sehingga memberikan nilai, arti dan manfaat yang mendukung proses pengambilan keputusan dalam perusahaan [3].

Adapun ciri – ciri informasi yang berkualitas adalah sebagai berikut [4] :

- a. Akurat, informasi yang diterima harus jelas mencerminkan maksudnya. Informasi menjadi tidak bernilai jika informasi tersebut sendiri tidak akurat karena dapat mengakibatkan kesalahan dalam pengambilan keputusan.
- b. Tepat waktu, informasi akan lebih bermanfaat jika informasi tersebut diterima oleh pengguna yang berada pada situasi dan kondisi yang tepat. Informasi menjadi tidak berharga atau bernilai jika informasi tersebut terlambat diterima karena tidak dapat dimanfaatkan untuk pengambilan keputusan.
- c. Relevan, merupakan informasi tersebut memiliki manfaat bagi pengguna. Relevansi informasi yang diterima oleh setiap orang dapat berbeda – beda
- d. Lengkap, informasi harus diberikan secara lengkap karena bila informasi yang dihasilkan sebagian, maka akan mempengaruhi pengguna dalam mengambil keputusan.
- e. *Correctness*, informasi yang dihasilkan harus memiliki nilai kebenaran / fakta yang terjadi dilapangan.

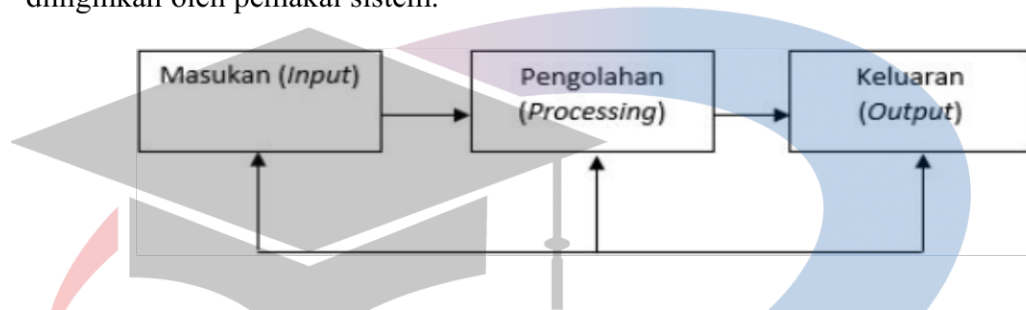
Jadi dapat disimpulkan bahwa informasi adalah data yang sudah diolah menjadi suatu informasi yang bermanfaat serta mendukung pengguna dalam mengambil keputusan [7].

2.1.3 Pengertian Sistem Informasi

Berdasarkan pengertian dari masing – masing kata yaitu sistem dan informasi, jika digabungkan maka sistem informasi adalah suatu komponen atau elemen yang saling berhubungan yang mengumpulkan (atau mendapatkan kembali), memproses, menyimpan, dan mendistribusikan informasi untuk mendukung pengambilan keputusan dan kendali dalam suatu organisasi. Sistem informasi juga dapat diartikan sebagai suatu sistem yang mengkombinasikan antara perangkat lunak (*software*), perangkat keras (*hardware*), orang (*people*), dan jaringan (*network*) yang saling berkaitan untuk menciptakan sebuah sistem dimana sistem tersebut dapat mengolah data menjadi informasi yang akan disajikan dan digunakan oleh penerima informasi [3].

Komponen sistem informasi terdiri dari [3]:

- a. *Input*, yaitu data yang dimasukkan kedalam sistem informasi untuk diproses.
- b. *Process*, data yang telah dimasukkan ke dalam sistem akan diproses dengan prosedur tertentu untuk menghasilkan output yang sesuai.
- c. *Output*, setelah *input* selesai diproses, maka sistem akan menghasilkan *output* yang diinginkan oleh pemakai sistem.



Gambar 2.1 Tiga Aktivitas Sistem Informasi

Jadi dapat disimpulkan bahwa sistem informasi adalah kombinasi seperangkat komponen yang terdiri dari orang, perangkat keras, perangkat lunak, jaringan dan data yang saling bekerja sama untuk mengumpulkan, mengolah, menyimpan, dan menyebarkan informasi untuk mendukung pengambilan keputusan, pengendalian, serta menganalisis masalah dan visualisasi dalam perusahaan [5].

Sistem informasi terdiri dari komponen yang disebut dengan istilah blok bangunan (*building block*) yang terdiri dari blok masukan (*input block*), blok model (*model block*), blok keluaran (*output block*), blok teknologi (*technology block*), blok basis data (*database block*) dan blok kendali (*controls block*). Sebagai suatu sistem keenam blok tersebut masing – masing saling berinteraksi antara satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sarannya [4].

1. Blok Masukan (*Input Block*)

Blok masukan mewakili data yang masuk kedalam sistem informasi. masukan disini termasuk metode – 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 dibasis data dengan cara 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 (*user*).

4. Blok Teknologi (*Technology Block*)

Blok teknologi digunakan untuk menerima masukan, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan. Teknologi terdiri dari 2 bagian utama yaitu perangkat lunak (*software*) dan perangkat keras (*hardware*).

5. Blok Basis Data (*Database Block*)

Blok basis data (*database block*) adalah kumpulan dari data yang saling berhubungan satu sama lain yang tersimpan diperangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok Kendali (*Controls Block*)

Pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal – hal dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi, kesalahan – kesalahan dapat langsung diatasi.

2.2 *System Development Life Cycle (SDLC)*

System Development Life Cycle (SDLC) adalah pendekatan melalui beberapa tahap untuk menganalisis dan merancang sistem. Siklus hidup pengembangan sistem digunakan untuk menganalisis dan merancang sistem yang dimana sistem tersebut telah dikembangkan dengan sangat baik melalui penggunaan siklus kegiatan penganalisis dan pemakai secara spesifik [3].

Siklus hidup pengembangan sistem dibagi dalam tujuh tahap, yaitu [4]:

a. Mengidentifikasi masalah, peluang, dan tujuan.

Di tahap pertama dari siklus hidup pengembangan sistem ini, penganalisis mengidentifikasi masalah, peluang, dan tujuan-tujuan yang hendak dicapai. Tahap ini sangat penting bagi keberhasilan proyek, karena tidak seorang pun yang ingin membuang-buang waktu jika tujuan masalah keliru. Tahap pertama ini berarti bahwa penganalisis melihat dengan jujur pada apa yang terjadi di dalam bisnis. Peluang yaitu situasi dimana penganalisis yakin bahwa peningkatan bisa dilakukan melalui penggunaan sistem informasi terkomputerisasi. Peluang dan tujuan didapatkan dengan mengidentifikasi masalah-masalah yang terjadi di dalam perusahaan tersebut.

b. Menentukan syarat – syarat informasi.

Pada tahap ini, penganalisis memasukkan apa saja yang menentukan syarat-syarat informasi untuk para pemakai yang terlibat. Perangkat-perangkat yang digunakan untuk menetapkan syarat-syarat informasi di dalam bisnis diantaranya yaitu menentukan sampel dan memeriksa data mentah, wawancara, mengamati perilaku pembuat keputusan, lingkungan kantor dan *prototyping*.

c. Menganalisis kebutuhan – kebutuhan sistem.

Dalam tahap ini, penganalisis melakukan analisis kebutuhan dari sistem dengan perangkat dan teknik-teknik tertentu. Perangkat yang dimaksud adalah penggunaan diagram alir data untuk menyusun daftar *input*, *proses*, dan *output* fungsi bisnis dalam bentuk grafik terstruktur. Dari diagram alir data, dikembangkan suatu kamus data berisikan daftar seluruh item data yang digunakan dalam sistem.

d. Merancang sistem yang direkomendasikan.

Dalam tahap desain dari siklus hidup pengembangan sistem, penganalisis sistem menggunakan informasi yang terkumpul sebelumnya untuk mencapai desain sistem informasi yang logis. Penganalisis merancang proses *data entry* sedemikian rupa sehingga data yang dimasukkan kedalam sistem informasi benar-benar akurat. Selain itu, penganalisis menggunakan teknik-teknik bentuk dan perancangan layar tertentu untuk menjamin keefektifan *input* sistem informasi. Bagian dari perancangan sistem informasi yang logis adalah peralatan antarmuka pengguna dimana antarmuka ini

menghubungkan pemakai dengan sistem sehingga peranannya menjadi sangat penting. Contoh dari antarmuka pemakai adalah *keyboard*, menu-menu pada layar serta berbagai jenis *Graphical User Interface* (GUI).

- e. Mengembangkan dan mendokumentasikan perangkat lunak.

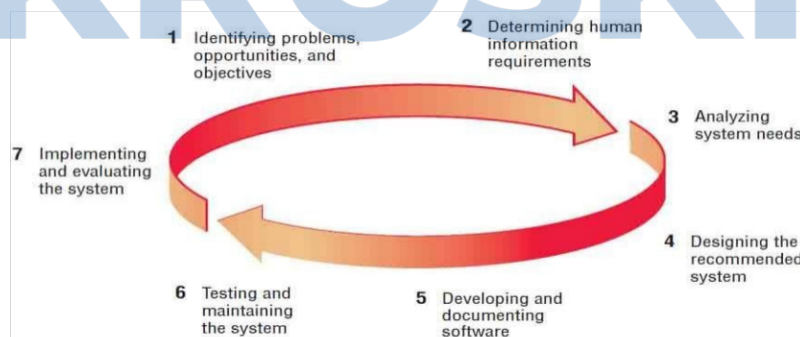
Pada tahap ini, penganalisis bekerja sama dengan *programmer* untuk mengembangkan suatu perangkat lunak awal yang diperlukan. Selain itu, penganalisis juga bekerja sama dengan pemakai untuk mengembangkan dokumentasi perangkat lunak yang efektif seperti melakukan prosedur secara manual, bantuan *online* serta membuat fitur *Frequently Asked Question* (FAQ) yang dikirimkan bersama-sama dengan penggunaan perangkat lunak dan apa yang harus dilakukan bila perangkat lunak mengalami masalah.

- f. Menguji dan mempertahankan sistem.

Pada tahap ini, harus dilakukan pengujian terlebih dahulu sebelum sistem informasi dapat digunakan. Jika dapat menangkap adanya masalah sebelum sistem tersebut ditetapkan maka akan lebih menghemat biaya. Sebagian pengujian dilakukan oleh *programmer* sendiri dan yang lainnya dilakukan oleh penganalisis sistem.

- g. Mengimplementasi dan mengevaluasi sistem.

Pada tahap terakhir ini, penganalisis membantu untuk mengimplementasikan sistem informasi. Tahap ini melibatkan pelatihan bagi pemakai untuk mengendalikan sistem. Evaluasi ditujukan sebagai bagian dari tahap terakhir siklus hidup pengembangan sistem.



Gambar 2.2 *System Development Life Cycle* (SDLC)

2.3 Teknik Pengembangan Sistem

Teknik pengembangan sistem adalah suatu proses pengembangan sistem yang formal dan presisi yang mendefinisikan serangkaian aktivitas, metode, *best practice* dan *tools* yang terautomatisasi bagi para pengembang dan manajer proyek dalam rangka mengembangkan dan merawat secara keseluruhan sistem informasi atau *software* [4].

2.3.1 Diagram Aliran Data / *Data Flow Diagram (DFD)*

DFD atau diagram aliran data adalah suatu diagram yang menggambarkan aliran data dari sebuah proses atau sistem informasi dimana menyediakan informasi mengenai *input* dan *output* dari setiap entitas dan proses itu sendiri. Pendekatan aliran data ini menekankan logika yang mendasari sistem. Dengan adanya DFD, dapat membantu pemakai sistem yang kurang memahami bidang komputer dapat lebih mengerti sistem yang sedang berjalan. Dengan menggunakan kombinasi dari empat simbol, penganalisis sistem dapat menciptakan suatu gambaran proses-proses yang bisa menampilkan dokumentasi sistem yang solid. Untuk mendukung aktivitas dalam fase desain, kita dapat mendeskripsi logika proses yang memberikan detail yang cukup. Detail logika proses akan digunakan untuk diterjemahkan ke dalam bahasa pemrograman yang tepat. Tujuan dari logika proses adalah untuk menjelaskan apa yang dilakukan proses dan memberikan informasi tambahan yang tidak disediakan oleh DFD. Dalam DFD, teknik deskripsi proses digunakan untuk menjelaskan proses level detail DFD, yaitu proses yang sudah primitive. Teknik ini dibuat untuk memperjelas serangkaian langkah ataupun aturan pemrosesan dan logika bisnis tertentu. Dengan menggunakan teknik ini, analis dapat membuat model proses yang akurat, lengkap, dan ringkas. Contoh logika proses [5]:

```

IF data pembayaran memadai AND belum ada pada tabel pembayaran
    THEN Insert data pembayaran ke tabel pembayaran
ELSE IF data pembayaran memadai AND sudah ada pada tabel pembayaran
    THEN Update data pembayaran ke tabel pembayaran
ELSE

```

Tampilkan pesan kesalahan data pembayaran tidak memadai

END IF

Di dalam DFD terdapat 3 level yaitu [8]:

a. Diagram konteks

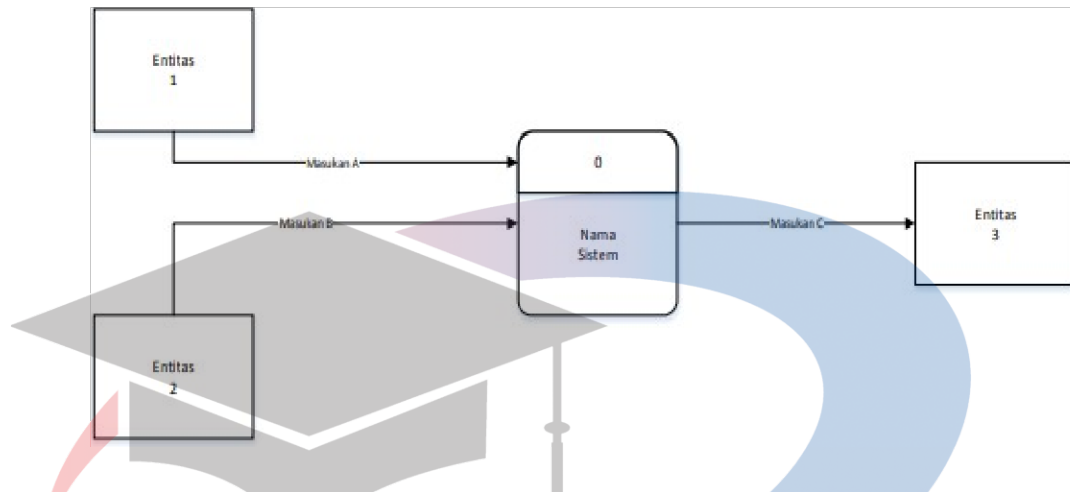
Diagram konteks adalah sebuah bagian level dari *Data Flow Diagram* yang digunakan untuk menetapkan konteks serta batasan-batasan sistem pada sebuah pemodelan. Menggambarkan satu lingkaran besar yang dapat mewakili seluruh proses yang biasanya diberi nomor nol. Semua entitas *eksternal* yang ditunjukkan pada diagram konteks berikut aliran-aliran data utama menuju dan dari sistem. Diagram ini sama sekali tidak memuat penyimpanan data dan tampak sederhana untuk diciptakan. Fungsi dari diagram konteks adalah untuk membatasi sistem dan menunjukkan adanya interaksi sistem dengan komponen diluar sistem, maka perlu dibuat diagram konteks yang merupakan gambaran sistem secara keseluruhan. Diagram konteks ini menggambarkan aliran pengolahan data keanggotaan, penjualan produk dan pembelian produk serta informasi dari sistem informasi pengolahan data penjualan dan pembelian produk. Dalam diagram konteks, tidak ada data store karena diagram konteks hanya menggambarkan hubungan antara sistem dengan entitas luar. Secara umum berikut aturan dalam diagram konteks:

- a. Menggambarkan hubungan sistem dengan entitas luar yang berhubungan langsung dengan sistem.
- b. Terdapat komponen sistem, entitas, dan aliran data.
- c. Aliran data dari entitas ke sistem maupun sebaliknya.
- d. Tidak boleh dituliskan aliran data dari entitas ke entitas langsung tanpa melaluis sistem, jika memang hal itu terjadi dalam proses yang ada maka tidak perlu digambarkan atau dituliskan pada diagram konteks.

Manfaat diagram konteks yaitu:

- a. Memperlihatkan ruang lingkup dan batas-batas suatu sistem termasuk sistem lain yang berinteraksi dengannya.
- b. Tidak ada pengetahuan teknis yang diperlukan untuk memahami diagram.

- c. Mudah menggambar dan mengubah karena notasi yang terbatas.
- d. Mudah diperluas dengan menambahkan berbagai level DFD.

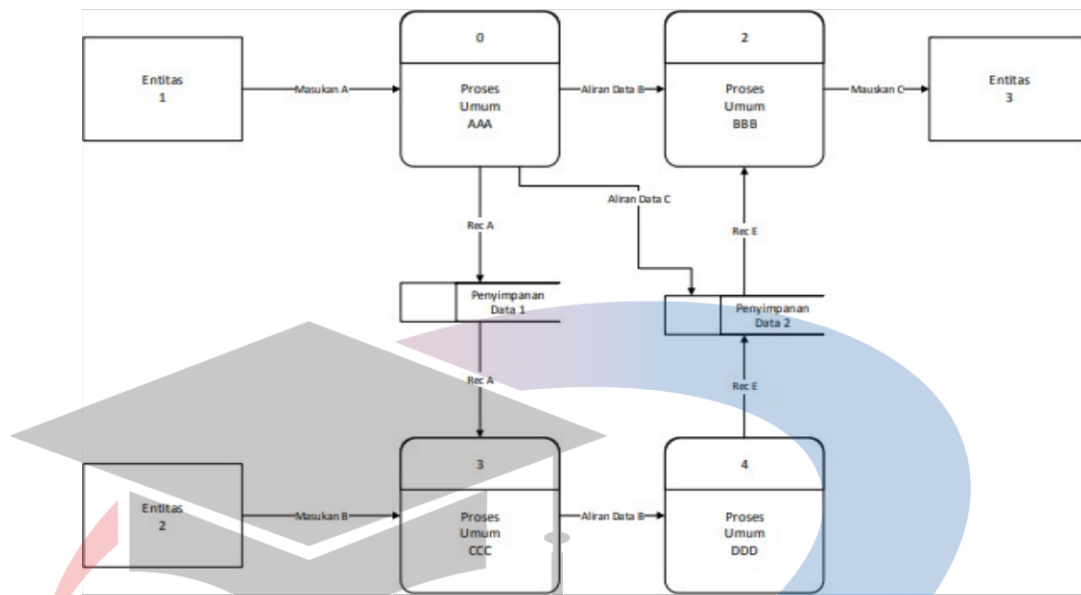


Gambar 2.3 Contoh Diagram Konteks

b. Diagram Nol

Diagram nol merupakan pemecahan dari diagram konteks ke diagram nol, dimana proses yang ada didalam sistem lebih rinci dan lengkap karena proses utama yang dipecah menjadi beberapa sub proses dengan fungsi masing-masing. Di dalam diagram ini memuat penyimpanan data. Diagram nol adalah pengembangan dari diagram konteks dan biasa mencakup sampai sembilan proses. Setiap proses diberi nomor bilangan bulat, umumnya dimulai dari sudut sebelah kiri atas diagram dan mengarah ke sudut sebelah kanan bawah. Penyimpanan data utama dari sistem dan semua entitas *eksternal* dimasukkan ke dalam diagram nol. Proses dalam pembuatan diagram nol yaitu :

- a. Tentukan proses utama yang ada pada sistem.
- b. Tentukan apa yang di terima masing-masing dari sistem dan memperhatikan konsep keseimbangan (alur data yang keluar/masuk dari suatu level/ diagram konteks harus sama dengan alur data yang keluar/masuk pada level berikutnya).
- c. Apabila diperlukan munculkan *data store* (master) sebagai sumber maupun tujuan alur data.
- d. Beri nomor pada proses utama.



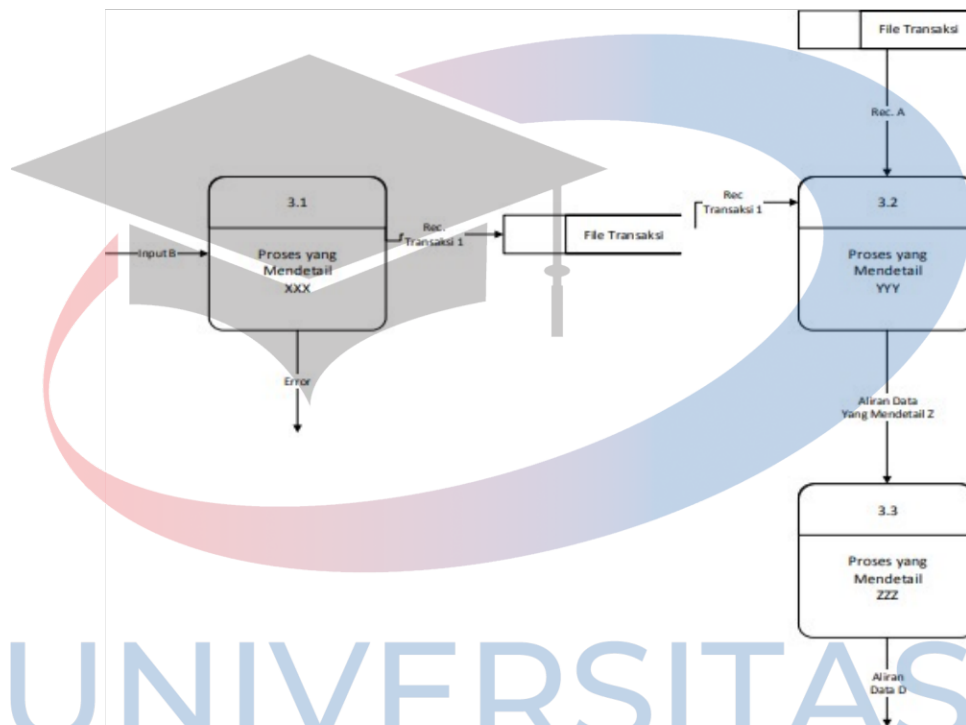
Gambar 2.4 Contoh Diagram Nol

c. Diagram Rinci (DFD level 2)

Diagram rinci adalah rincian proses dari DFD level 1 yang bertugas untuk menguraikan apa saja proses-proses yang ada dalam lingkup sistem. Setiap proses dalam diagram nol bisa dikembangkan untuk menciptakan diagram anak yang lebih mendetail. Proses pada diagram nol yang dikembangkan ini disebut *parent process* (proses induk) dan diagram yang dihasilkan disebut *child diagram* (diagram anak). Semua aliran data yang menuju atau keluar dari proses induk harus ditunjukkan mengalir ke dalam atau ke luar dari diagram anak. Bila proses induk memiliki aliran data yang terhubung ke penyimpanan data, diagram anak bisa memasukkan penyimpanan data tersebut. Selain itu, diagram pada level yang lebih rendah ini bisa memasukkan penyimpanan data yang tidak ditunjukkan dalam proses induk. Proses dalam pembuatan diagram anak atau diagram rinci ini sebagai berikut :

- a. Tentukan proses yang lebih kecil (sub-proses) dari proses utama.
- b. Tentukan apa yang diberikan/ diterima masing-masing sub-proses dari sistem dan perhatikan konsep keseimbangan.

- c. Apabila diperlukan munculkan *data store* (transaksi) sebagai sumber maupun tujuan alur data.
- d. Hindari perpotongan arus data.
- e. Beri nomor pada masing-masing sub-proses yang menunjukkan dekomposisi dari proses sebelumnya.



Gambar 2.5 Contoh Diagram Rinci

Beberapa simbol yang digunakan di DFD yaitu [9]:

1. *External entity* (kesatuan luar)

Setiap sistem pasti mempunyai batas sistem yang memisahkan suatu sistem *input* dan menghasilkan *output* kepada lingkungan luar. Entitas merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang memberikan *input* atau menerima *output* dari sistem.

2. *Data flow* (arus data)





Arus data ditandai dengan simbol berupa anak panah. Arus data ini mengalir diantara proses, simpanan data dan kesatuan luar. Arus data ini menunjukkan arus dari data yang berupa masukan untuk sistem atau hasil dari proses sistem.

3. *Process* (proses)

Suatu proses adalah kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer dari hasil suatu arus data yang masuk ke dalam proses untuk dihasilkan arus data yang akan keluar dari proses.

4. *Data store* (simpanan data)

Data Store merupakan simpanan dari data yang dapat berupa *file*, arsip, tabel dan lain-lain.

Simbol	Keterangan
	<i>External Entity</i> , merupakan kesatuan di lingkungan luar sistem yang bisa berupa orang, organisasi atau sistem lain.
	<i>Process</i> , merupakan proses seperti perhitungan aritmatik penulisan suatu formula atau pembuatan laporan
	<i>Data Store</i> (Simpan Data), dapat berupa suatu file atau database pada sistem komputer atau catatan manual
	<i>Data Flow</i> (arus data), arus data ini mengalir diantara proses, simpan data dan kesatuan luar

Gambar 2.6 Simbol Aliran Data

2.3.2 Kamus Data

Kamus data adalah suatu aplikasi khusus dari beberapa jenis kamus yang digunakan sebagai *referensi* kehidupan setiap hari. Kamus data merupakan hasil *referensi* mengenai data, suatu data yang disusun oleh penganalisis sistem untuk membimbing mereka selama melakukan analisis. Kamus data mengumpulkan dan mengkoordinasi istilah-istilah data tertentu, dan menjelaskan apa arti dari setiap istilah

yang ada. Proses penyusunan suatu kamus data bisa membantu analisis sistem mengkonseptualisasikan sistem dan cara kerjanya [3]. Sebagai tambahan untuk dokumentasi serta mengurangi redudansi, kamus data bisa digunakan untuk [3]:

1. Memvalidasi diagram aliran data dalam hal kelengkapan dan keakuratan.
2. Menyediakan suatu titik awal untuk mengembangkan layar dan laporan-laporan.
3. Menentukan muatan data yang disimpan dalam *file-file*.
4. Mengembangkan logika untuk proses-proses diagram aliran data. Notasi aljabar menggunakan simbol-simbol sebagai berikut:
 1. Tanda sama dengan (=), artinya “terdiri dari”.
 2. Tanda plus (+), artinya “dan”.
 3. Tanda kurung {}, menunjukan elemen-elemen *repetitive*, juga disebut kelompok berulang didalam kelompok tersebut. Kelompok berulang bisa mengandung keadaan-keadaan tertentu, seperti misalnya, jumlah pengulangan yang pasti atau batas tertinggi dan batas terendah untuk jumlah pengulangan.
 4. Tanda kurung [], menunjukan salah satu dari dua situasi tertentu. Satu elemen bisa ada sedangkan elemen lainnya juga ada, tetapi tidak bisa kedua-duanya ada secara bersamaan. Elemen-elemen yang ada didalam tanda kurung ini saling terpisah satu sama lain.
 5. Tanda kurung (), menunjukan suatu elemen yang bersifat pilihan. Elemen-elemen yang bersifat pilihan ini bisa dikosongkan pada layar masukan atau bisa juga dengan memuat spasi atau nol untuk field-field numeric pada struktur *file*.

Kategori kamus data terdiri atas [3]:

1. Aliran Data

Aliran data biasanya merupakan komponen pertama yang harus ditetapkan. Masukan dan keluaran sistem ditentukan dari wawancara, observasi terhadap pengguna, dan menganalisis dokumen – dokumesn dari sistem yang sudah ada. Informasi yang ditangkap untuks setiap aliran data bisa diringkas menggunakan sebuah formulir yang membuat informasi – informasi berikut:

- a. Identitas, nomor identifikasi yang bersifat pilihan. Kadang – kadang identitas dikodekan menggunakan suatu skema untuk mengidentifikasi sistem dan aplikasi didalam suatu sistem.
- b. Nama deskriptif unik untuk aliran data. Nama ini menggunakan teks yang harus muncul pada diagram dan bisa direferensikan dalam semua deskripsi yang menggunakan aliran data.
- c. Deskripsi umum aliran data.
- d. Sumber aliran data. Sumber bisa berupa entitas eksternal, yakni: proses atau aliran data yang datang dari suatu simpanan data.
- e. Tujuan aliran data (item – item yang sama disebutkan dibawah sumber).
- f. Indikasi mengenai apakah aliran data merupakan record yang memasuki atau meninggalkan file dan *record* yang memuat laporan, formulir, atau layar. Bila aliran data memuat data yang digunakan diantara proses – proses maka ditandai sebagai internal.
- g. Nama struktur data yang menggambarkan elemen – elemen yang ditentukan dalam aliran data ini. Untuk aliran data sederhana, bisa berupa satu atau beberapa elemen.
- h. Ukuran per satuan waktu. Data – data bisa berupa *record* per hari atau satuan – satuan waktu lainnya.
- i. Area untuk komentar dan catatan – catatan lainnya mengenai aliran data.

2. Struktur Data

Struktur data biasanya digambarkan dengan menggunakan notasi aljabar. Metode ini memungkinkan penganalisis membuat suatu gambaran mengenai elemen – elemen yang membentuk struktur data bersama dengan informasi terkait elemen tersebut.

3. Elemen – Elemen Data

masing – masing elemen data bisa ditetapkan sekali dalam kamus data dan juga dimasukkan sebelumnya pada formulir deskripsi elemen.

Contoh kamus data [9]:

1. Kamus data yang berasal dari tempat penyimpanan (*Data Store*) dapat didefinisikan seperti dibawah ini:
 - a. Produk = kode_pdk + nama_pdk + satuan + kuantitas_brg + hrg_beli + hrg_jual + jlh_stok
 - b. Pembayaran = no_faktur + tanggal + jlh_byr
 - c. Penjualan = no_faktur + kode_brg + kuantitas_pdk + hrg_jual
 - d. *Supplier* = kode_pms + nama_pms + alamat + kota + telepon
2. Kamus data yang berasal dari arus data (*Data Flow*) dapat didefinisikan seperti dibawah ini:
 - a. Bulan = no_bulan + tahun
 - b. Bukti_jual = no_faktur + tanggal + {nama_pdk + hrg_jual + kuantitas + jumlah} + total
 - c. Id_produk = kode_pdk + nama_pdk + satuan + jenis + hrg_jual + jml_stok + kode_pms
 - d. Id_pdk_terjual = kode_pdk + kuantitas

2.3.3 Basis Data

Basis data tidak hanya merupakan kumpulan *file*. Lebih dari itu, basis data adalah pusat sumber data yang caranya dipakai oleh banyak pemakai untuk diberbagai aplikasi. Inti dari basis data adalah *Database Management System* (DBMS), yang membolehkan pembuatan, modifikasi, dan pembaharuan basis data; mendapatkan kembali data; dan menghasilkan laporan [3].

Tujuan basis data yang efektif yaitu [3]:

- a. Memastikan bahwa data dapat dipakai diantara pemakai untuk berbagai aplikasi.
- b. Memelihara data baik keakuratan maupun konsistensinya.
- c. Memastikan bahwa semua data yang diperlukan untuk aplikasi sekarang dan yang akan datang disediakan dengan cepat.
- d. Membolehkan basis data dan kebutuhan pemakai untuk berkembang.

- e. Membolehkan pemakai untuk membangun pandangan personalnya tentang data tanpa memperhatikan cara data disimpan secara fisik.

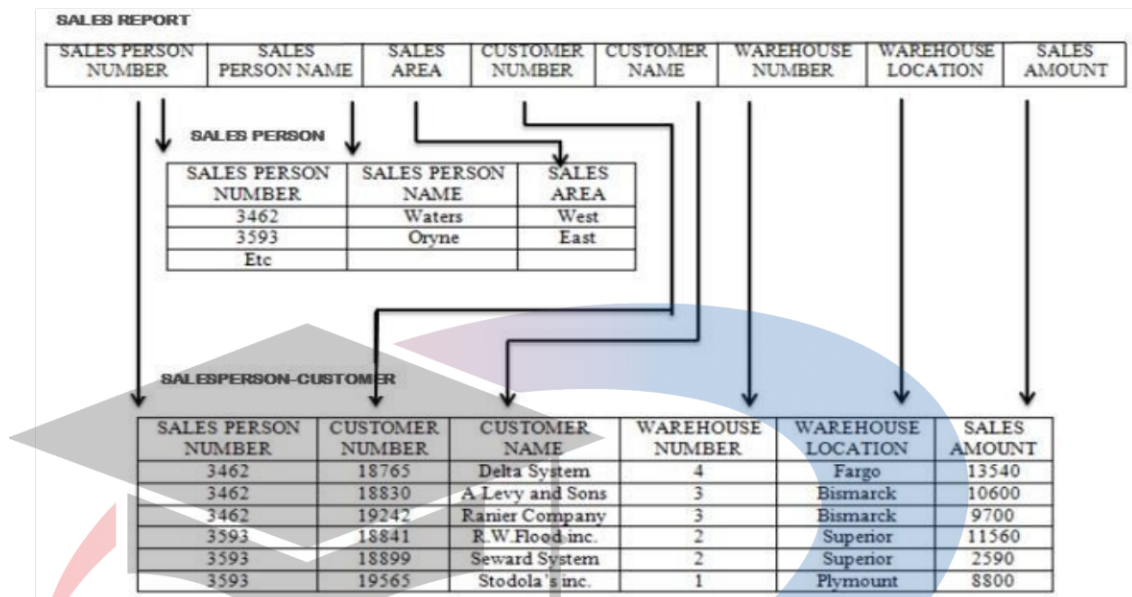
Tujuan yang telah disebutkan diatas memberikan keuntungan dan kerugian pendekatan basis data. Pertama, pemakaian data berarti bahwa data perlu disimpan hanya sekali. Membantu mencapai integritas data, karena mengubah data yang diselesaikan lebih mudah dan dapat dipercaya jika data muncul hanya sekali dalam banyak *file* yang berbeda. Keuntungan pendekatan basis data membolehkan pemakai untuk memiliki pandangan sendiri mengenai data. Pemakai tidak perlu memperhatikan struktur sebenarnya basis data atau penyimpanan fisiknya. Sedangkan kerugian pendekatan basis data adalah bahwa semua data disimpan dalam satu tempat. Oleh karena itu, data lebih mudah diserang bencana dan membutuhkan backup yang lengkap [3].

2.3.4 Normalisasi

Normalisasi diartikan sebagai suatu teknik yang menstrukturkan data dalam cara-cara tertentu untuk mencegah timbulnya permasalahan pengolahan data dalam basis data. Proses normalisasi merupakan proses pengelompokkan elemen data menjadi tabel-tabel yang menunjukkan entitas dan relasinya. Pada proses normalisasi selalu diuji pada beberapa kondisi, apakah ada kesulitan pada saat menambah (*insert*), menghapus (*delete*), mengubah (*update*), dan membaca (*retrieve*) pada suatu basis data. Bila ada kesulitan pada pengujian tersebut, maka relasi tersebut dipecahkan menjadi beberapa tabel lagi, sehingga diperoleh *database* yang optimal [4]. Pada proses normalisasi ini perlu dikenal terlebih dahulu mengenai definisi dari tahap-tahap normalisasi. Tahap-tahap normalisasi terdiri dari [3]:

1. Bentuk normal pertama (1NF)

Langkah pertama dalam normalisasi hubungan adalah untuk menghilangkan kelompok berulang. Dalam contoh ini laporan penjualan dinormalisasikan dengan pemisahan hubungan ke dalam dua hubungan baru. Hubungan baru ini akan dinamai tenaga penjualan dan *customer* tenaga penjualan.

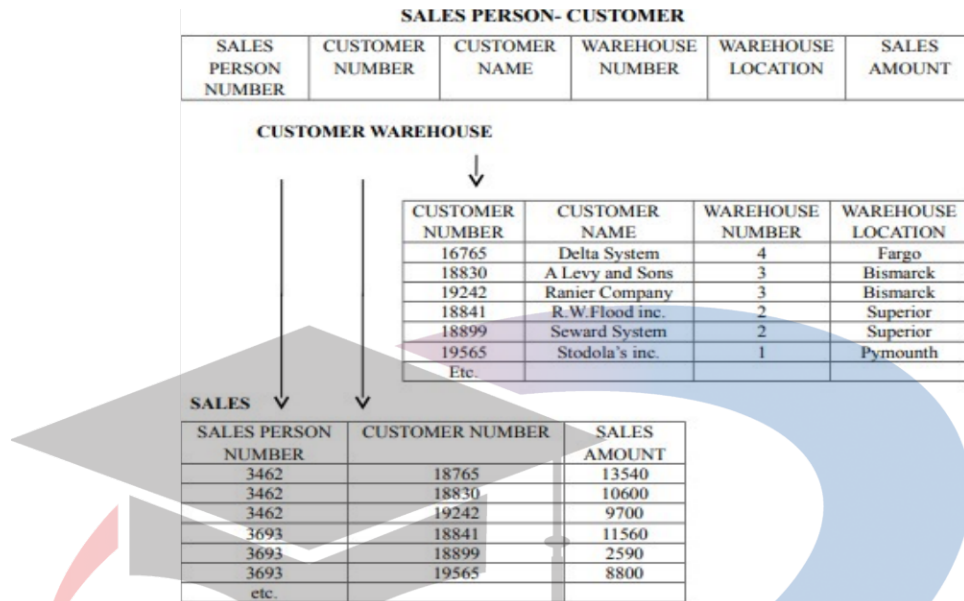


Gambar 2.7 Contoh Normalisasi Pertama

2. Bentuk normal kedua (2NF)

Dalam bentuk normalisasi kedua, semua atribut akan secara fungsional tergantung pada kunci utama. Oleh karena itu, langkah selanjutnya adalah menghilangkan semua atribut yang tergantung sebagian dan meletakkannya dalam hubungan lain.

UNIVERSITAS
MIKROSKIL

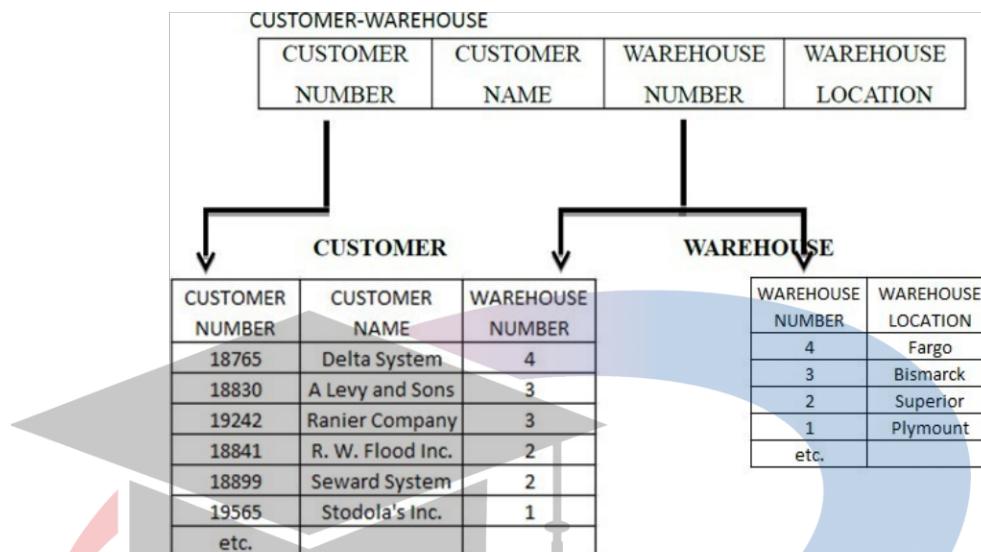


Gambar 2.8 Contoh Normalisasi Kedua

3. Bentuk normal ketiga (3NF)

Hubungan dinormalisasi dalam bentuk normalisasi ketiga jika semua atribut bukan kunci sepenuhnya tergantung secara fungsional pada kunci utama dan tidak terdapat ketergantungan transitif (bukan kunci). Dalam cara yang sama dengan langkah sebelumnya, dimungkinkan untuk memecah hubungan gudang *customer* ke dalam dua hubungan [3].

UNIVERSITAS MIKROSKIL



Gambar 2.9 Contoh Normalisasi Ketiga

2.4 Penjualan

Penjualan adalah total jumlah yang dibebankan kepada *customer* atas produk yang terjual, baik secara tunai maupun kredit. Baik retur dan potongan penjualan maupun diskon penjualan dikurangkan dari penjualan untuk mendapatkan penjualan bersih. Penjualan merupakan sumber hidup dari suatu perusahaan, karena dari penjualan dapat diperoleh laba serta suatu usaha memikat konsumen yang diusahakan untuk mengetahui daya tarik mereka sehingga mengetahui produk yang dihasilkan [6].

2.5 Retur Penjualan

Transaksi retur penjualan terjadi jika perusahaan menerima pengembalian produk dari *customer*. Pengembalian produk oleh *customer* harus diotorisasi oleh fungsi penjualan dan diterima oleh fungsi penerimaan [3].

Fungsi yang terkait dalam transaksi retur penjualan adalah [3]:

a. Fungsi penjualan

- b. Fungsi penerimaan
- c. Fungsi gudang
- d. Fungsi akuntansi

Dalam transaksi retur penjualan, fungsi penjualan bertanggung jawab atas penerimaan pemberitahuan mengenai pengembalian produk yang telah dibeli oleh *customer*. Otorisasi penerimaan kembali produk yang telah dijual tersebut dilakukan dengan cara membuat memo kredit yang dikirimkan kepada fungsi penerimaan [3].

2.6 Pembelian

Pembelian digunakan dalam perusahaan untuk pengadaan produk yang diperlukan oleh perusahaan. Fungsi pembelian bertanggung jawab untuk memperoleh informasi mengenai harga produk, menentukan *supplier* yang dipilih dalam pengadaan produk, dan mengeluarkan pesanan pembelian kepada *supplier* yang dipilih [10].

Secara garis besar transaksi pembelian mencakup prosedur berikut ini [10]:

- a. Fungsi gudang mengajukan permintaan pembelian ke fungsi pembelian.
- b. Fungsi pembelian meminta penawaran harga dari berbagai *supplier*.
- c. Fungsi pembelian menerima penawaran harga dari berbagai *supplier* dan melakukan pemilihan *supplier*.
- d. Fungsi pembelian membuat pesanan pembelian kepada *supplier* yang telah dipilih.
- e. Fungsi penerimaan memeriksa dan menerima produk yang dikirim oleh *supplier*.
- f. Fungsi penerimaan menyerahkan produk yang diterima kepada fungsi gudang untuk disimpan.
- g. Fungsi penerimaan untuk melaporkan penerimaan produk kepada fungsi akuntansi.
- h. Fungsi akuntansi menerima faktur tagihan dari *supplier* atas dasar faktur dari *supplier* tersebut, fungsi akuntansi mencatat kewajiban yang timbul dari transaksi pembelian.

2.7 Retur Pembelian

Produk yang sudah diterima dari *supplier* terkadang tidak sesuai dengan produk yang dipesan menurut surat pesanan pembelian. Ketidaksesuaian tersebut terjadi kemungkinan karena produk yang diterima tidak cocok dengan spesifikasi yang tercantum dalam surat pesanan pembelian, produk mengalami kerusakan dalam pengiriman, atau produk diterima melewati tanggal pengiriman yang dijanjikan oleh *supplier*. Sistem retur pembelian digunakan perusahaan untuk mengembalikan produk yang sudah dibeli kepada *supplier* [3].

Fungsi terkait dalam sistem retur pembelian adalah:

- a. Fungsi gudang
- b. Fungsi pembelian
- c. Fungsi pengiriman
- d. Fungsi akuntansi

Retur pembelian digunakan untuk mencatat transaksi pembelian digunakan untuk mencatat transaksi retur pembelian mengurangi jumlah persediaan dan utang dagang. Jika perusahaan tidak melakukan jurnal khusus karena rendahnya frekuensi transaksi retur pembelian, perusahaan menggunakan jurnal umum untuk mencatat transaksi tersebut [3].

2.8 Persediaan

Pengertian persediaan (*inventory cost*) merupakan sejumlah bahan / produk yang disediakan oleh perusahaan, baik berupa produk jadi, bahan mentah, maupun produk dalam proses yang disediakan untuk menjaga kelancaran operasi perusahaan guna memenuhi permintaan konsumen setiap waktu. Pengertian manajemen persediaan (*Inventory Management*) kegiatan untuk menentukan jumlah dan komposisi persediaan sehingga perusahaan dapat melindungi kelancaran produksi dan penjualan serta kebutuhan – kebutuhan pembelanjaan perusahaan dengan efektif dan efisien, juga termasuk pengaturan dan pengawasan atas pelaksanaan pengadaan bahan / produk yang

diperlukan sesuai dengan jumlah dan waktu yang dibutuhkan dengan biaya yang serendah – rendahnya [2].

