

BAB II

TINJAUAN PUSTAKA

2.1 Konsep Sistem Informasi

2.1.1 Sistem

Sistem merupakan suatu bentuk integrasi antara satu komponen dengan komponen lain karena sistem memiliki sasaran yang berbeda untuk setiap kasus yang terjadi yang ada di dalam sistem tersebut.[1]

Model umum sebuah sistem adalah input, proses, dan output. Hal ini merupakan konsep sebuah sistem yang sangat sederhana sebab sebuah sistem dapat mempunyai beberapa masukan dan keluaran. Selain itu, sebuah sistem mempunyai karakteristik atau sifat-sifat tertentu yang mencirikan bahwa hal tersebut bisa dikatakan sebagai suatu sistem. Adapun karakteristik yang dimaksud adalah sebagai berikut :[1]

a. Komponen Sistem (*Components*)

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem, Setiap subsistem memiliki sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai sistem yang lebih besar atau sering disebut "supra sistem".

b. Batasan Sistem (*Boundary*)

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

c. Lingkungan Luar Sistem (*Environment*)

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem ini dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. Dengan demikian, lingkungan luar tersebut harus tetap dijaga dan dipelihara,

Lingkungan luar yang merugikan harus dikendalikan. Kalau tidak, maka akan mengganggu kelangsungan hidup sistem tersebut.

d. Penghubung Sistem (*interface*)

Media yang menghubungkan sistem dengan subsistem lain disebut penghubung sistem atau interface. Penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lain. Bentuk keluaran dari satu subsistem akan menjadi masukan untuk subsistem lain melalui penghubung tersebut. Dengan demikian, dapat terjadi suatu integrasi sistem yang membentuk satu kesatuan.

e. Masukan Sistem (*Input*)

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*). Contoh, di dalam suatu unit sistem komputer, "program" adalah maintenance input yang digunakan untuk mengoperasikan komputernya dan "data" adalah signal input untuk diolah menjadi informasi.

f. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi sub sistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini dapat digunakan sebagai masukan untuk pengambilan Keputusan atau hal-hal lain yang menjadi input bagi subsistem lain.

g. Pengolah Sistem (*Proses*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran, contohnya adalah sistem akuntansi. Sistem ini akan mengolah data transaksi menjadi laporan-laporan yang dibutuhkan oleh pihak manajemen.

h. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti. Kalau suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan.

2.1.2 Informasi

Informasi adalah data yang telah diklasifikasi atau diinterpretasi untuk digunakan dalam proses pengambilan Keputusan, Sistem pengolahan informasi

mengolah data menjadi informasi atau tepatnya mengolah data dari bentuk tak berguna menjadi berguna bagi penerimanya. Nilai informasi berhubungan dengan Keputusan maka informasi menjadi tidak diperlukan keputusan dapat berkisar dari keputusan berulang sederhana sampai keputusan strategis jangka panjang.[1]

Sumber informasi adalah data. Data merupakan kenyataan yang menggambarkan suatu kejadian serta merupakan suatu bentuk yang masih mentah yang belum dapat bercerita banyak sehingga perlu diolah lebih lanjut melalui suatu model untuk menghasilkan informasi.[1]

2.1.3 Sistem informasi

Sistem informasi merupakan suatu kombinasi teratur dari orang-orang, *hardware*, *software*, jaringan komunikasi dan sumber daya data yang mengumpulkan, mengubah dan menyebarkan informasi dalam sebuah organisasi.[2]

Fungsi sistem informasi sebagai berikut :[2]

1. Untuk meningkatkan aksesibilitas data yang ada secara efektif dan efisien kepada pengguna, tanpa dengan perantara sistem informasi.
2. Memperbaiki produktivitas aplikasi pengembangan dan pemeliharaan sistem.
3. Menjamin tersedianya kualitas dan keterampilan dalam memanfaatkan sistem informasi secara kritis.
4. Mengidentifikasi kebutuhan mengenai keterampilan pendukung sistem informasi.
5. Mengantisipasi dan memahami akan konsekuensi ekonomi.
6. Menetapkan investasi yang akan diarahkan pada sistem informasi.
7. Mengembangkan proses perencanaan yang efektif.

Komponen-komponen dari sistem informasi adalah sebagai berikut :[2]

1. Komponen input, adalah data yang masuk ke dalam sistem informasi
2. Komponen model, adalah kombinasi prosedur, logika dan model matematika yang memproses data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.
3. Komponen output, adalah hasil informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.
4. Komponen teknologi, adalah alat dalam sistem informasi, teknologi digunakan dalam menerima *input*, menjalankan model, menyimpan dan mengakses data,

menghasilkan dan mengirimkan output dan memantau pengendalian sistem.

5. Komponen basis data, adalah kumpulan data yang saling berhubungan yang tersimpan di dalam komputer dengan menggunakan *software database*.

6. Komponen kontrol, adalah komponen yang mengendalikan gangguan terhadap sistem informasi.









Ciri-ciri sistem informasi :[2]

1. Baru, adalah informasi yang didapat sama sekali barudan segar bagi penerima
2. Tambahan, adalah informasi dapat diperbarui ataumemberikan tambahan terhadap informasi yangsebelumnya telah ada.
3. Kolektif, adalah informasi yang dapat menjadi suatukoreksi dari informasi yang salah sebelumnya.
4. Penegas, adalah informasi yang dapat mempertegasinformasi yang telah ada

2.2 Teknik Perancangan Sistem

2.2.1 Data Flow Diagram (DFD)

Data Flow Diagram adalah alat pembuatan model yang memungkinkan professional sistem untuk menggambarkan sistem sebagai suatu jaringan proses fungsional yang dihubungkan satu sama lain dengan alur data, baik secara manual maupun terkomputerisasi. Simbol-simbol DFD ditunjukkan pada gambar berikut :[3]

Keterangan	DeMarco and Yourdan Simbol	Gane and Sarson Simbol
Entitas Luar		
Proses		
Aliran data (data flow)		
Simpan data		

Gambar 2.1 Simbol-simbol DFD

Penjelasan simbol-simbol di DFD :[4]

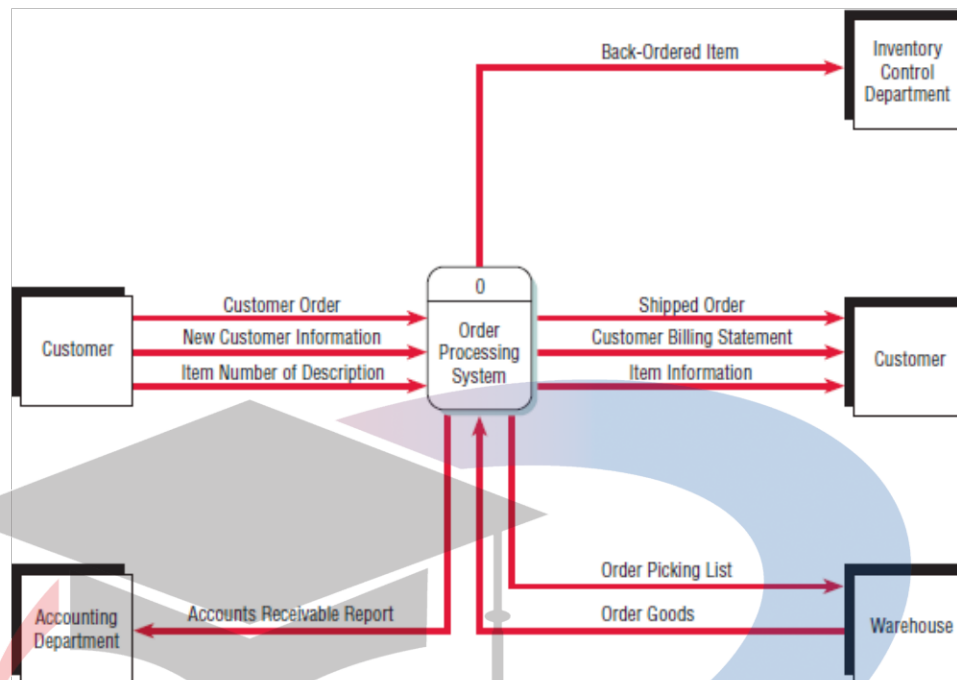
1. Kotak rangkap dua digunakan untuk menggambarkan suatu entitas eksternal (bagian lain sebuah perusahaan, seseorang, atau sebuah mesin) yang dapat mengirim data atau menerima data dari sistem. Entitas eksternal, atau hanya entitas, disebut juga sumber atau tujuan data dan dianggap eksternal terhadap sistem yang sedang digambarkan. Setiap entitas diberi label dengan sebuah nama yang sesuai. Meskipun berinteraksi dengan sistem, namun dianggap di luar batas-batas sistem. Entitas-entitas tersebut harus diberi nama dengan suatu kata benda. Entitas yang sama bisa digunakan lebih dari sekali atas suatu diagram aliran data tertentu untuk menghindari persilangan antara jalur-jalur aliran data.
2. Arus data pada *flow diagram* diberi simbol suatu panah. Karena sebuah tanda panah menunjukkan seseorang, tempat, atau sesuatu, maka harus digambarkan dalam kata benda. Arus data ini menunjukkan arus dari data yang bisa berupa masukan untuk sistem atau hasil dari proses suatu sistem. Arus data harus diberi nama yang jelas dan mempunyai arti, dimana nama dari arus data dituliskan diatas garis panahnya.

3. Proses adalah suatu kejadian kegiatan atau kerja yang dilakukan oleh orang, mesin atau komputer. Proses berfungsi untuk mengolah arus data yang masuk kedalamnya (*input*), kemudian dari proses itu juga menghasilkan arus data keluar (*output*). Untuk proses sebaiknya menggunakan nama yang mengacu pada fungsi, yaitu gabungan antara kata kerja yang spesifik dan objeknya.
4. Penyimpanan data (*data store*) dapat berupa suatu *file* atau *database* pada sistem komputer, arsip atau catatan manual, kotak tempat data, tabel acuan manual suatu agenda atau buku.

Dalam pembuatan DFD dibagi menjadi 3 tahapan proses yaitu :[5]

1. Diagram Konteks

Diagram konteks adalah tingkatan tertinggi dalam diagram aliran data dan hanya memuat satu proses, menunjukkan sistem secara keseluruhan. Proses tersebut diberi nama nol. Semua entitas eksternal yang ditunjukkan pada diagram konteks berikut aliran data utama menuju dan dari sistem. Diagram tersebut tidak memuat penyimpanan data dan tampak sederhana untuk diciptakan, begitu entitas-entitas eksternal serta aliran data menuju dan dari sistem diketahui penganalisis dari wawancara dengan pengguna dan sebagai hasil analisis dokumen. Berikut ini adalah contoh penggambaran diagram konteks.

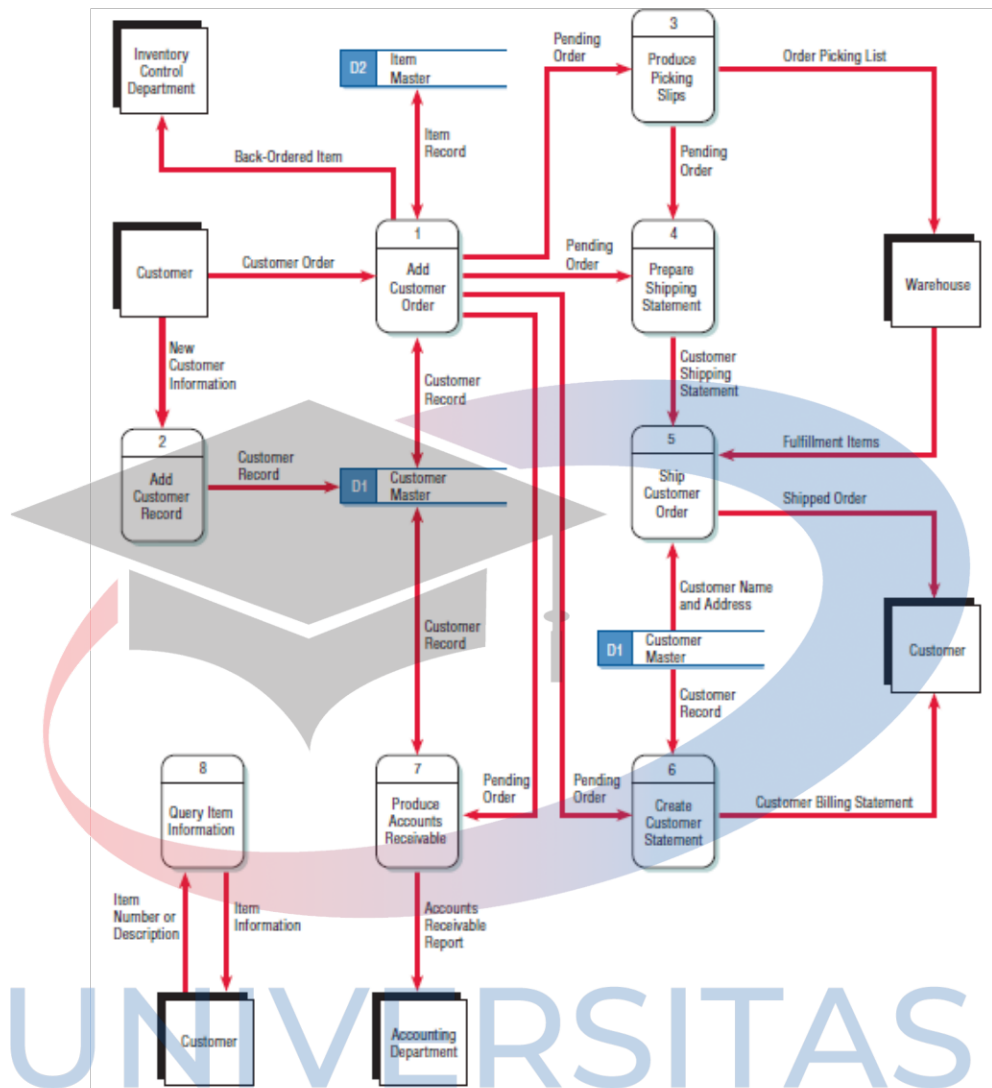


Gambar 2.2 Contoh Penggambaran Diagram Konteks

2. Diagram 0 (Level Berikutnya)

Lebih mendetail dibandingkan diagram konteks yang diperbolehkan, bisa dicapai dengan “mengembangkan diagram”. Masukan dan keluaran yang ditetapkan dalam diagram yang pertama tetap konstan dalam gambar terperinci yang melibatkan tiga sampai Sembilan proses dan menunjukkan penyimpanan data dan aliran data baru pada level yang lebih rendah.

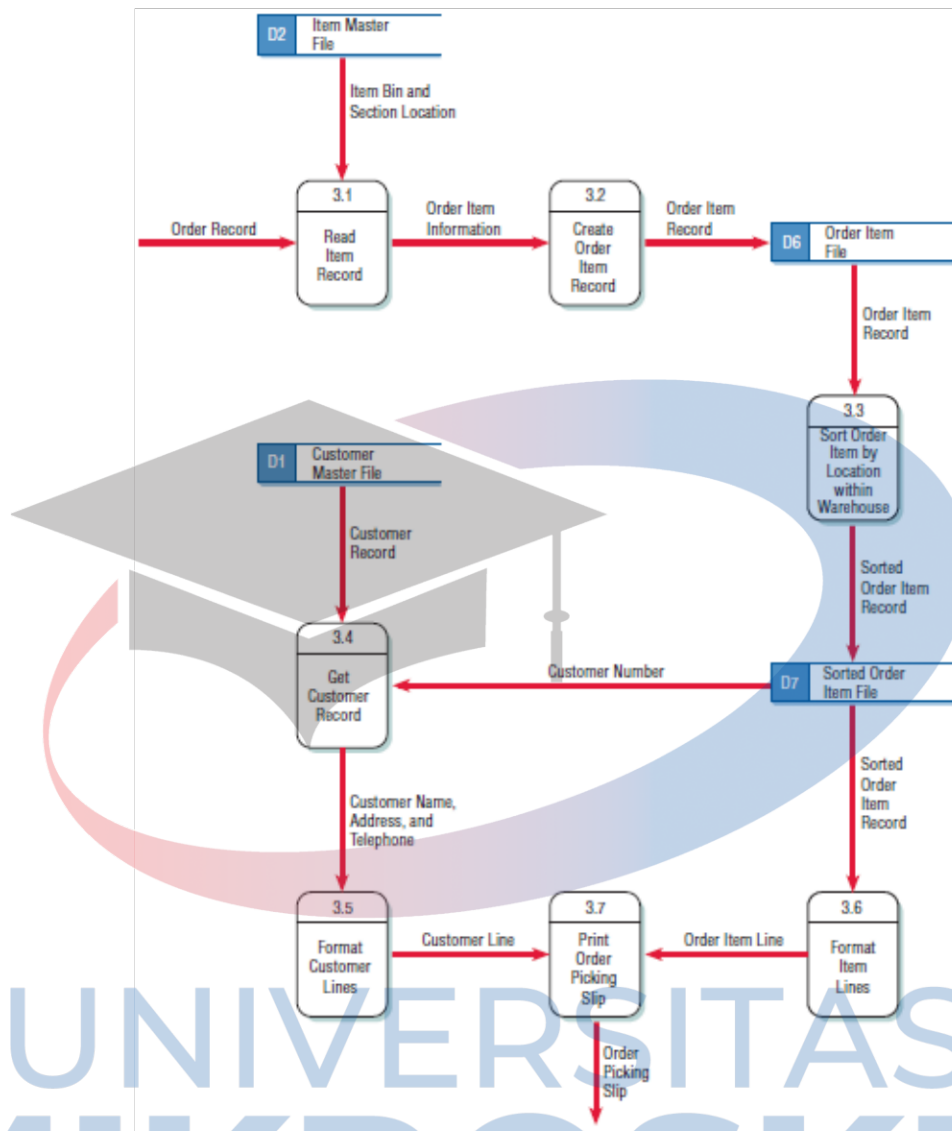
Diagram nol adalah pengembangan diagram konteks dan bisa mencakup sampai Sembilan proses. Memasukkan lebih banyak proses pada level ini akan terjadi dalam suatu diagram yang kacau dan sulit dipahami. Setiap proses diberi nomor bilangan bulat, umumnya dimulai dari sebelah kiri atas diagram dan mengarah ke sudut sebelah kanan bawah. Berikut ini adalah contoh penggambaran diagram 0.



Gambar 2.3 Contoh Penggambaran Diagram 0

3. Diagram Anak (Tingkat yang Lebih Mendetail)

Setiap proses dalam diagram nol dikembangkan untuk menciptakan diagram anak yang lebih mendetail. Proses pada diagram nol yang dikembangkan itu disebut *parent process* (proses induk) dan diagram yang dihasilkan disebut *child diagram* (diagram anak). Aturan utama untuk menciptakan diagram anak, keseimbangan vertical, menyatakan bahwa suatu diagram anak tidak bisa menghasilkan suatu keluaran atau menerima masukan dimana proses induknya juga tidak menghasilkan atau tidak menerima. Semua aliran data yang menuju atau keluar dari proses induk harus ditunjukkan mengalir ke dalam atau keluar dari diagram anak. Berikut ini adalah contoh penggambaran diagram anak.



Gambar 2.4 Penggambaran Diagram Anak

Diagram aliran data dapat dikategorikan menjadi dua model yaitu : Model Diagram aliran data *logika* dan Model Diagram aliran data *fisik*. Diagram aliran data *logika* memfokuskan pada bisnis serta bagaimana bisnis tersebut beroperasi dan tidak berhubungan dengan bagaimana sistem tersebut dibangun. Melainkan, menggambarkan peristiwa – peristiwa bisnis yang dilakukan serta data – data yang diperlukan dan dihasilkan setiap peristiwa tersebut. Sebaliknya, Diagram aliran data *fisik* menunjukkan bagaimana sistem tersebut akan diimplementasikan, termasuk perangkat keras, perangkat lunak, file – file dan orang – orang yang

terlibat dalam sistem. Jadi kesimpulannya bahwa model *logika* merefleksikan bisnis sedangkan model *fisik* menggambarkan sistem.

Beberapa kesalahan dalam pembuatan DFD antara lain sebagai berikut:[6]

1. Proses mempunyai *input* tetapi tidak menghasilkan *output*. Kesalahan ini disebut dengan *black hole* (lubang hitam), karena data masuk ke dalam proses dan lenyap tidak berbekas seperti dimasukkan ke dalam lubang hitam.
2. Proses menghasilkan *output* tetapi tidak pernah menerima *input*. Kesalahan ini disebut dengan *miracle* (ajaib) karena ajaib dihasilkan *output* tanpa pernah menerima *input*.
3. *Input* yang masuk tidak sesuai dengan kebutuhan proses.
4. *Data Store* tidak memiliki keluaran.
5. *Data Store* tidak memiliki masukan.
6. Hubungan langsung antar entitas luar.
7. Masukan langsung entitas *data store*.
8. Keluaran langsung dari *data store* ke entitas luar.
9. Hubungan langsung antar *data store*.
10. Data masukan dan keluaran yang tidak bersesuaian dalam *datastore*.

2.2.2 Kamus Data

Kamus data (*data dictionary*) berisi katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Analisis sistem dapat mendefinisikan setiap data yang mengalir pada sistem secara lengkap. Kamus data biasanya dibuat pada tahap analisis dan perancangan, pada tahap analisis digunakan sebagai alat komunikasi antara analisis sistem dengan pemakai sistem, sedangkan pada tahap perancangan digunakan untuk merancang *input*, *file/database* dan *output*. Kamus data dibuat berdasarkan arus data yang mengalir pada DFD (fisik dan logika), dimana di dalamnya terdapat struktur dari arus data secara detail.[7]

Kamus data berisi tentang : [7]

1. Nama arus data dan penyimpanan data dalam DFD (menjelaskan komposisi paket data dan komposisi penyimpanan data). Kamus data dibuat berdasarkan arus data

dan penyimpanan data, maka nama arus data dan penyimpanan data juga harus terdapat di dalam kamus data, sehingga bila pemakai/*user* membaca dan ingin lebih tahu mengenai lebih lanjut tentang suatu arus data atau penyimpanan data dapat mencarinya langsung di dalam kamus data.

2. Alias/nama lain, karena ada data yang sama tetapi mempunyai nama berbeda. Sebagai contoh bagian administrasi menyebut data kesehatan, sedangkan dokter menyebutnya dengan data diagnosa.
3. Bentuk data, bisa berupa formulir, hasil cetakan komputer/kertas, variabel, parameter dan *field* yang dapat digunakan untuk merancang *database*.
4. Arus data, menunjukkan dari mana data mengalir dan kemana data akan menuju. Keterangan ini perlu dicatat dalam kamus data agar mudah untuk mencarinya dalam DFD.
5. Keterangan, untuk lebih memperjelas makna dari arus data. Sebagai contoh data kesehatan pasien dipakai dokter untuk mengetahui riwayat penyakit para pasiennya.
6. Periode, untuk menunjukkan kapan terjadinya arus data ini. Kapan input data diberikan, kapan proses dilakukan dan kapan laporan-laporan dihasilkan.

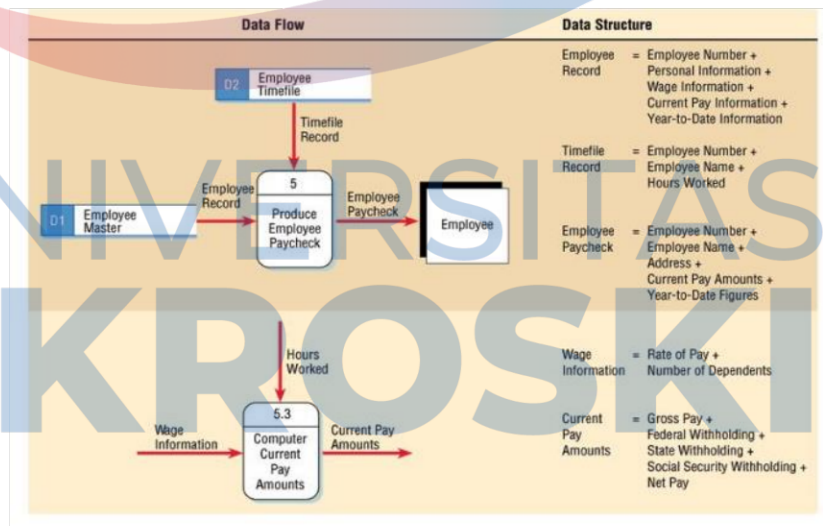
Kamus data mendefinisikan elemen data dengan fungsi sebagai berikut : [8]

1. Menjelaskan arti aliran data dan penyimpanan data dalam DFD
2. Mendeskripsikan komposisi paket data yang bergerak melalui aliran (misalnya alamat diuraikan menjadi kota, negara dan kode pos)
3. Mendeskripsikan komposisi penyimpanan data.
4. Menspesifikasikan nilai dan satuan yang relevan bagi penyimpanan dan aliran.
5. Mendeskripsikan hubungan detail antar penyimpanan (yang akan menjadi titik perhatian dalam *entity-relationship diagram*).

Notasi yang umum digunakan dalam menganalisis system dengan menggunakan sejumlah symbol yaitu : [8]

1. (=) : Terdiri dari, mendefinisikan, diuraikan menjadi
2. (+) : Dan

3. (): Menunjukkan suatu elemen yang bersifat pilihan (opsional). 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*.
4. ({}):Menunjukkan elemen-elemen repetitif, juga disebut kelompok berulang atau tabel-tabel. Kemungkinan bisa ada satu atau beberapa elemen berulang di dalam kelompok tersebut. Kelompok berulang bisa mengandung keadaan-keadaan tertentu, seperti misalnya, jumlah pengulangan yang pasti atau batas tertinggi dan batas terendah untuk jumlah pengulangan.
5. ([]):Menunjukkan salah satu dari dua situasi tertentu. Satu elemen bisa ada sedangkan elemen lainnya juga ada, tetapi tidak bisa kedua-duanya ada seara bersamaan. Elemen-elemen yang ada di dalam tanda kurung ini saling terpisah satu sama lain. (dengan kata lain, memilih salah satu dari sejumlah alternatif, seleksi)
6. (|): Pemisah sejumlah alternatif pilihan antara simbol []
7. (@): Identifikasi atribut kunci
8. (**): Komentar



Gambar 2.5 Penggunaan Struktur Data

2.2.3 User Interface

User Interface adalah penghubung antara manusia sebagai pengguna (*user*) dengan *software* pada perangkat (*interface*). Tidak hanya harus menarik, tampilan UI juga harus disesuaikan dengan berbagai perangkat agar pengguna mudah dan nyaman menggunakan websitenya.[9]

Pentingnya UI adalah untuk:

1. Memudahkan Interaksi pengguna dengan produk

Pada dasarnya, desain UI merupakan elemen visual sistem produk. Hal itu termasuk desain layar, tombol, ikon, gambar, teks, dan semua elemen visual lainnya berperan sebagai jembatan produk dengan pengguna

2. Meningkatkan Penjualan dan Pertumbuhan Bisnis

Tujuan utama setiap bisnis tentu meningkatkan penjualan dan pertumbuhan bisnis. Salah satu cara meningkatkan pertumbuhan tersebut adalah dengan mengembangkan UI/UX. Desain UI dapat meningkatkan pengalaman dan kepuasan pengguna hingga pada akhirnya bisa meningkatkan jumlah pengguna.

3. Meningkatkan Kualitas *Branding*

Selain penting untuk interaksi dan pertumbuhan bisnis, tampilan UI juga bisa menjadi ciri khas sebuah produk. Sederhananya bisa disebut sebagai branding. Misalnya website produk fashion, kuliner, dan teknologi. Contoh lainnya, yaitu tampilan sistem operasi iOS dan Android. Elemen tampilan UI yang sesuai dengan konsep produk tentu akan lebih mudah dikenal oleh pengguna.

Karakteristik UI :

1. Jelas dan ringkas
2. Desain Responsive
3. Informasi Terstruktur
4. Konsisten
5. Kontras warna yang baik
6. Intuitif

2.2.4 Database

Sebuah basis data (*database*) adalah sebuah kumpulan data yang saling berhubungan secara logis, dan merupakan sebuah penjelasan dari data tersebut, yang didesain untuk menemukan data yang dibutuhkan oleh sebuah organisasi. [10]

Di dalam basis data, semua data diintegrasikan dengan menghindari duplikasi data. Basis data dapat digunakan oleh banyak departemen dan pemakai. Basis data tidak hanya memegang data operasional organisasi, tetapi juga penjelasan mengenai data tersebut. Karena alasan tersebut basis data dapat juga dideskripsikan sebagai kumpulan data yang saling terintegrasi. Basis data juga merupakan sekumpulan

elemen data terintegrasi yang secara logika saling berhubungan. Basis data mengkonsolidasikan berbagai catatan yang terlebih dahulu disimpan dalam file-file terpisah ke dalam satu gabungan umum elemen data yang menyediakan data untuk banyak aplikasi. Elemen data mendeskripsikan entitas-entitas dan hubungan antara entitas-entitas tersebut.[10]

Dalam implementasinya, untuk memudahkan dalam mengakses data, data disusun dalam suatu struktur logis yang menjelaskan bahwa:[11]

1. Kumpulan tabel menyusun basis data.
2. Tabel tersusun atas sejumlah *record*.
3. Sebuah *record* mengandung sejumlah *field*.
4. Sebuah *field* disimpan dalam bentuk kumpulan bit.

Berikut merupakan pengertian dari istilah di atas:[11]

1. *Field* menyatakan data terkecil yang memiliki makna. Istilah lain untuk *field* yaitu elemen data, kolom item, dan atribut. Contoh *field* yaitu nama, alamat, telepon, dan jenis kelamin.
2. *Record* menyatakan kumpulan dari sejumlah elemen data yang saling terkait. Sebagai contoh, nama, alamat, bit dan jenis kelamin dari seseorang menyusun sebuah *record*. Istilah lain yang juga menyatakan *record* yaitu tupel dan baris.
3. Tabel menghimpun sejumlah *record*. Sebagai contoh, data pribadi dari semua pegawai disimpan dalam sebuah tabel.
4. Basis data (*database*), adalah sekumpulan data terintegrasi, yang diorganisasi untuk memenuhi kebutuhan para pemakai di dalam suatu organisasi. Sebagai contoh, basis data akademik

Dalam pengembangan suatu sistem informasi, basis data memiliki peran yang penting dan sangat diperlukan. Berikut merupakan alasan perlunya basis data:[11]

1. Salah satu komponen penting dalam suatu sistem informasi adalah basis data, karena basis data merupakan dasar dalam menyediakan informasi.
2. Informasi dikatakan lebih bernilai jika memiliki manfaat yang efektif dibandingkan dengan biaya dalam mendapatkannya. Dalam hal ini, basis data akan menentukan kualitas informasi.
3. Data akan dapat saling berelasi dengan mengimplementasikan basis data.

4. Basis data mengurangi duplikasi data (*data redundancy*). Basis data dapat mengurangi pemborosan tempat simpanan luar.

Basis data dirancang sedemikian rupa untuk selanjutnya diintegrasikan dengan sistem informasi. Orang yang berkepentingan dengan basis data meliputi:[11]

1. Pemakai akhir dan vendor DBMS (*Data Base Management System*).
2. Programmer aplikasi basis data.
3. Administrator Basis Data (*Database Administrator*).

Tujuan dalam merancang basis data adalah:[11]

1. Kecepatan dan kemudahan (*Speed*)

Pemanfaatan basis data memungkinkan untuk dapat menyimpan data atau melakukan perubahan/manipulasi terhadap data atau menampilkan kembali data tersebut dengan cepat dan mudah.

2. Efisiensi ruang penyimpanan (*Space*)

Penggunaan ruang penyimpanan di dalam basis data dilakukan untuk mengurangi jumlah *redundancy* (pengulangan) data, baik dengan melakukan penerapan sejumlah pengkodean atau dengan membuat relasi-relasi (dalam bentuk *file*) antarkelompok data yang saling berhubungan.

3. Keakuratan (*Accuracy*)

Pemanfaatan pengkodean atau pembentukan relasi antar data bersama dengan penerapan aturan/batasan tipe data, domain data, keunikan data, dan sebagainya, yang diterapkan dalam basis data, sangat berguna untuk menentukan ketidaktepatan pemasukan atau penyimpanan data.

4. Ketersediaan (*Availability*)

Pertumbuhan data (baik dari jumlah maupun jenisnya) sejalan dengan waktu akan semakin membutuhkan ruang penyimpanan yang besar. Data yang sudah jarang atau bahkan tidak pernah lagi digunakan dapat diatur untuk dilepaskan dari sistem basis data dengan cara penghapusan atau dengan memindahkannya ke media penyimpanan.

5. Kelengkapan (*Completeness*)

Lengkap atau tidaknya data yang dikelola bersifat relatif, baik terhadap kebutuhan pemakai maupun terhadap waktu. Dalam sebuah basis data, struktur dari basis data harus disimpan. Untuk mengakomodasi kebutuhan kelengkapan data yang semakin

berkembang, tidak cukup hanya menambah *record-record* data, tetapi juga harus melakukan penambahan struktur dalam basis data.

6. Keamanan (*Security*)

Sistem keamanan digunakan untuk dapat menentukan siapa saja yang boleh menggunakan basis data dan menentukan jenis operasi apa saja yang boleh dilakukan.

7. Kebersamaan pemakai (*Shareability*)

Pemakai basis data sering kali tidak terbatas hanya pada satu pemakai saja atau oleh satu sistem aplikasi saja. Basis data yang dikelola oleh sistem (aplikasi) yang mendukung lingkungan *multiuser*, akan dapat memenuhi kebutuhan ini, tetapi dengan menjaga/menghindari terhadap munculnya persoalan baru seperti inkonsistensi data (karena data yang sama diubah oleh banyak pemakai pada saat bersamaan).

Database Management System (DBMS) adalah perangkat lunak yang didesain untuk menangani dalam hal pemeliharaan dan utilitas kumpulan data dalam jumlah besar. DBMS dapat menjadi alternatif penggunaan secara khusus untuk aplikasi, misalnya penyimpanan data dalam *field* dan menulis kode aplikasi yang spesifik untuk pengaturannya.[11]

Fungsi-fungsi yang disediakan DBMS antara lain:[11]

1. Penyimpanan, pengambilan dan perubahan data. Sebuah DBMS harus menyediakan kemampuan menyimpan, mengambil dan merubah data dalam basis data.
2. Katalog yang dapat diakses pemakai. Menyediakan sebuah katalog yang berisi deskripsi item data yang disimpan dan diakses oleh pemakai.
3. Mendukung transaksi. Menyediakan mekanisme yang akan menjamin semua perubahan yang berhubungan dengan transaksi yang sudah ada atau yang akan dibuat.
4. Melayani *Control concurrency*. Sebuah DBMS harus menyediakan mekanisme yang menjamin basis data *ter-update* secara benar pada saat beberapa pemakai melakukan perubahan terhadap basis data yang sama secara bersamaan.
5. Melayani *recovery*. Menyediakan mekanisme untuk mengembalikan basis data ke keadaan sebelum terjadinya kerusakan pada basis data tersebut.
6. Melayani otorisasi. Sebuah DBMS harus menyediakan mekanisme untuk menjamin bahwa hanya pemakai yang berwenang saja yang dapat mengakses basis data.

7. Mendukung komunikasi data. Sebuah DBMS harus mampu terintegrasi dengan *software* komunikasi.
8. Melayani *integrity*. Sebuah DBMS bertujuan untuk menjamin semua data dalam basis data dan setiap terjadi perubahan data harus sesuai dengan aturan yang berlaku.
9. Melayani data *independence*. Sebuah DBMS harus mencakup fasilitas untuk mendukung kemandirian program dari struktur basis data yang sesungguhnya.
10. Melayani *utility*. Sebuah DBMS sebaiknya menyediakan kumpulan layanan *utility*.

2.2.5 Normalisasi

Normalisasi merupakan sebuah proses yang dilakukan dalam mendesain tabel yang termasuk ke dalam proses *logical design*. Adapun tujuan dilakukannya normalisasi adalah sebagai berikut:[12]

1. Memecah tabel ke beberapa tabel-tabel.
2. Meminimalisasi redundansi atau perulangan data.
3. Mempercepat proses *Update, Insert, dan Delete*.

Normalisasi memiliki efek samping yaitu untuk proses *select* yang memerlukan data yang lengkap harus dilakukan dengan penggabungan beberapa tabel atau *join* yang akan menyebabkan *performance* menurun. Ada beberapa jenis normalisasi yaitu: 1NF (*First Normal Form*), 2NF (*Second Normal Form*), 3NF (*Third Normal Form*), dan seterusnya. Pada umumnya proses normalisasi hanya sampai tahap 3NF.[12]

1. Normalisasi 1NF.

Berikut poin-poin penjelasan tentang normalisasi 1NF:

- a. Baris data dalam tabel harus unik (Unique Key di tabel).
- b. Atribut harus atomic. Atomic : (FullName)?, (FirstName & LastName)?, (FirstName, MiddleName, LastName)?.
- c. Sifat atomic tergantung dari kebutuhan aplikasi. Sifat atomic terhadap sebuah atribut sangat subjektif, tergantung dari kebutuhan aplikasi. Contoh atribut yang dapat melanggar aturan sifat atomic misalnya yaitu apakah atribut nama akan dibuat menjadi satu atribut atau lebih dari satu dengan membaginya ke beberapa atribut seperti FirstName, MiddleName, dan LastName.

Contoh pada sewa menyewa rumah dalam tabel berikut :[10]

No. Penyewa	Nama Penyewa	No. Properti	Alamat Properti	Tgl Mulai Sewa	Tgl Akhir Sewa	Sewa Per Bulan	No. Pemilik	Nama Pemilik
S001	Indrajani	PR1	Jl. Kebun Jeruk No. 1, Jakarta Barat	01/11/2008	01/12/2008	500,000	PP99	Matus
		PR4	Jl Gatot Subroto No. 100, Jakarta	01/01/2009	01/06/2009	1,000,000	PP77	Wahyu
S003	Indrayeti	PR4	Jl Gatot Subroto No. 100, Jakarta	01/11/2008	01/12/2009	1,000,000	PP77	Wahyu
		PR1	Jl. Kebun Jeruk No. 1, Jakarta Barat	01/11/2007	01/12/2007	500,000	PP99	Matus
		PR2	Jl. Sudirman No. 2, Bandung	01/11/2008	01/12/2008	750,000	PP99	Matus

Gambar 2.6 Sewa Penyewa Rumah UNF

Dari bentuk UNF di atas, terdapat pengulangan grup, yaitu:

No_Properti, Alamat_Properti, Tgl_Mulai_Sewa, Tgl_Akhir_Sewa, Sewa Perbulan, No Pemilik, dan Nama Pemilik.

Penulisan UNF dengan notasi :

SewaRumah = No_Penyewa + Nama_Penyewa + {No_Properti + Alamat_Properti + Tgl_Mulai_Sewa + Tgl_Akhir_Sewa + Sewa_Perbulan + No_Pemilik + Nama_Pemilik}

Penyewa	
No. Penyewa	Nama Penyewa
S001	Indrajani
S003	Indrayeti

SewaRumah								
No. Penyewa	No. Properti	Alamat Properti	Tgl Mulai Sewa	Tgl Akhir Sewa	Sewa Per Bulan	No. Pemilik	Nama Pemilik	
S001	PR1	Jl. Kebun Jeruk No. 1, Jakarta Barat	01/11/2008	01/12/2008	500,000	PP99	Matus	
S001	PR4	Jl Gatot Subroto No. 100, Jakarta	01/01/2009	01/06/2009	1,000,000	PP77	Wahyu	
S003	PR4	Jl Gatot Subroto No. 100, Jakarta	01/11/2008	01/12/2009	1,000,000	PP77	Wahyu	
S003	PR1	Jl. Kebun Jeruk No. 1, Jakarta Barat	01/11/2007	01/12/2007	500,000	PP99	Matus	
S003	PR2	Jl. Sudirman No. 2, Bandung	01/11/2008	01/12/2008	750,000	PP99	Matus	

Gambar 2.7 INF

Penulisan INF menjadi :

- Penyewa = No_Penyewa + Nama_Penyewa
- SewaRumah = No Penyewa + No Properti + Alamat Properti + TgL_Mulai_Sewa + Tgl_akhir_Sewa + Sewa_PerBulan + No_Pemilik + Nama Pemilik

2. Normalisasi 2NF

Berikut poin-poin penjelasan tentang normalisasi 2NF:

- Memenuhi syarat 1NF.
- Hubungan antara atribut non-key dengan candidate key.
- Setiap *non-key* atribut harus bergantung sepenuhnya secara fungsionalitas kepada semua candidate key.
- Sebuah *non-key* atribut tidak boleh bergantung sepenuhnya secara fungsionalitas kepada bagian dari sebuah *candidate key*.

Penyewa	
No. Penyewa	Nama Penyewa
S001	Indrajani
S003	Indrayeti

SewaRumah			
No. Penyewa	No. Properti	Tgl Mulai Sewa	Tgl Akhir Sewa
S001	PR1	01/1/2008	01/12/2008
S001	PR4	01/01/2009	01/06/2009
S003	PR4	01/1/2008	01/12/2009
S003	PR1	01/1/2007	01/12/2007
S003	PR2	01/1/2008	01/12/2008

Properti Pemilik				
No. Properti	Alamat Properti	Sewa Per Bulan	No. Pemilik	Nama Pemilik
PR1	Jl. Kebun Jeruk No. 1, Jakarta Barat	500,000	PP99	Matus
PR2	Jl. Sudirman No. 2, Bandung	750,000	PP99	Matus
PR4	Jl. Gatot Subroto No. 100, Jakarta	1,000,000	PP77	Wahyu

Gambar 2.8 2NF

Penulisan 2NF menjadi :

- Penyewa = No Penyewa + Nama_Penyewa
- SewaRumah = No Penyewa + No Properti + Tgl_Mulai_Sewa + Tgl_Akhir_Sewa + Sewa_PerBulan
- Properti_Pemilik = No Properti + Alamat_Pemilik + Sewa_PerBulan + No Pemilik + Nama Pemilik

3. Normalisasi 3NF

Berikut poin-poin penjelasan tentang normalisasi 3NF:

- Memenuhi syarat 2NF.
- Semua non-key atribut harus bergantung pada candidate key.
- Semua non-key atribut harus independen satu sama lainnya.
- Satu non-key atribut tidak dapat bergantung pada non-key atribut lainnya

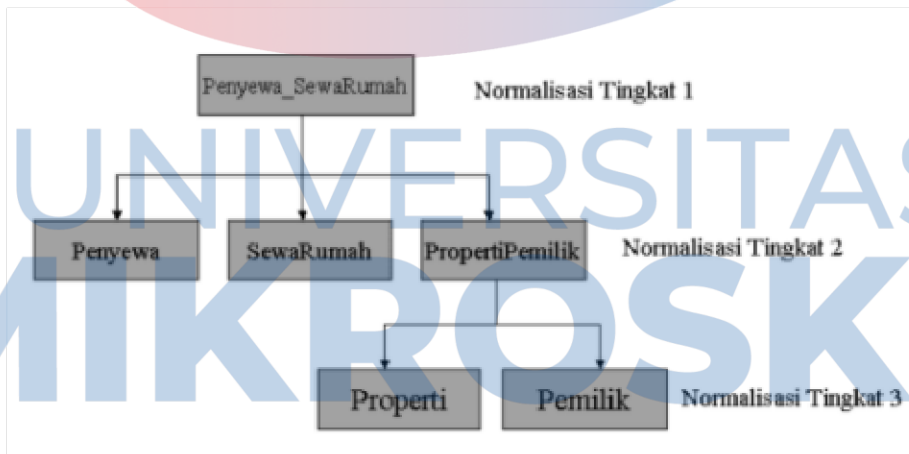
Properti				Pemilik	
No. Properti	Alamat Properti	Sewa Per Bulan	No. Pemilik	No. Pemilik	Nama Pemilik
PR1	Jl. Kebun Jeruk No. 1, Jakarta Barat	500,000	PP99	PP99	Matus
PR2	Jl. Sudirman No. 2, Bandung	750,000	PP99	PP77	Wahyu
PR4	Jl. Gatot Subroto No. 100, Jakarta	1,000,000	PP77		

Gambar 2.9 3NF

Penulisan 3NF menjadi :

- Penyewa = No Penyewa + Nama_Penyewa
- SewaRumah = No Penyewa+ No Properti + Tgl_Mulai_Sewa + Tgl_Akhir_Sewa + Sewa_PerBulan
- Properti = No Properti + Alamat_Properti + Sewa_PerBulan + No_Pemilik
- Pemilik = No Pemilik+ Nama_Pemilik

Dekomposisi 1NF → 3NF :



Gambar 2.10 Dekomposisi 1NF -> 3NF

Hasil dari normalisasi :

Penyewa	
No. Penyewa	Nama Penyewa
S001	Indrayani
S003	Indrayeti

SewaRumah			
No. Penyewa	No. Properti	Tgl Mulai Sewa	Tgl Akhir Sewa
S001	PR1	01/1/2008	01/1/2/2008
S001	PR4	01/01/2009	01/06/2009
S003	PR4	01/1/2008	01/1/2/2009
S003	PR1	01/1/2007	01/1/2/2007
S003	PR2	01/1/2008	01/1/2/2008

Properti				Pemilik	
No. Properti	Alamat Properti	Sewa Per Bulan	No. Pemilik	No. Pemilik	Nama Pemilik
PR1	Jl. Kebun Jeruk No. 1, Jakarta Barat	500,000	PP99	PP99	Matus
PR2	Jl. Sudirman No. 2, Bandung	750,000	PP99	PP77	Wahyu
PR4	Jl. Gatot Subroto No. 100, Jakarta	1,000,000	PP77		

Gambar 2.11 Hasil Normalisasi

2.2.6 Metode Piesces

Untuk mengidentifikasi masalah, harus dilakukan analisis terhadap kinerja, informasi, ekonomi, keamanan aplikasi, efisiensi, dan pelayanan pelanggan. Panduan ini dikenal dengan analisis PIECES (*performance, information, economy, control, efficiency, dan services*). Dari analisis ini biasanya didapatkan beberapa masalah utama. Hal ini penting karena biasanya yang muncul di permukaan bukan masalah utama, tetapi hanya gejala dari masalah utama saja. [13]

Analisis *PIECES* ini sangat penting untuk dilakukan sebelum mengembangkan sebuah sistem informasi karena dalam analisis ini biasanya akan ditemukan beberapa masalah utama maupun masalah yang bersifat gejala dari masalah utama. Metode ini menggunakan enam variable evaluasi yaitu : [14]

1. *Performance* (kinerja)

Kinerja merupakan variabel pertama dalam metode analisis *PIECES*. Dimana memiliki peran penting untuk menilai apakah proses atau prosedur yang ada masih mungkin ditingkatkan kinerjanya, dan melihat sejauh mana dan seberapa handalkah suatu sistem informasi dalam berproses untuk menghasilkan tujuan yang diinginkan. Dalam hal ini kinerja diukur dari:

a. *throughput*, yaitu jumlah pekerjaan/output/deliverables yang dapat dilakukan/dihasilkan pada saat tertentu.

b. *response time*, yaitu waktu yang dibutuhkan untuk menyelesaikan serangkaian kegiatan untuk menghasilkan *output/deliverables* tertentu.

2. *Information* (informasi)

Menilai apakah prosedur yang ada saat ini masih dapat diperbaiki sehingga kualitas informasi yang dihasilkan menjadi semakin baik. Informasi yang disajikan haruslah benar-benar mempunyai nilai yang berguna. Hal ini dapat diukur dengan :

a. Keluaran (*outputs*): Suatu sistem dalam memproduksi keluaran.

b. Masukan (*inputs*): Dalam memasukkan suatu data sehingga kemudian diolah untuk menjadi informasi yang berguna.

3. *Economic* (ekonomi)

Menilai apakah prosedur yang ada saat ini masih dapat ditingkatkan manfaatnya (nilai gunanya) atau diturunkan biaya penyelenggaraannya.

4. *Control* (pengendalian)

Menilai apakah prosedur yang ada saat ini masih dapat ditingkatkan sehingga kualitas pengendalian menjadi semakin baik, dan kemampuannya untuk mendeteksi kesalahan/kecurangan menjadi semakin baik pula.

5. *Efficiency* (efisiensi)

Menilai apakah prosedur yang ada saat ini masih dapat diperbaiki, sehingga tercapai peningkatan efisiensi operasi, dan harus lebih unggul dari pada sistem manual.

6. *Service* (layanan)

Menilai apakah prosedur yang ada saat ini masih dapat diperbaiki kemampuannya untuk mencapai peningkatan kualitas layanan. Buatlah kualitas layanan yang sangat user friendly untuk end – user (pengguna) sehingga pengguna mendapatkan kualitas layanan yang baik.

Jenis Kebutuhan	Penjelasan
Non-Fungsional	

Kinerja (<i>Performance</i>)	1. Pekerjaan diharapkan dapat diselesaikan lebih cepat, sehingga dapat meningkatkan <i>trouhgput system</i> yang sesuai dengan jenis proses pada <i>System Inprovement Objectives</i> .
Informasi (<i>Information</i>)	1. Mencegah terjadinya redudansi data 2. Data harus akurat, yaitu dengan meminimalisasi kesalahan durasi waktu untuk pengelolaan data absensi 3. Data harus konsisten
Ekonomi (<i>Economic</i>)	1. Biaya yang dibutuhkan untuk pengembangan system tidak melebihi budget yang telah ditentukan
Pengontrolan Sistem (<i>Control</i>)	1. Membuat <i>log</i> dari setiap aktivitas <i>user</i> 2. Mengotentikasi <i>user</i> yang boleh menggunakan system sesuai dengan jabatannya
Pelayanan Sistem (<i>Service</i>)	1. Memberikan data-data yang akurat dan lengkap untuk pengambilan keputusan pihak owner 2. Data yang ditampilkan harus mudah dibaca dan terstruktur

Tabel 2.1 Tabel Metode PIECES

2.3 Bengkel

Bengkel adalah tempat di mana seseorang mekanik melakukan pekerjaannya melayani jasa perbaikan dan perawatan kendaraan. Bengkel umum kendaraan bermotor adalah bengkel umum yang berfungsi untuk membetulkan, memperbaiki, dan merawat kendaraan bermotor agar tetap memenuhi persyaratan teknis dan layak jalan. Hal ini guna memenuhi tuntutan PP No. 44 Tahun 1993 tentang Kendaraan dan pengemudi pasal 126, 127, 128, dan 129, dinyatakan bahwa setiap kendaraan bermotor harus memenuhi persyaratan teknis dan kelayakan kendaraan bermotor. Bengkel merupakan suatu usaha jenis wirausaha kecil dan menengah yang bergerak dalam bidang jasa pelayanan perbaikan, baik itu sepeda motor atau mobil. Usaha bengkel sepeda motor adalah usaha yang melakukan perbaikan sepeda motor agar dapat

kembali berjalan dengan baik sesuai dengan keinginan pemilik atau bentuk asli dari sepeda motor tersebut.[15]

Bengkel sendiri memiliki beberapa jenis yang memiliki perbedaan mendasar. Berdasarkan SK Menperindag nomor 551/MPP/Kep/10/1999 tentang bengkel umum kendaraan bermotor dibedakan berdasarkan sistem mutu dan sumber daya manusia nya, menjadi beberapa kelas dan tipe, terdiri atas : [15]

1. Bengkel kelas I tipe A, B, dan C
2. Bengkel kelas II tipe A, B, dan C
3. Bengkel kelas III tipe A, B, dan C

Klasifikasi bengkel kelas I, II, III dapat dilihat spesifikasinya sesuai dengan lampiran I yang terlampir pada peraturan tersebut. Sedang tipe bengkel yang dimaksud adalah sebagai berikut :[15]

1. Bengkel tipe A merupakan bengkel yang mampu melakukan jenis pekerjaan perawatan berkala, perbaikan kecil, perbaikan besar, perbaikan chassis dan body
2. Bengkel tipe B merupakan bengkel yang mampu melakukan jenis pekerjaan perawatan berkala, perbaikan kecil dan perbaikan besar, atau jenis pekerjaan perawatan berkala, perbaikan kecil serta perbaikan chassis dan body.
3. Bengkel tipe C merupakan bengkel yang mampu melakukan jenis pekerjaan perawatan berkala, perbaikan kecil.

2.4 Persediaan

Persediaan adalah sebagai suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam suatu periode usaha yang normal, atau persediaan barang-barang masih dalam pengerjaan / proses produksi, ataupun persediaan barang baku yang menunggu penggunaannya dalam suatu proses produksi. Berdasarkan keterangan di atas dapat diketahui bahwa persediaan sangat penting artinya bagi suatu perusahaan karena berfungsi menghubungkan antara operasi yang berurutan dalam pembuatan suatu barang dan menyampaikannya kepada konsumen.[16]

Dilihat dari fungsinya, fungsi-fungsi persediaan dapat dikelompokkan ke dalam empat jenis, yaitu:[16]

1. *Fluctuation Stock*, merupakan persediaan yang dimaksudkan untuk menjaga terjadi fluktuasi permintaan yang tidak diperkirakan sebelumnya, dan untuk mengatasi bila terjadi kesalahan/penyimpangan dalam perkiraan penjualan waktu produksi, atau pengiriman barang.

2. *Anticipation Stock*, merupakan persediaan untuk menghadapi permintaan yang dapat diramalkan, misalnya pada musim permintaan tinggi, tetapi kapasitas produksi pada saat itu tidak mampu memenuhi permintaan. Persediaan ini juga dimaksudkan untuk menjaga kemungkinan sukarnya diperoleh bahan baku sehingga tidak mengakibatkan terhentinya produksi.

3. *Lot-size Inventory*, merupakan persediaan yang diadakan dalam jumlah yang lebih besar daripada kebutuhan pada saat itu. Persediaan dilakukan untuk mendapatkan keuntungan dari harga barang (berupa diskon) karena membeli dalam jumlah yang besar, atau untuk mendapatkan penghematan dari biaya pengangkutan per unit yang lebih rendah.

4. *Pipeline Inventory*, merupakan persediaan yang dalam proses pengiriman dari tempat asal ke tempat di mana barang itu akan digunakan. Misalnya, barang yang dikirim dari pabrik menuju tempat penjualan, yang dapat memakan waktu beberapa hari atau minggu.

Beberapa cara penilaian persediaan, yaitu : [17]

1. Metode *First In First Out* (FIFO)

Dalam metode ini *cost* per unit yang digunakan untuk menentukan *cost* persediaan, yaitu menggunakan *cost* per unit dari barang yang terakhir kali masuk, dan untuk menentukan *cost* barang yang terjual didasarkan pada *cost* per unit dari barang dagangan yang pertama kali masuk.

2. Metode *Last In First Out* (LIFO)

Penentuan *cost* persediaan akhir dan *cost* barang yang terjual didasarkan atas *cost* barang yang terakhir kali masuk (dibeli), dan untuk menentukan *cost* persediaan akhir berdasarkan *cost* barang dagangan yang pertama kali masuk (dibeli).

3. Metode Rata-Rata (*Average Method*)

Metode ini sering disebut dengan istilah metode rata-rata tertimbang (*weighted average method*), didasarkan atas anggapan bahwa *cost* yang harus diperhitungkan untuk menentukan *cost* sediaan akhir dan *cost* barang yang terjual dengan menggunakan *cost* rata-rata per unit.



UNIVERSITAS MIKROSKIL