

BAB II

KAJIAN LITERATUR

2.1 Konsep Sistem Informasi

Sistem berasal dari Bahasa Yunani yaitu “*systeme*” yang mempunyai pengertian [1]:

1. Suatu keseluruhan yang tersusun dari sekian banyak bagian.
2. Hubungan yang berlangsung di antara satuan-satuan atau komponen-komponen secara teratur.

Sistem dapat didefinisikan sebagai kumpulan dari elemen-elemen berupa data, jaringan kerja dari prosedur-prosedur yang saling berhubungan, sumber daya manusia, teknologi baik *hardware* maupun *software* yang saling berinteraksi sebagai satu kesatuan untuk mencapai tujuan/sasaran tertentu yang sama. Urutan kegiatan dalam prosedur digunakan untuk menjelaskan apa yang harus dikerjakan, mengapa dikerjakan dan bagaimana mengerjakannya. Suatu sistem sendiri dapat terdiri dari beberapa subsistem. Misalnya sistem akuntansi dapat terdiri dari beberapa subsistem-subsistem, yaitu subsistem akuntansi penjualan, subsistem akuntansi pembelian, subsistem akuntansi penggajian, subsistem akuntansi biaya dan sebagainya. [2]

Ciri-ciri atau sifat-sifat sistem adalah [1]:

1. Terdiri dari komponen-komponen yang saling berinteraksi.
2. Mempunyai lingkungan luar.
3. Mempunyai *interface* (jalinan).
4. Terdiri dari masukan, pengolahan dan keluaran.

Sebuah sistem harus memenuhi syarat minimumnya yaitu memiliki 3 unsur pembentuk sistem, terdiri dari *input*, proses dan *output*. *Input* adalah data atau informasi yang dibutuhkan oleh sebuah sistem untuk selanjutnya diproses sesuai dengan ketentuan proses yang telah ditentukan. Pada akhirnya sistem akan menghasilkan keluaran (*output*) yang bila diperlukan lagi maka hasil *output* tersebut akan kembali menjadi sebuah *input*, begitu seterusnya, ini yang disebut dengan *system life cycle* (siklus hidup sistem) [2].

Informasi merupakan hasil pengolahan data dengan cara tertentu sehingga lebih berarti dan berguna bagi penerimanya. Menurut George R. Terry, bahwa informasi adalah data yang penting yang memberikan pengetahuan yang berguna. Sedangkan menurut Gordon B. Davis, informasi adalah data yang telah diolah menjadi sebuah bentuk yang penting bagi penerima dan

mempunyai nilai yang nyata atau yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau yang akan datang [1]. Informasi dikatakan berkualitas jika memenuhi 3 aspek yaitu akurat (*accurate*), tepat waktu (*timeliness*), dan relevan (*relevance*) [3].

1. Akurat (*Accuracy*)

Informasi harus tepat dan tidak bias dan terbebas dari kesalahan-kesalahan dan tidak menyesatkan. Sebuah informasi harus sesuai, tidak *hoax* dan tidak ambigu ketika sampai ke penerima informasi [4].

2. Tepat waktu (*Timeliness*)

Informasi harus sampai ke penerima dengan waktu yang tepat dan tidak boleh terlambat, serta sebuah informasi yang tidak bernilai adalah informasi yang sudah usang [4].

Sejarah ini penyampaian informasi sangat mudah dan penerima juga cepat dalam memperoleh informasi sehingga membutuhkan teknologi mutakhir dan informasi tersebut adalah informasi terbaru [4].

3. Relevan (*Relevance*)

Sebuah informasi yang baik bermanfaat bagi penerimanya. Sebuah relevansi informasi terjadi ketika perbedaan yang didapat oleh orang satu dengan yang lainnya [4].

Sistem informasi merupakan proses pengumpulan, penyimpanan, analisis sebuah informasi dengan tujuan tertentu. Sistem informasi yang terdiri dari data (*input*) dan menghasilkan laporan (*output*) sehingga diterima oleh sistem lainnya serta kegiatan strategi dalam suatu organisasi dalam melakukan tindakan atau keputusan [4].

Sistem informasi merupakan kombinasi antara teknologi, manusia, dan data yang digunakan untuk menyediakan dukungan terhadap fungsi bisnis seperti *order processing*, *inventory control*, *human resources*, *accounting* dan lain sebagainya. Sistem informasi terdiri atas 5 (lima) komponen kunci, yaitu *hardware*, *software*, data, proses, dan manusia [5].

1. *Hardware*

Hardware terdiri atas segala sesuatu di lapisan fisik sistem informasi. Sebagai contoh, *hardware* dapat terdiri dari *server*, *workstation*, jaringan, peralatan telekomunikasi, kabel *fiber-optic*, perangkat seluler, pemindai, perangkat penangkap digital, dan infrastruktur berbasis teknologi lainnya.

2. *Software*

Software mengacu pada program yang mengontrol *hardware* dan menghasilkan informasi atau hasil yang diinginkan. *Software* terdiri dari perangkat lunak sistem dan perangkat lunak aplikasi.

3. Data

Data adalah bahan mentah yang diubah oleh sistem informasi menjadi informasi yang berguna.

4. Proses

Proses menggambarkan tugas dan fungsi bisnis yang dilakukan pengguna, manajer, dan anggota staf TI untuk mencapai hasil tertentu. Proses adalah blok bangunan dari sistem informasi karena mereka mewakili operasi bisnis sehari-hari yang sebenarnya.

5. Manusia

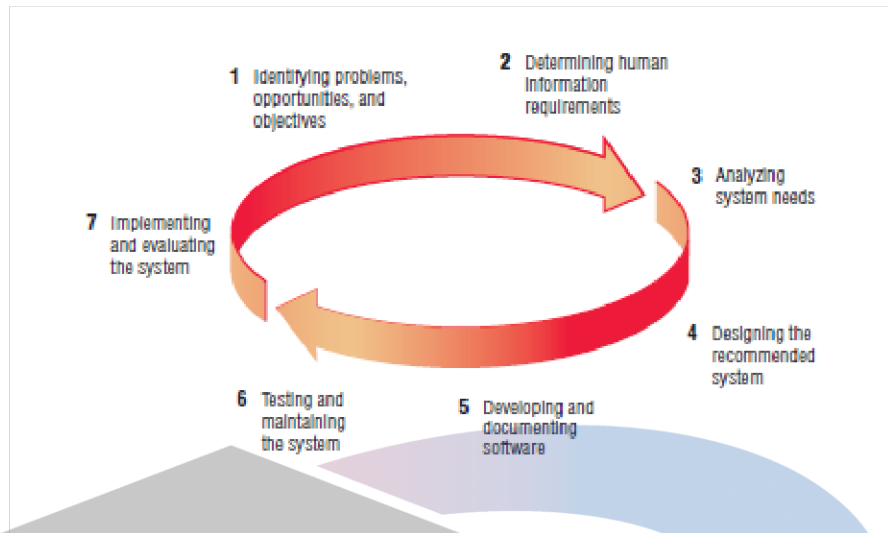
Manusia yang berkepentingan dengan suatu sistem informasi disebut *stakeholder* (pemangku kepentingan). Pemangku kepentingan termasuk kelompok manajemen yang bertanggung jawab atas sistem, pengguna (kadang-kadang disebut pengguna akhir) di dalam dan di luar perusahaan yang akan berinteraksi dengan sistem, dan anggota staf TI, seperti *systems analysts*, *programmers* dan *network administrators* yang mengembangkan dan mendukung sistem [5].

2.2 Siklus Hidup Pengembangan Sistem

Analisis terstruktur adalah teknik pengembangan sistem tradisional yang telah teruji waktu dan mudah dipahami. Analisis terstruktur menggunakan serangkaian fase, yang disebut *systems development life cycle* (SDLC), untuk merencanakan, menganalisis, merancang, mengimplementasikan, dan mendukung sistem informasi [5].

Software Development Life Cycle atau SDLC adalah proses yang digunakan untuk merancang, mengembangkan, dan menguji software yang berkualitas tinggi. Tujuan dari SDLC adalah untuk menyediakan alur terstruktur dalam membantu organisasi menghasilkan software berkualitas tinggi yang diselesaikan dengan perkiraan waktu yang singkat dan biaya yang lebih rendah, namun tetap memenuhi atau melebihi harapan pelanggan [6].

Tujuh tahap SDLC berdasarkan Kendall dan Kendall adalah [6]:



Gambar 2.1 Siklus Hidup Pengembangan Sistem

1. Mengidentifikasi masalah, peluang, dan tujuan

Pada tahap pertama ini, analis memperhatikan dengan benar masalah yang diidentifikasi, peluang, serta tujuannya. Tahap ini sangat penting untuk kesuksesan proyek berikutnya. Orang-orang yang terlibat di antaranya adalah pengguna, analis, dan manajer koordinator sistem proyek ini.

2. Menentukan informasi yang dibutuhkan oleh *user*

Tahap berikutnya yaitu menentukan informasi yang dibutuhkan, menggunakan beberapa alat untuk mengetahui bagaimana user berinteraksi dalam konteks kerja dengan sistem informasi sebelumnya. Analis akan menggunakan metode langsung seperti wawancara, sampel, investigasi data, dan kuesioner. Orang-orang yang terlibat di antaranya adalah penganalisis dan pemakai. Penganalisis sistem perlu tahu secara detail fungsi-fungsi sistem yang ada, siapa (orang-orang yang terlibat), apa (kegiatan proyek), di mana (lingkungan di mana pekerjaan itu dilakukan), kapan (waktu yang tepat), dan bagaimana (bagaimana prosedur yang harus dijalankan) dari proyek yang sedang dipelajari.

3. Menganalisa kebutuhan sistem

Tahap berikutnya sistem analisis melakukan analisa sistem yang dibutuhkan. Beberapa alat dan bantuan teknis membantu analisis membuat penentuan kebutuhan. Alatnya berupa diagram *fishbone*, diagram aliran data (*data flow diagram*) dan kamus data (*data dictionary*).

4. Mendesain sistem yang telah direkomendasikan

Pada tahap ini, sistem analis menggunakan informasi yang telah dikumpulkan sebelumnya untuk memenuhi desain dari sistem informasi. Analis mendesain prosedur yang ditujukan ke pengguna untuk membantu mereka memasukkan data yang benar sehingga data yang ada dalam sistem informasi adalah valid. Penganalisis harus merancang prosedur cadangan dan sebuah kontrol yang bertujuan untuk melindungi sistem, data dan menghasilkan paket spesifikasi bagi setiap *programmer*.

5. Membangun dan mendokumentasikan perangkat lunak

Pada tahap ini, sistem analis bersama *programmer* membangun perangkat lunak yang dibutuhkan. Selama tahap ini analis bekerja dengan pengguna dalam membangun dokumentasi yang efektif untuk perangkat lunak. *Programmer* memiliki tugas merancang, menghapus sintaks dari kode program serta membuat kode. Pengguna dapat mengetahui cara menggunakan perangkat lunak dan masalah perangkat lunak dari dokumentasi.

6. Menguji dan memperbaiki sistem

Sebelum sistem informasi digunakan, harus diuji dulu. Sebagian pengujian telah dilakukan *programmer* dan sebagian lagi oleh sistem analis bersama dengan *programmer*. Sedangkan perbaikan sistem dan dokumentasinya dimulai juga pada tahap ini dan dilakukan secara rutin sepanjang pengembangan sistem informasi.

7. Mengimplementasikan dan mengevaluasi sistem

Pada tahap terakhir ini, sistem analis melakukan implementasi sistem yang sudah dikembangkan. Proses ini termasuk mengubah *file* dari format yang lama ke format baru, membuat *database*, dan meng-*install* sebuah alat yang baru. Tahap ini juga termasuk pelatihan kepada pengguna dalam menjalankan sistem.

Evaluasi dilakukan untuk melihat keberhasilan dalam kegiatan, efektivitas sistem baru, perkiraan biaya, ketepatan waktu pelaksanaan proyek, biaya pemeliharaan sistem. Evaluasi diharapkan agar sistem baru lebih efisien operasionalnya dan efektif dalam pencapaian tujuan, lebih mudah digunakan dan fleksibel terhadap perubahan-perubahan yang diinginkan [6].

2.3 Diagram *Fishbone*

Fishbone Diagram (diagram tulang ikan) sering disebut juga diagram Ishikawa atau *cause-and-effect diagram* (diagram sebab-akibat). Diagram *fishbone* adalah alat untuk mengidentifikasi berbagai sebab potensial dari satu efek atau masalah, dan menganalisis

masalah tersebut melalui sesi *brainstorming*. Fungsi dasarnya adalah untuk mengidentifikasi dan mengorganisasi penyebab-penyebab yang mungkin timbul dari suatu efek spesifik dan kemudian memisahkan akar penyebabnya [7].

Tahapan dalam membuat *fishbone diagram*:

1. Mengidentifikasi masalah

Tahapan pertama dari diagram *fishbone* yaitu menulis masalah yang sebenarnya dihadapi dengan mencari tahu siapa yang terlibat, permasalahannya, dan di mana hal itu terjadi. Masalah utama tersebut digambarkan dalam bentuk kotak sebagai kepala dari diagram *fishbone*. Masalah tersebut akan diidentifikasi dan untuk selanjutnya memberikan ruang untuk mengembangkan ide atau solusi yang berkaitan.

2. Mengumpulkan ide untuk mencari faktor utama penyebab

Langkah berikutnya adalah mengkategorikan penyebab dari masalah yang dihadapi, termasuk faktor-faktor yang mungkin menjadi bagian dari masalah, seperti peralatan, *material*, sistem, SDM, dan lainnya. Faktor-faktor tersebut merupakan penyusun tulang utama dari diagram *fishbone*.

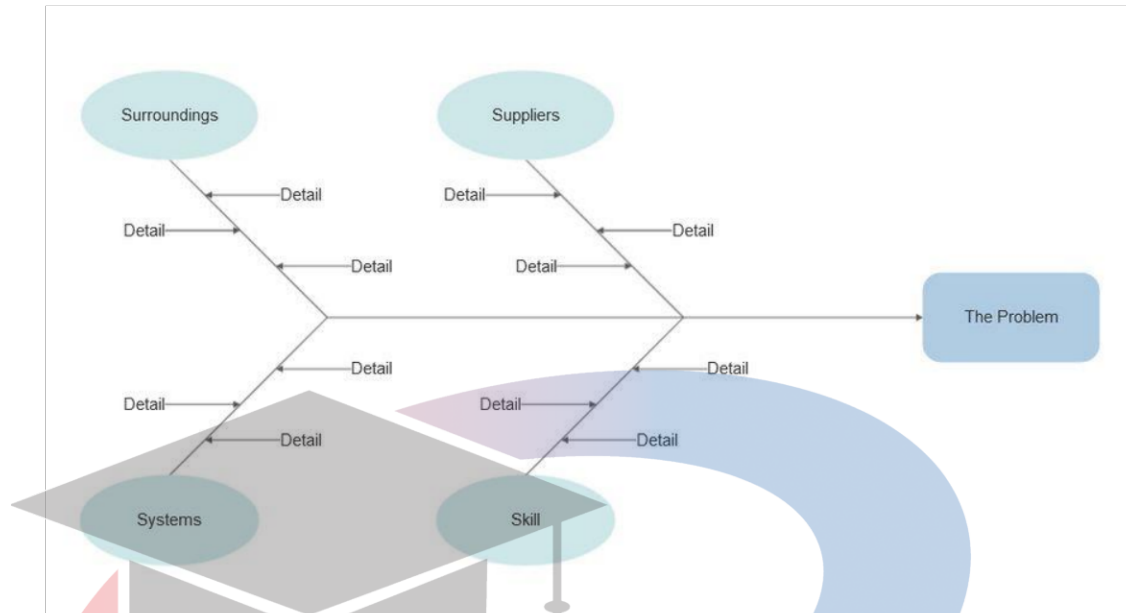
3. Mengidentifikasi kemungkinan penyebab dari masalah

Tahap berikutnya yaitu mengidentifikasi setiap kemungkinan penyebab dari setiap faktor dan menggambarannya sebagai “tulang” kecil dari “tulang” utama. Selanjutnya yaitu mencari tahu akar penyebab dari setiap kemungkinan tersebut dengan melakukan *brainstorming* atau analisis keadaan dengan observasi.

4. Menganalisis diagram yang sudah dibuat

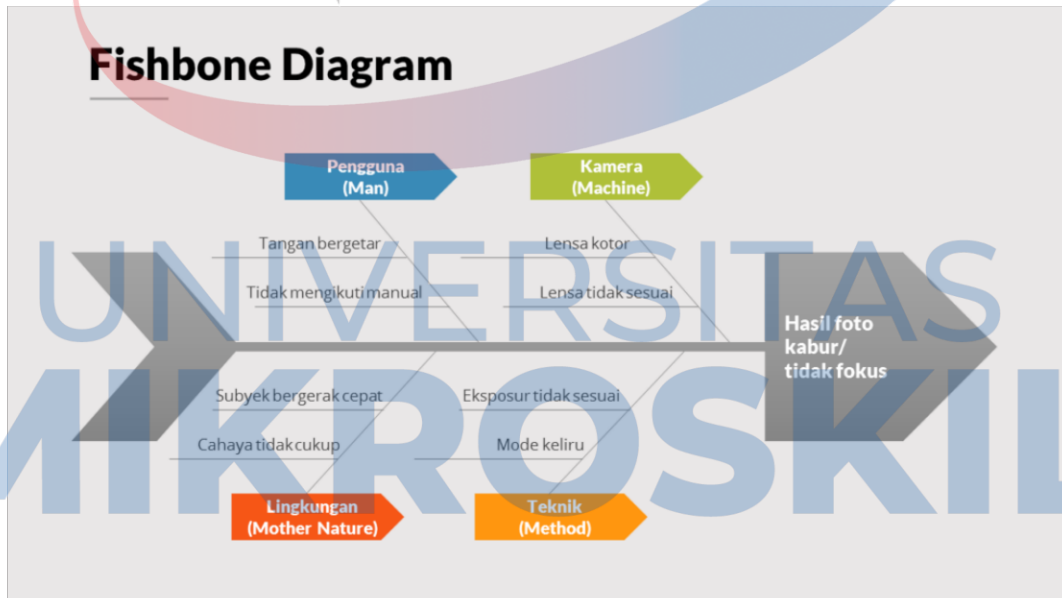
Tahapan terakhir yaitu melakukan analisis lebih lanjut mengenai akar penyebab dengan menginvestigasi atau survei untuk melihat penyebab potensial yang berkontribusi dalam masalah utama dan menemukan solusi untuk menyelesaikannya.

Di bawah ini dapat dilihat contoh susunan diagram *fishbone* :



Gambar 2.2 *Fishbone* Diagram

Berikut merupakan contoh diagram *fishbone*:



Gambar 2.3 Contoh Diagram *Fishbone*

2.4 Diagram Aliran Data (*Data Flow Diagram*)

Salah satu perangkat dalam menggambarkan pemodelan sistem yang paling umum adalah Diagram aliran data (DFD), terutama untuk menggambarkan sistem operasional di mana fungsi sistem sangat penting dan kompleks dibandingkan data yang dimanipulasi sistem. Keunggulan dari DFD yaitu DFD mudah dipahami oleh orang teknik maupun non-teknik,

memberikan gambaran sistem secara menyeluruh, lengkap dengan lingkup sistem dan hubungan ke sistem lainnya dan memberikan tampilan komponen-komponen sistem secara *detail*. Secara ideal, sistem dibangun melalui analisa *current system* (seperti oleh *current logical DFD*), dianalisa dan ditambahkan fitur-fitur untuk memperbaiki atau meningkatkan kinerja sistem tersebut [2].

Data Flow Diagram (DFD) merupakan diagram yang cocok digunakan dalam pemodelan perangkat lunak yang akan diimplementasikan dengan menggunakan pemrograman terstruktur membagi-bagi bagiannya dengan fungsi-fungsi dan prosedur. DFD merupakan diagram yang menggambarkan kegiatan-kegiatan atau aliran data yang ada dalam suatu sistem [8].

Dalam menggambarkan DFD, dikenal empat simbol dasar yang digunakan untuk memetakan gerakan diagram aliran data yaitu sebagai berikut [2]:

1. *External Entity* (Entitas)/*terminator*

Menyatakan entitas atau *external entities* (peran, divisi, mesin, organisasi atau sistem lain) asal atau tujuan dari data, di mana data melakukan komunikasi. Suatu entitas dapat disimbolkan dengan suatu notasi kotak. Entitas ini disebut juga sumber atau tujuan data dan dianggap eksternal terhadap sistem yang sedang digambarkan. Setiap entitas diberi *label* dengan sebuah nama yang sesuai dan diberi nama dengan suatu kata benda. Bentuk dari *external entity* adalah:

1. Suatu kantor, departemen/divisi dalam perusahaan.
2. Orang atau sekelompok orang di organisasi.
3. Suatu organisasi atau orang yang berada di luar organisasi (pelanggan, pemasok).
4. Sistem informasi yang lain di luar sistem yang sedang dikembangkan.
5. Sumber asli dari suatu transaksi.
6. Penerima akhir dari suatu laporan yang dihasilkan oleh sistem.

2. *Data Flow*/Arus Data

Arus atau aliran data (*data flow*) merupakan data yang bergerak dari satu tempat di dalam sistem ke tempat lainnya. Sebuah aliran data dapat terdiri dari banyak potongan data yang dihasilkan. Suatu aliran data dapat disimbolkan dengan menggunakan suatu notasi tanda panah,

dengan kepala tanda panah mengarah ke tujuan data. Karena sebuah tanda panah menunjukkan seseorang, tempat atau sesuatu, maka harus digambarkan dalam kata benda.

Bentuk dari arus data di antaranya adalah sebagai berikut:

1. Formulir atau dokumen yang digunakan di perusahaan.
2. Laporan tercetak yang dihasilkan oleh sistem.
3. Tampilan atau *output* di layar komputer yang dihasilkan oleh sistem.
4. Masukan untuk komputer.
5. Komunikasi ucapan.
6. Surat-surat atau memo.
7. Data yang dibaca atau direkamkan ke suatu *file*.
8. Suatu isian yang dicatat pada buku agenda.
9. Transmisi data dari suatu komputer ke komputer yang lain.

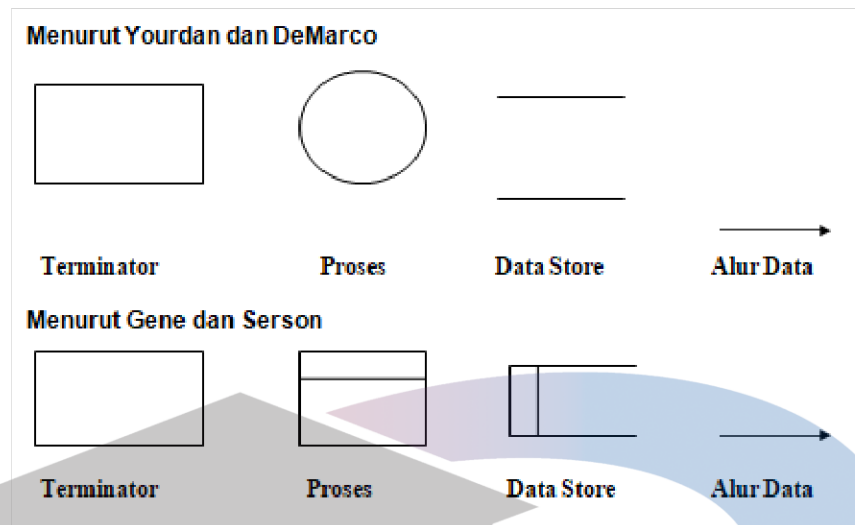
3. *Process/proses*

Bujur sangkar dengan sudut membulat digunakan untuk menunjukkan adanya proses transformasi. Proses-proses tersebut menyatakan proses, pekerjaan atau tindakan yang dilakukan pada data sehingga data yang meninggalkan suatu proses selalu diberi *label* yang berbeda dari aliran data yang masuk. Proses-proses yang menunjukkan hal itu di dalam sistem dan harus diberi nama menggunakan salah satu format berikut ini. Sebuah nama yang jelas memudahkan untuk memahami proses apa yang sedang dilakukan. Pemberian nama pada proses menggunakan format kata kerja/kata sifat/kata benda untuk proses-proses yang *men-detail*. Kata kerja yang menggambarkan jenis kegiatan seperti ini, misalnya menghitung, validasi, menyiapkan, mencetak atau menambahkan.

4. *Data store*

Data store dilambangkan dengan bujur sangkar dengan ujung terbuka. Penyimpanan data (*data store*) sebuah *data store* dapat mewakili salah satu dari banyak lokasi fisik data yang berbeda yang menunjukkan penyimpanan data, seperti *file* atau basis data terkomputerisasi. Karena penyimpanan data mewakili seseorang, tempat atau sesuatu, maka diberi nama dengan sebuah kata benda. Penyimpanan data sementara seperti kertas catatan atau sebuah *file* komputer sementara tidak dimasukkan ke dalam diagram aliran data.

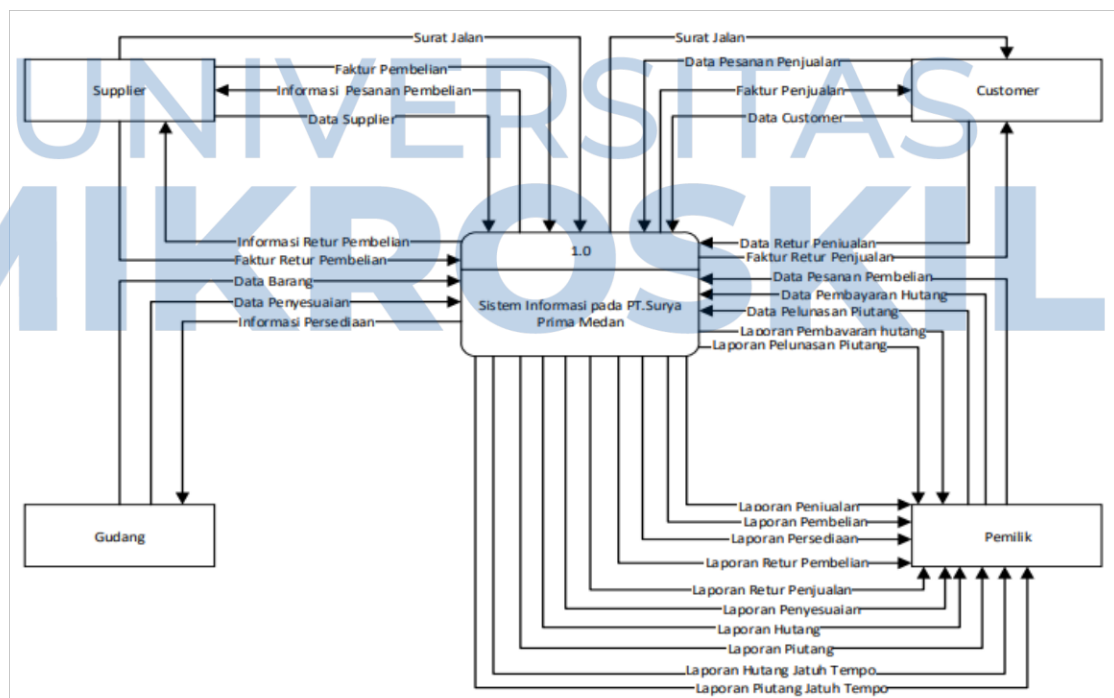
Berikut merupakan simbol-simbol *data flow diagram*:



Gambar 2.4 Simbol-simbol *Data Flow Diagram*

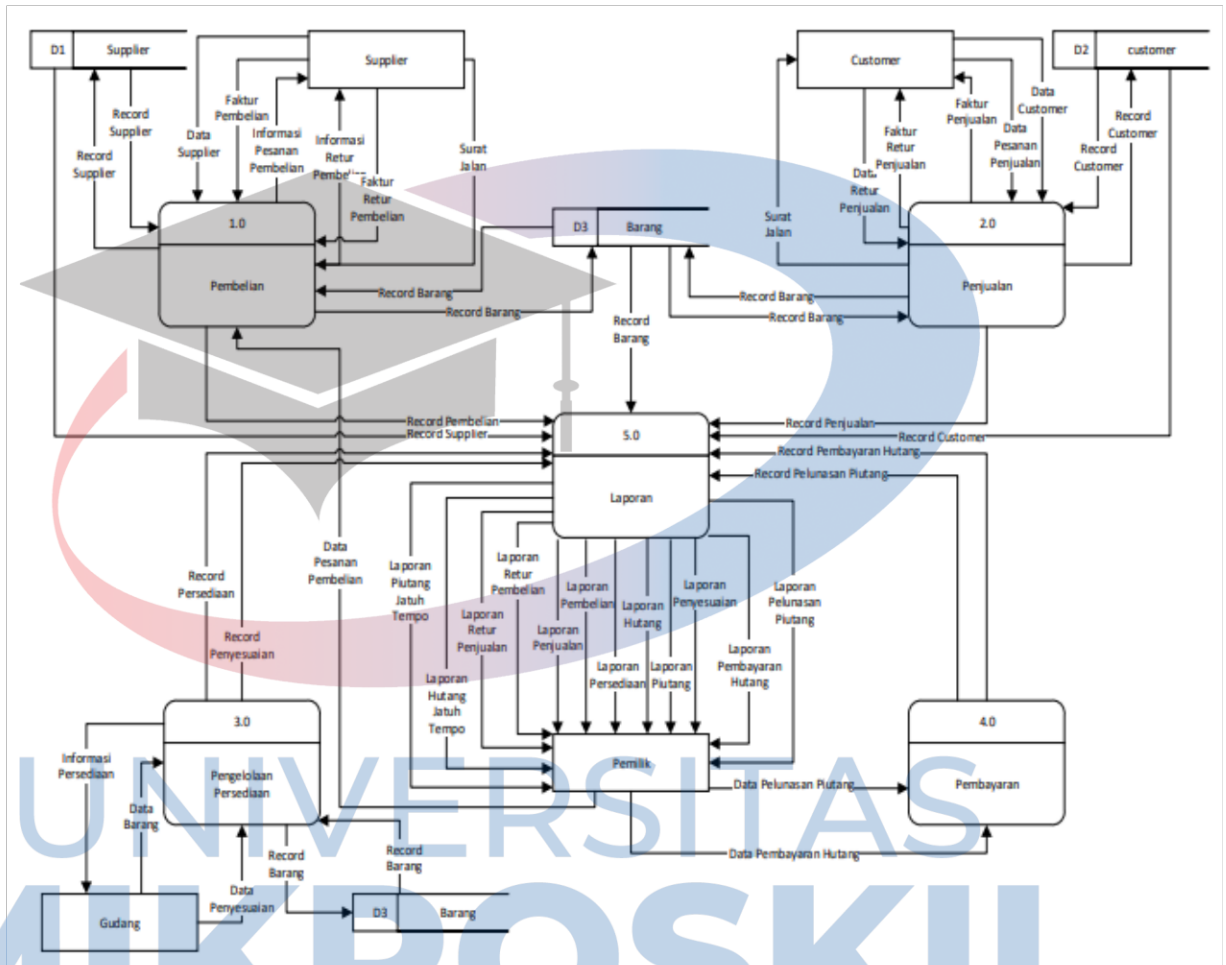
Pedoman penggambaran DFD dapat mengikuti langkah-langkah berikut ini [9]:

1. Identifikasi entitas eksternal yang terlibat
2. Identifikasi semua *input* dan *output* data untuk masing-masing entitas eksternal.
3. Gambarlah terlebih dahulu suatu diagram konteks (*context diagram*). Berikut merupakan contoh diagram konteks (*context diagram*):



Gambar 2.5 Contoh Diagram Konteks

- Gambarlah bagan berjenjang (*hierarchy chart*) untuk mempersiapkan penggambaran DFD ke *level-level* lebih bawah lagi.
- Gambarlah sketsa DFD untuk *overview diagram* (*level 0*). Berikut merupakan contoh *overview diagram* (*level 0*):



Gambar 2.6 Contoh DFD Level 0

- Gambarlah DFD untuk *level-level* berikutnya (1,2, dan seterusnya).
- Gambarlah DFD gabungan semua *level*.

2.5 Kamus Data (*Data Dictionary*)

Kamus Data (*Data Dictionary*) tidak menggunakan notasi grafis sebagaimana halnya DFD, kamus data juga mempunyai fungsi yang sama dalam pemodelan sistem, yaitu sebagai katalog data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi. Sehingga kamus data berfungsi membantu pelaku sistem untuk memahami aplikasi secara *detail*, kamus data mengorganisasikan kembali semua elemen data yang digunakan dalam sistem dengan presisi

yang sedemikian rupa sehingga pemakai dan penganalisa sistem memiliki dasar pengertian yang sama tentang masukan, keluaran, penyimpanan dan proses [2].

Kamus data selain digunakan untuk dokumentasi dan mengurangi redundansi dalam basis data, juga dapat digunakan untuk [2]:

- 1) Validasi keakuratan dan kelengkapan DFD;
- 2) Merencanakan *user interface* baik *input* dan *output*;
- 3) Menentukan muatan data yang disimpan dalam *file-file*; serta
- 4) Mengembangkan logika untuk proses-proses DFD.

Kamus data mendefinisikan elemen data dengan fungsi sebagai berikut [2]:

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 rinci antar penyimpanan (yang akan menjadi titik perhatian dalam *entity-relationship diagram*).

Notasi yang umumnya digunakan dalam menganalisa sistem dengan menggunakan sejumlah simbol yaitu [2]:

Tabel 2.1 Simbol-simbol Kamus Data

No.	Simbol	Uraian
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 <i>field-field</i> numerik pada struktur <i>file</i> .
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 keduanya ada secara 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

Contoh kamus data:

Primary Key : id_pelanggan

Foreign Key : -

Jumlah *field* : 4

Tabel 2.2 Contoh Kamus Data

No.	Nama <i>Field</i>	Type/Lebar	Keterangan
1.	id_pelanggan	<i>Integer</i> (auto increment)	Berisi id atau kode unik pelanggan
2.	nama	<i>Varchar</i> (45)	Berisi nama pelanggan
3.	alamat	<i>Varchar</i> (45)	Berisi alamat pelanggan
4.	notelp	<i>Varchar</i> (12)	Berisi nomor telepon pelanggan

Data pelanggan = id_pelanggan + nama + alamat + notelp

2.6 Kerangka PIECES

PIECES *framework* adalah sebuah metode yang digunakan untuk mengklasifikasikan suatu *problem*, *opportunities*, dan *directives* yang terdapat pada bagian *scope definition*, analisis dan perancangan sistem. Dengan menggunakan metode ini dapat dihasilkan hal-hal baru yang dapat dijadikan pertimbangan dalam pengembangan sistem [10].

Dalam PIECES terdapat enam variabel yang digunakan untuk menganalisis sistem informasi, yaitu [10]:

1. *Performance* (Keandalan)

Variabel ini digunakan untuk mengetahui kinerja sebuah sistem, apakah berjalan dengan baik atau tidak. Kinerja ini dapat diukur dari jumlah temuan data yang dihasilkan dan seberapa cepat suatu data dapat ditemukan.

Tabel 2.3 Contoh Hasil Analisa *Performance*

Parameter	Hasil
Throughput	Penyajian informasi yang dibutuhkan memerlukan waktu yang lama karena banyaknya kegiatan dilakukan secara manual, seperti mencari data pelanggan, <i>input</i> stok barang, transaksi penjualan kepada pelanggan.
Respond Time	Waktu pencarian yang dibutuhkan cukup lambat saat mencari data yang diinginkan di buku arsip.

2. *Information and Data* (Data dan Informasi)

Dalam sebuah temuan data pasti akan dihasilkan sebuah informasi yang akan ditampilkan, variabel ini digunakan untuk menganalisis seberapa banyak dan seberapa jelas informasi yang akan dihasilkan untuk satu pencarian.

Tabel 2.4 Contoh Hasil Analisa *Information*

Parameter	Hasil
Akurat	Tingkat kesalahan pada penyajian informasi masih sering terjadi kesalahan dan kurang terperinci seperti mencari data pelanggan, stok dari barang, dan lain-lain. hal ini disebabkan sistemnya masih bersifat manual.
Relevan	Tingkat relevansinya kurang, karena seharusnya sistem yang dibutuhkan dan dihasilkan harus memberikan informasi yang jelas kepada semua pihak yang terkait atau yang membutuhkan.

Tepat Waktu	Proses pencarian data membutuhkan waktu yang lama, sehingga berpengaruh kepada waktu tunggu informasi yang ingin didapatkan.
-------------	--

3. *Economics* (Nilai Ekonomis)

Variabel ini digunakan untuk melakukan analisis pada sistem, untuk mengetahui apakah suatu sistem tersebut tepat diterapkan pada suatu lembaga informasi dilihat dari segi finansial dan biaya yang dikeluarkan. Hal ini sangat penting karena suatu sistem juga dipengaruhi oleh besarnya biaya yang dikeluarkan.

Tabel 2.5 Contoh Hasil Analisa *Economics*

Parameter	Hasil
Biaya	Pembelian alat tulis, buku, lemari arsip, nota penjualan dan lain-lain yang terus-menerus membutuhkan biaya yang tidak sedikit, belum lagi resiko kehilangan dan kerusakan data. Karena hal ini dilakukan pada buku dan disimpan di lemari arsip.

4. *Control and security* (Pengendalian dan Pengamanan)

Dalam suatu sistem perlu diadakan sebuah kontrol atau pengawasan agar sistem itu berjalan dengan baik. Analisis ini digunakan untuk mengetahui sejauh mana pengawasan dan kontrol yang dilakukan agar sistem tersebut berjalan dengan baik.

Tabel 2.6 Contoh Hasil Analisa *Control and Security*

Parameter	Hasil
Control Sistem	Dalam setiap <i>input</i> -an pemasukan dan pengeluaran barang, laporan penjualan, laporan stok dan lain-lain kurang pengontrolan terhadap setiap laporan, sehingga informasi yang didapatkan kurang akurat.

5. *Efficiency* (Efisiensi)

Efisiensi dan efektivitas sebuah sistem perlu dipertanyakan dalam kinerja dan alasan mengapa sistem itu dibuat. Sebuah sistem harus bisa secara efisien menjawab dan membantu suatu permasalahan khususnya dalam hal otomasi. Analisis ini dilakukan untuk mengetahui apakah suatu sistem itu efisien atau tidak, dengan *input* yang sedikit bisa menghasilkan sebuah *output* yang memuaskan.

Tabel 2.7 Contoh Hasil Analisa *Efficiency*

Parameter	Hasil
Sumber Biaya	Peralatan yang dibutuhkan untuk proses pencatatan laporan antara lain buku, kertas, nota, kwitansi, pulpen, lemari arsip dan sebagainya. Peralatan tersebut biasanya digunakan secara berlebihan ketika salah dalam proses <i>input</i> .
Sumber Tenaga	Proses pencatatan pengadaan atau stok alat sampai ke proses penjualan kepada pelanggan memerlukan beberapa orang dan menghabiskan waktu dan tenaga.

6. *Service* (Pelayanan)

Dalam hal pemanfaat suatu sistem adalah pelayanan masih menjadi suatu hal yang penting dan perlu diperhatikan. Suatu sistem yang diterapkan akan berjalan dengan baik dan seimbang bila diimbangi dengan pelayanan yang baik juga. Analisis ini digunakan untuk mengetahui bagaimana pelayanan yang dilakukan dan mengetahui permasalahan-permasalahan yang ada terkait tentang pelayanan. [10]

Tabel 2.8 Contoh Hasil Analisa *Service*

Parameter	Hasil
Proses Layanan	Proses pelayanan pada sistem berjalan untuk saat ini belum mempermudah penggunaannya, antara lain kepala gudang, bagian gudang, bagian pembelian, <i>accounting</i> , <i>sales marketing</i> , kurir dan pelanggan karena pengolahan data dan laporan masih manual,

	sehingga membutuhkan waktu dan tenaga yang banyak.
--	--

2.7 Basis Data

Basis adalah gudang / markas / tempat berkumpul / tempat bersarang. Data adalah representasi fakta dunia nyata yang mewakili suatu objek (manusia, benda, kejadian, dan lain-lain) yang disimpan dalam bentuk teks, angka, gambar bunyi, simbol, atau kombinasinya [11].

Basis data adalah kumpulan dari *item* data yang saling berhubungan satu dengan lainnya yang diorganisasikan berdasarkan sebuah skema atau struktur tertentu, tersimpan di *hardware* komputer dan dengan *software* digunakan untuk melakukan manipulasi data (diperbaharui, dicari, diolah dengan perhitungan-perhitungan tertentu, dan dihapus) dengan tujuan tertentu [11].

Untuk menentukan kebutuhan-kebutuhan suatu sistem basis data, pertama harus mengenal bagian-bagian lain dari sistem informasi yang akan berinteraksi dengan sistem basis data, termasuk para pemakai yang ada dan para pemakai baru, serta aplikasi-aplikasinya. Kebutuhan-kebutuhan dari para pemakai dan aplikasi inilah yang kemudian dikumpulkan dan dianalisa [11]. Berikut merupakan langkah-langkah dalam menentukan kebutuhan-kebutuhan suatu sistem basis data [11]:

1. Menentukan kelompok pemakai dan bidang-bidang aplikasinya yang artinya menentukan aplikasi utama dan kelompok pengguna yang akan menggunakan basis data. Individu utama pada tiap-tiap kelompok pemakai dan bidang aplikasi yang telah dipilih merupakan peserta utama pada langkah-langkah berikutnya dari pengumpulan dan spesifikasi data.
2. Melakukan peninjauan atas dokumentasi yang ada, berarti dokumen yang ada yang berhubungan dengan aplikasi-aplikasi kita pelajari dan dianalisa. Dokumen-dokumen lainnya (seperti kebijakan-kebijakan, *form*, *report*, dan bagan organisasi) diuji dan ditinjau kembali untuk menguji apakah dokumen-dokumen tersebut berpengaruh terhadap kumpulan data dan proses spesifikasi.
3. Analisa lingkungan operasi dan pemrosesan data mengartikan informasi yang sekarang dan yang akan datang diperinci dan dipelajari. Termasuk juga analisa jenis-jenis transaksi dan frekuensi-frekuensi transaksinya dan juga arus informasi dalam sistem. Informasi tersebut berupa *input-output* data.

4. Membuat dan menyusun daftar pertanyaan dan wawancara yang di mana jawaban pertanyaan-pertanyaan yang telah dikumpulkan dari para pemakai basis data yang berpotensi akan diberikan kepada ketua kelompok (individu utama) untuk dapat diwawancarai sehingga *input* yang banyak dapat diterima dari mereka dengan memperhatikan informasi yang berharga dan mengadakan prioritas.

Basis data relasional merupakan suatu cara untuk mengelola data secara fisik ke dalam memori. Basis data relasional menggunakan tabel dua dimensi yang terdiri atas baris dan kolom untuk memberi gambaran sebuah berkas data. Untuk menerapkan sebuah basis data (yang terdiri atas sejumlah tabel yang saling berhubungan), dibutuhkan perangkat lunak khusus. Perangkat lunak ini disebut sistem pengelola basis data [11].

Ada beberapa macam kunci relasional yaitu [11]:

1. *Super key*

Adalah sebuah atau sekumpulan atribut yang secara unik mengidentifikasi sebuah baris dalam tabel relasi.

2. *Candidate key*

Relasi mungkin mempunyai lebih dari satu *key*. Masing-masing disebut *CANDIDATE KEY*.

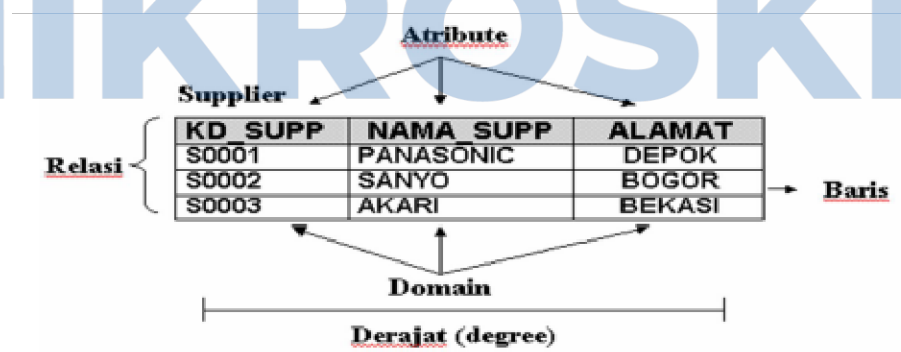
3. *Primary Key*

Adalah *candidate key* yang dipilih sebagai pengidentifikasi unik untuk sebuah tabel relasi.

4. *Alternate key*

Adalah suatu skema relasi dapat memiliki atribut yang menunjuk ke *PRIMARY KEY* dari relasi lain. Atribut ini disebut *FOREIGN KEY*.

Berikut merupakan contoh basis data:



Gambar 2.7 Contoh Basis Data

2.8 Normalisasi

Normalisasi adalah suatu proses memperbaiki atau membangun dengan model data relasional, dan secara umum lebih tepat dikoneksikan dengan model data logika [11].

Proses normalisasi adalah proses pengelompokkan data elemen menjadi tabel-tabel yang menunjukkan entitas dan relasinya. Pada proses normalisasi dilakukan pengujian pada beberapa kondisi apakah ada kesulitan pada saat menambah atau menyisipkan, menghapus, mengubah dan mengakses pada suatu basis data. Bila terdapat kesulitan pada pengujian tersebut maka perlu dipecahkan relasi pada beberapa tabel lagi atau dengan kata lain perancangan basis data belum optimal [11].

Terdapat beberapa tujuan dari normalisasi [11]:

1. Untuk menghilangkan kerangkapan data.
2. Untuk mengurangi kompleksitas.
3. Untuk mempermudah pemodifikasian data.

Proses normalisasi ada dua langkah [11]:

1. Data diuraikan dalam bentuk tabel, selanjutnya dianalisis berdasarkan persyaratan tertentu ke beberapa tingkat.
2. Apabila tabel yang diuji belum memenuhi persyaratan tertentu, maka tabel tersebut perlu dipecah menjadi beberapa tabel yang lebih sederhana sampai memenuhi bentuk yang optimal.

Tahapan dari normalisasi [11]:

1. Bentuk tidak normal

Kumpulan data yang direkam, tidak ada keharusan mengikuti suatu format tertentu sehingga bisa saja datanya tidak lengkap atau terduplikasi. Data dikumpulkan apa adanya sesuai dengan kedatangannya. Berikut merupakan contoh bentuk tidak normal:

KodeBarang	HargaSatuan	KodeJual
NamaBarang	SubtotalHarga	NomorReturBeli
Merk	NomorPO	TanggalReturBeli
HargaJual	KodeBarang	JamReturBeli
Stok	NomorPB	TotalReturBeli
Satuan	TanggalPB	TotalHargaReturBeli
KodePengguna	JamPB	NomorPB
NamaPengguna	TotalPB	KodePengguna
KataKunci	NomorPO	qty_returbeli
HakAkses	KodePemasok	SubtotalReturBeli
KodePemasok	KodePengguna	KodeBarang
NamaPemasok	qty_PB	NomorReturBeli
Alamatpemasok	KodeBarang	NomorReturJual
KotaPemasok	NomorPB	TanggalReturJual
TeleponPemasok	KodeJual	JamReturJual
ContactPerson	TanggalJual	TotalReturJual
FaxPemasok	JamJual	TotalHargaReturJual
NomorPO	TotalHargaJual	KodePengguna
TanggalPO	PPN	KodeJual
JamPO	Diskon	qty_returjual
TotalItemDiorder	TotalBayarJual	SubtotalReturJual
TotalHargaDiorder	KodePengguna	KodeBarang
KodePengguna	qty_jual	NomorReturJual
KodePemasok	Subtotal	
quantity	KodeBarang	

Gambar 2.8 Contoh bentuk tidak normal

2. Bentuk normal pertama (1NF)

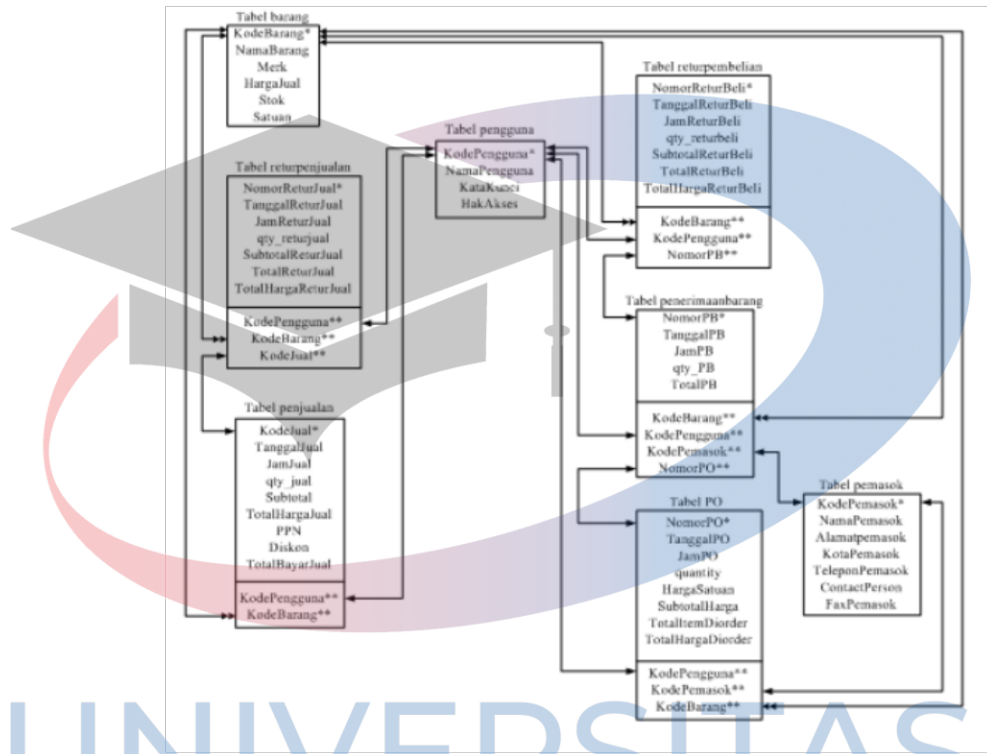
Bentuk normal pertama terpenuhi jika sebuah tabel tidak memiliki atribut yang bernilai banyak (*multi value attribute*) artinya setiap pertemuan baris dan kolom hanya berisikan satu nilai (*single value attribute*). Berikut merupakan contoh bentuk normal pertama (1NF):

KodeBarang*	TanggalPO	PPN
NamaBarang	JamPO	Diskon
Merk	Quantity	TotalBayarJual
HargaJual	HargaSatuan	NomorReturBeli*
Stok	SubtotalHarga	TanggalReturBeli
Satuan	TotalItemDiorder	JamReturBeli
KodePengguna*	TotalHargaDiorder	qty_returbeli
NamaPengguna	NomorPB*	SubtotalReturBeli
KataKunci	TanggalPB	TotalReturBeli
HakAkses	JamPB	TotalHargaReturBeli
KodePemasok*	qty_PB	NomorReturJual*
NamaPemasok	TotalPB	TanggalReturJual
Alamatpemasok	KodeJual*	JamReturJual
KotaPemasok	TanggalJual	qty_returjual
TeleponPemasok	JamJual	SubtotalReturJual
ContactPerson	qty_jual	TotalReturJual
FaxPemasok	Subtotal	TotalHargaReturJual
NomorPO*	TotalHargaJual	

Gambar 2.9 Contoh bentuk normal pertama (1NF)

3. Bentuk normal kedua (2NF)

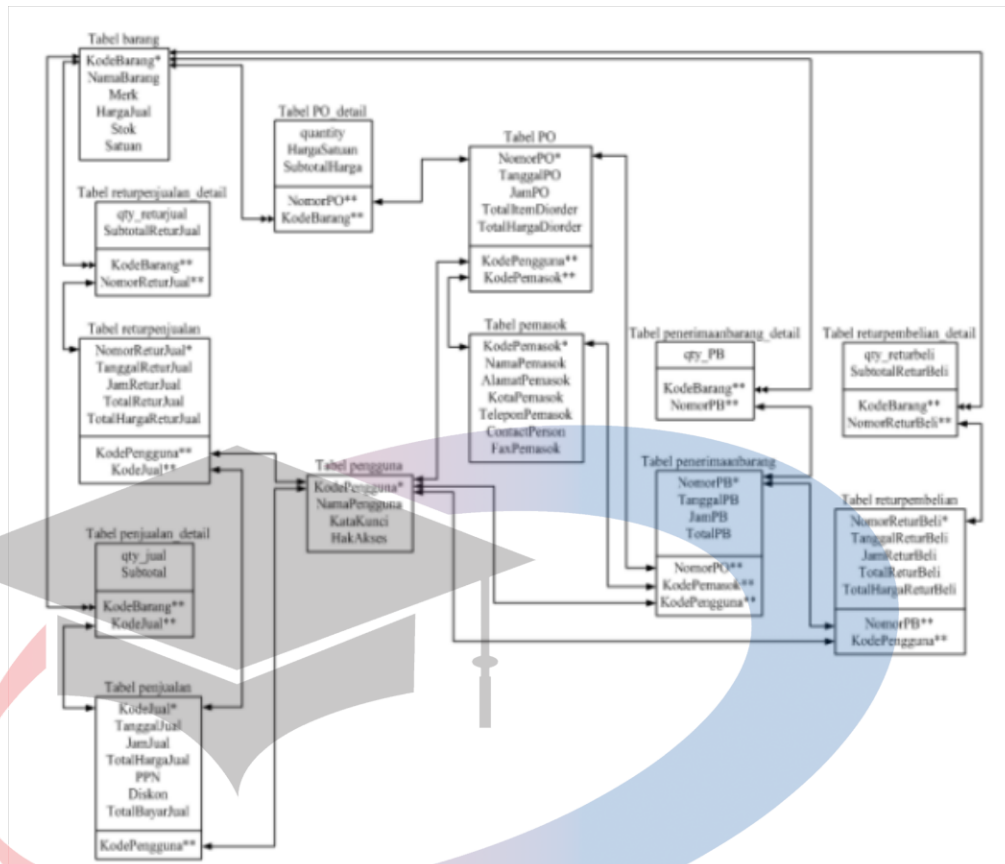
Bentuk normal kedua akan terpenuhi jika bentuk data telah memenuhi kriteria bentuk normal pertama dan setiap atribut yang bukan kunci haruslah bergantung secara fungsional (*functional dependency*) terhadap atribut kunci atau *primary key*. Sehingga untuk membentuk normal ke dua haruslah sudah ditentukan *field* kunci. Berikut merupakan contoh bentuk normal kedua (2NF):



Gambar 2.10 Contoh bentuk normal kedua (2NF)

4. Bentuk normal ketiga (3NF)

Bentuk normal ketiga telah memenuhi kriteria bentuk normal pertama dan kedua serta tidak terdapat *transitive dependency* yaitu sebuah atribut yang bukan kunci selain bergantung kepada atribut kunci, juga bergantung kepada atribut bukan kunci lainnya. Sehingga setiap atribut bukan kunci haruslah bergantung hanya pada atribut kunci (*primary key*). Berikut merupakan contoh bentuk normal ketiga (3NF):



Gambar 2.11 Contoh bentuk normal ketiga (3NF)

5. Bentuk normal Boyce-Codd (BCNF)

BCNF memiliki ketentuan yaitu masing-masing atribut utama bergantung fungsional penuh pada masing-masing kunci di mana kunci tersebut bukan bagiannya. Relasi adalah BCNF (optimal) jika setiap determinan atribut-atribut relasi adalah kunci relasi. Relasi adalah BCNF (optimal) jika kapan pun fakta-fakta disimpan mengenai beberapa atribut, maka atribut-atribut ini merupakan satu kunci relasi. BCNF dapat memiliki lebih dari satu kunci. Properti penting BCNF adalah relasi tidak memiliki informasi yang redundan.

6. Bentuk normal keempat (4NF)

Relasi dalam bentuk normal keempat (4NF) jika relasi dalam BCNF dan tidak berisi ketergantungan banyak nilai. Untuk menghilangkan ketergantungan banyak nilai dari satu relasi, maka perlu membagi relasi menjadi dua relasi baru. Masing-masing relasi berisi dua atribut yang mempunyai hubungan banyak nilai.

7. Bentuk normal kelima (5NF)

Bentuk normal kelima (5NF) berurusan dengan properti yang disebut *join* tanpa adanya kehilangan informasi (*lossless join*). Bentuk normal kelima (5NF) juga disebut PJNF (*Projection-join normal form*). Kasus-kasus ini sangat jarang muncul dan sulit untuk dideteksi secara praktis.

2.9 Penjualan, Pembelian dan Persediaan

Penjualan adalah salah satu fungsi pokok yang dilaksanakan dalam kegiatan pemasaran suatu produk oleh suatu organisasi pemasaran, kegiatan untuk menyampaikan produk barang kepada konsumen. Penjualan adalah proses di mana sang penjual memastikan, mengantisipasi, dan memuaskan kebutuhan atau keinginan sang pembeli agar dapat dicapai manfaat, baik bagi yang menjual maupun bagi sang pembeli yang berkelanjutan dan menguntungkan kedua belah pihak. Penjualan merupakan proses pertukaran barang atau jasa antara penjual dan pembeli, dengan alat tukar berupa uang dan orang yang menjual sesuatu akan mendapatkan imbalan berupa uang [12].

Pembelian merupakan kegiatan menukarkan uang sebagai alat transaksi yang sah dengan barang yang dilakukan oleh dua orang atau lebih. Dengan demikian, pembeli memperoleh barang yang diinginkannya, dan penjual mendapatkan keuntungan atas barang yang dibeli tersebut. Pada perusahaan dagang, pembelian ini dilakukan untuk mendapatkan barang dagangan atau persediaan barang dagangan, yang nantinya akan dijual kembali kepada konsumen [13].

Persediaan (*inventory*) merupakan barang yang dimiliki untuk dijual sebagai bagian dari operasi bisnis normal perusahaan. dengan mengecualikan organisasi jasa tertentu, persediaan merupakan aset yang dibutuhkan dan penting bagi perusahaan. pemeriksaan dengan cermat atas persediaan perlu dilakukan karena persediaan merupakan komponen penting dalam aset operasi dan secara langsung mempengaruhi penentuan laba [13].