

BAB II

KAJIAN LITERATUR

2.1. Konsep Sistem Informasi

2.1.1. Pengertian Sistem

Sistem merupakan suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. [4] Sistem memiliki pendekatan yang ditekankan dalam sebuah prosedur jaringan kerja yang bekerja bersama untuk mendapatkan pencapaian sasaran yang diinginkan. Dalam prosedur terdapat instruksi dengan tahapan – tahapan yang berurutan dimana apa (*what*) yang dikerjakan, siapa (*who*) yang melakukan pekerjaan, kapan (*when*) pengerjaannya dan bagaimana (*how*) cara kerjanya. Pendekatan lebih menekankan pada bagian komponen dengan artian bahwa “sistem” merupakan interaksi dari kumpulan elemen dalam suatu tujuan yang dicapai. [5]

Beberapa karakteristik yang harus dimiliki sebuah sistem yaitu:

1. Komponen (*component*)

Di dalam sistem terdapat beberapa komponen, beberapa diantaranya melakukan interaksi dengan membentuk satu kesatuan dan saling bekerja sama yang terdiri dari berbagai cabang sistem.

2. Lingkungan luar sistem (*Environment*)

Lingkungan luar pada sistem (*environment*) merupakan pengaruh operasi sistem oleh lingkungan luar sistem. Lingkungan luar sangat mempengaruhi, yang bersifat menguntungkan harus di jaga dan yang bersifat merugikan tetap di jaga namun dikendalikan.

3. Batasan sistem (*Boundary*)

Lingkup luar sistem yang dibatasi oleh ruang lingkup (*scope*) atau sistem dengan batas sistem lain yang sesuai bundaran daerahnya.

4. Penghubung sistem (*Interface*)

Penghubung sistem merupakan alat bantu yang menghubungkan antara satu subsistem ke subsistem lainnya. Melalui penghubung sumber-sumber daya dimungkinkan mengalir dari

subsistem ke subsistem lain. Keluaran (*output*) dari subsistem ini akan menjadi masukan (*input*) untuk subsistem dengan alat bantu penghubung ini.

5. Masukkan sistem (*Input*)

Masukkan sistem merupakan sumber daya yang dimasukkan ke dalam sistem, yang dapat berupa perawatan (*maintenance input*), dan masukkan sinyal (*signal input*). *Maintenance input* adalah sumber daya yang dimasukkan agar sistem dapat beroperasi. *Signal input* adalah sumber daya yang diproses untuk didapatkan keluaran. Sebagai contohnya adalah *software* yang merupakan *maintenance* supaya sistem tetap berjalan. Sedangkan *signal input* adalah data sinyal pada proyektor.

6. Keluaran sistem (*Output*)

Keluaran sistem adalah energi yang dihasilkan setelah pemrosesan inputan keluaran yang dibuang maupun dibutuhkan. Contoh sebuah komputer mengeluarkan suhu panas dikatakan sebagai energi buang dan informasi sebagai keluaran energi yang dipakai.

7. Pengolah sistem

Pengolahan sistem merupakan bagian proses yang merubah *input* menjadi *output*. Contohnya sistem akuntansi dengan pengolahan data menjadi laporan-laporan keuangan. Sistem mesin cuci yang merubah baju kotor menjadi bersih.

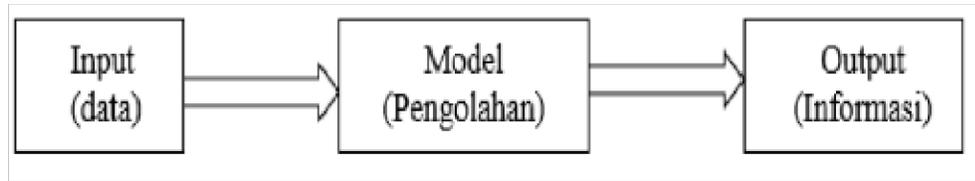
8. Sasaran sistem

Sasaran sistem merupakan tujuan (*goal*) atau sasaran (*objective*).

2.1.2. Pengertian Informasi

Informasi merupakan hasil pengolahan data dengan cara tertentu sehingga lebih berarti dan berguna bagi penerimanya. [5]

Sumber dari informasi berupa data yang menggambarkan kejadian nyata yang telah terjadi pada saat tertentu. Sumber ini perlu diolah melalui sebuah siklus yang dinamakan sebagai siklus pengolahan data (*data processing life cycle*). [5]



Gambar 2.1 Siklus Pengolahan Data (data processing life cycle)

Informasi dikatakan berkualitas jika memenuhi 3 aspek, diantaranya:

1. Akurat (*Accuracy*)

Informasi harus tepat dan tidak bias, terbebas dari kesalahan apapun. Sebuah informasi harus sesuai, bukan merupakan *hoax*, dan tidak ambigu.

2. Tepat waktu (*Timeliness*)

Informasi harus sampai ke penerima dengan waktu yang tepat dan tidak boleh terlambat.

3. Relevan (*Relevance*)

Sebuah informasi yang relevan terjadi ketika informasi tersebut bermanfaat dan memenuhi kebutuhan penerimanya. [5]

2.1.3. Pengertian Sistem Informasi

Sistem informasi merupakan proses pengumpulan, penyimpanan, analisis sebuah informasi dengan tujuan tertentu. Sistem informasi terdiri dari data (*input*) dan menghasilkan laporan (*output*) sehingga di terima oleh sistem lainnya serta kegiatan strategi dalam suatu organisasi dalam melakukan tindakan atau keputusan. [5] Komponen sistem informasi adalah:

1. Perangkat Keras (*Hardware*)

Bagian ini merupakan bagian dari perangkat keras sistem informasi, yang terdiri dari mesin dan media yang digunakan untuk melakukan kegiatan sistem informasi. Contoh *hardware* antara lain: *mainframe*, *minicomputer*, dan *microcomputer*, yang meliputi perangkat *input* data, pengolah data, penyimpanan data dan *output* data / informasi, *peripheral* komunikasi (untuk jaringan komputer), dan komputer lainnya.

2. Perangkat Lunak Sistem (*Software*)

Software seperti sistem operasi, sistem komunikasi dan sistem utilitas, misalnya program *Microsoft Windows*, *LINUX*, *Novel Netware*, *AntiVirus*, *Norton Utilities*, *Disk Doctor*, dan lain-lain.

3. Data dan Informasi

Ini merupakan komponen dasar informasi dan selanjutnya akan diproses untuk menghasilkan informasi.

4. Prosedur

Bagian ini berisi tentang proses atau dokumentasi dari proses yang berlangsung di sistem. Program dapat berupa manual operasi (instruksi), termasuk program sistem pengendalian internal atau manual teknis, seperti manual bagi pengguna untuk menjalankan program komputer dan mempersiapkan input.

5. *Brainware*

Manajer manusia adalah salah satu alat sistem informasi terpenting. Oleh karena itu, hubungan sistem informasi dan pengelola sangat erat. Sistem informasi yang dibutuhkan tergantung pada kebutuhan pengelola. Administrator sistem informasi diatur ke dalam struktur manajemen. [6]

2.1.4. Jenis – Jenis Sistem Informasi

Data mengenai sekumpulan informasi transaksi dalam kegiatan operasional perusahaan merupakan salah satu bentuk pengelolaan data dalam *management information system*. Data yang masuk selanjutnya akan diproses ke dalam sistem entri dengan bukti rekaman atau editan. Pengguna pada proses akhir umumnya akan menginput data ke sistem komputer kemudian membuat catatan terkait kegiatan dalam sistem yang terekam berdasarkan sumber data inputan. Adapun jenis-jenis sistem informasi terdiri dari:

1. *Transaction Processing System (TPS)*

Transaction Processing System adalah sistem informasi yang digunakan untuk transaksi bisnis yang rutin seperti daftar gaji dan inventarisasi yang terkomputerisasi berdasarkan data-data yang diproses dalam jumlah yang besar.

2. *Office Automation System (OAS)* dan *Knowledge Work System (KWS)*

Sistem informasi yang mendukung pekerja data dari segi analisis informasi untuk ditransformasikan dengan cara-cara tertentu sebelum kemudian dibagikan atau disebarluaskan secara menyeluruh dengan organisasi maupun diluar organisasi. Sistem informasi ini pada umumnya memberikan pengetahuan baru bagi pengguna dan masyarakat.

3. Sistem Informasi Manajemen (SIM)

Suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi atau perusahaan tersebut dengan penggunaannya dapat dilakukan kapan saja. Sistem ini mencakup pengolahan transaksi yang terkomputerisasi dengan interaksi antara manusia dengan komputer. Dalam akses *management information system*, pengguna akan mengimplementasikan data-data tersebut sebagai dasar untuk mengambil keputusan.

4. *Decision Support System (DSS)*

Decision Support System merupakan bagian dari sistem informasi yang terkomputerisasi pada level yang lebih tinggi sebagai sistem yang menekankan pembuatan keputusan yang aktual. Sistem ini sangat cocok untuk orang-orang atau kelompok yang bergantung pada basis data sebagai sumber data.

5. Sistem Pakar dan Kecerdasan Buatan

Sistem pakar merupakan salah satu kelas spesial yang digunakan pada praktik bisnis sebagai dampak penggunaan *hardware* dan *software* yang semakin banyak. Secara efektif penggunaan sistem pakar dapat diterapkan dan diimplementasikan seorang ahli yang memiliki kapasitas pengetahuan yang memadai guna menyelesaikan masalah. Selanjutnya, kecerdasan buatan sebagai pendorong untuk mengembangkan penggunaan mesin-mesin yang berfungsi secara cerdas. Secara umum penggunaan sistem pakar digunakan untuk menyelesaikan permasalahan serta memberikan solusi bagi pengguna bisnis.

6. *Group Decision Support System (GDSS)* dan *Computer Supported Collaboration Work System (CSCWS)*

Group Decision Support System (GDSS) digunakan untuk memberikan alternatif keputusan yang terbaik sebagai solusi dalam pengambilan keputusan yang diambil oleh para eksekutif. *Group Decision Support System (GDSS)*, menggunakan ruangan khusus yang dilengkapi dengan beberapa model pengaturan yang berbeda-beda untuk mempermudah interaksi pada saat rapat atau grup diskusi khusus. Selanjutnya *Computer Supported Collaboration Work System (CSCWS)* merupakan sistem yang terhubung dengan jaringan menggunakan komputer sebagai pendukung perangkat lunak.

7. *Executive Support System (ESS)*

Executive Support System berfungsi untuk membantu para pengambil keputusan untuk menemukan serta membantu membuat kebijakan yang paling strategis bagi perusahaan. [7]

2.2. Sistem Informasi Penjualan, Pembelian, dan Persediaan

2.2.1. Sistem Informasi Penjualan

2.2.1.1. Definisi Penjualan dan Sistem Informasi Penjualan

Penjualan adalah suatu proses dimana sang penjual memuaskan atau memenuhi segala kebutuhan dan keinginan pembeli agar dicapai manfaatnya bagi sang penjual maupun sang pembeli yang berkelanjutan dan yang menguntungkan bagi kedua belah pihak. Penjualan juga hasil yang dicapai sebagai imbalan jasa-jasa yang diselenggarakan dan dilakukannya perniagaan transaksi dunia usaha. [8]

Sistem informasi penjualan adalah sebuah prosedur yang melaksanakan, mencatat, mengkalkulasi, membuat dokumen, dan informasi penjualan untuk keperluan manajemen dan bagian lain yang berkepentingan, dari mulainya *order* penjualan hingga transaksi dilaksanakan. [9]

2.2.1.2. Jenis-jenis Penjualan

Jenis-jenis penjualan yang dilakukan oleh sebuah perusahaan adalah [10]:

1. Penjualan Tunai

Penjualan tunai dilaksanakan oleh perusahaan dengan cara mengirim barang terlebih dahulu sesuai dengan *purchase order* yang diterima oleh pembeli atau pelanggan dan pembayarannya pada perusahaan ditagih dalam jangka waktu tertentu sesuai dengan syarat pembayarannya atau jangka waktu tempo yang telah ditentukan, dan pihak pelanggan langsung membayarnya sesuai dengan jumlah harga yang tertera pada dokumen penagihan tersebut tanpa mengangsur.

2. Penjualan Kredit

Penjualan kredit dilaksanakan oleh perusahaan dengan cara mengirim barang terlebih dahulu sesuai dengan *purchase order* yang diterima oleh pembeli atau pelanggan dan pembayarannya pada perusahaan ditagih secara berangsur-angsur sesuai dengan jangka waktu tempo atau sesuai dengan syarat pembayaran yang telah ditentukan. Konsep penjualan adalah pendekatan yang lazim diambil oleh banyak perusahaan terhadap pasar.

3. Penjualan Tender

Penjualan yang dilakukan melalui prosedur tertentu untuk memenuhi permintaan pihak pembeli yang membuka tender tersebut.

4. Penjualan *Export*

Penjualan yang dilaksanakan dengan pembeli dari luar negeri yang mengimpor barang tersebut.

5. Penjualan Konsinyasi

Penjualan secara titipan melalui penjualan lain.

6. Penjualan Grosir

Penjualan yang tidak langsung kepada pembeli tetapi melalui pedagang perantara. Grosir berfungsi sebagai perantara pabrik dengan pedagang.

2.2.2. Sistem Informasi Pembelian

2.2.2.1. Definisi Pembelian dan Sistem Informasi Pembelian

Pembelian adalah suatu sistem kegiatan dalam perusahaan untuk pengadaan barang yang diperlukan oleh perusahaan. [11]

Sistem informasi pembelian dibuat dengan tujuan mengidentifikasi kebutuhan informasi mengenai pembelian yang diperlukan dalam pengambilan keputusan. Dengan adanya sistem informasi pembelian, pimpinan bisa mengetahui pemasok mana yang dipilih untuk membeli barang, kapan jatuh tempo hutang supaya tidak terlambat membayar, kemudian laporan-laporan mengenai pembelian, pemasok, barang-barang yang dibutuhkan, menjaga persediaan agar selalu ada, kapan harus melakukan *reorder point*, akan *up-to-date* dan akurat. Adanya sistem informasi pembelian ini merupakan sarana yang efektif dan mempermudah untuk mengolah data pembelian sampai laporan pembelian. [12]

2.2.2.2. Jenis-jenis Pembelian

Jenis-jenis pembelian berdasarkan transaksi:

1. Transaksi pembelian tunai: adalah pembayaran yang dilakukan secara langsung.
2. Transaksi pembelian kredit: adalah pembayaran yang dilakukan tidak secara langsung pada saat barang diterima, tetapi dilakukan selang beberapa waktu sesuai perjanjian kedua belah pihak. [13]

2.2.3. Sistem Informasi Persediaan

2.2.3.1. Definisi Persediaan dan Sistem Informasi Persediaan

Persediaan merupakan suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam suatu periode tertentu, atau persediaan barang-barang yang masih dalam pengerjaan atau proses produksi, ataupun persediaan bahan baku yang menunggu penggunaannya dalam suatu proses produksi. [8]

Sistem informasi persediaan barang adalah struktur interaksi manusia, peralatan metode-metode, dan kontrol-kontrol yang disusun untuk mencapai tujuan berikut:

- a. Mendukung rutinitas kerja dalam suatu bagian di dalam suatu perusahaan
- b. Mendukung pembuatan keputusan untuk personil-personil yang mengatur gedung dan bagian kontrol persediaan
- c. Mendukung persiapan laporan-laporan internal dan laporan eksternal

Sistem persediaan mendukung rutin kerja dalam bagian kontrol persediaan, yaitu dengan menangkap dan mencatat data yang berhubungan dengan sistem persediaan, misalnya transaksi penerimaan barang dan transaksi penggunaan barang. [14]

2.2.3.2. Jenis-jenis Persediaan

Jenis persediaan biasanya mencakup beberapa jenis persediaan, seperti bahan mentah, persediaan setengah jadi dan persediaan barang jadi siap untuk dijual. Jenis – jenis persediaan dikelompokkan menjadi [8]:

1. Persediaan Bahan Baku (*Raw Material Stock*)

Persediaan bahan baku adalah persediaan dari barang-barang berwujud yang digunakan dalam proses produksi, barang mana dapat diperoleh dari sumber-sumber alam ataupun dibeli dari pemasok atau perusahaan yang menghasilkan bahan baku bagi perusahaan pabrik yang menggunakannya.

2. Persediaan bagian produk atau *parts* yang dibeli (*purchased parts/components stock*)

Persediaan bagian produk atau *parts* yang dibeli adalah persediaan barang-barang yang terdiri dari *parts* yang diterima dari perusahaan lain, yang dapat secara langsung di *assembling* dengan parts lain, tanpa melalui proses produk sebelumnya.

3. Persediaan bahan-bahan pembantu atau barang-barang perlengkapan (*supplies stock*)

Persediaan bahan-bahan pembantu atau barang-barang perlengkapan yaitu persediaan barang-barang atau bahan-bahan yang diperlukan dalam proses produksi untuk membantu keberhasilan produksi, tetapi bukan merupakan bagian atau komponen dari barang jadi.

4. Persediaan barang setengah jadi atau barang dalam proses (*work in process/progress stock*)
Persediaan barang setengah jadi atau barang dalam proses yaitu persediaan barang-barang yang keluar dari tiap-tiap bagian dalam satu pabrik atau bahan-bahan yang telah diolah menjadi suatu bentuk, tetapi lebih perlu diproses kembali untuk kemudian menjadi barang jadi.
5. Persediaan barang jadi (*finished goods stock*)
Persediaan barang jadi yaitu persediaan barang yang telah selesai diproses atau diolah dalam pabrik dan siap untuk dijual kepada pelanggan atau perusahaan lain.

2.2.3.3. Metode Pencatatan Persediaan

Metode pencatatan persediaan biasanya menggunakan metode: [15]

- a. Metode Sistem Periodik (Metode Fisik), adalah metode pencatatan dimana perusahaan tidak mencatat persediaan barang ketika adanya transaksi, baik pemasukan maupun pengeluaran persediaan. Namun baru akan dicatat ketika akhir periode penjualan. Caranya adalah dengan mengecek semua persediaan secara langsung (fisik).
- b. Metode Sistem Perpetual, adalah metode pencatatan dimana perusahaan mencatat persediaan barang setiap adanya transaksi, baik pemasukan maupun pengeluaran persediaan. Sehingga perusahaan sudah tahu berapa jumlah persediaan barang pada saat itu juga.

2.2.3.4. Metode Penilaian Persediaan

Penggunaan metode penilaian persediaan dimaksudkan agar dapat meminimalisasi pembelian barang yang sudah tidak layak jual. Metode penilaian persediaan dibagi menjadi 3, yaitu [15]:

- a. Metode *First in First Out* (FIFO)

Persediaan pertama yang dibeli akan dijual atau digunakan terlebih dahulu sehingga yang tertinggal hanya persediaan yang dibeli atau diproduksi kemudian. Keuntungan menggunakan metode ini adalah mendekati nilai persediaan akhir dengan biaya berjalan. Sedangkan kekurangannya adalah perhitungan laba rugi menjadi tidak akurat karena perbedaan antara modal yang digunakan dengan laba yang dihasilkan cukup besar.

- b. Metode *Last in First Out* (LIFO)

Persediaan yang terakhir dibeli menjadi barang yang pertama kali dijual. Keuntungan dari metode LIFO ini adalah laba rugi yang dihasilkan lebih akurat. Sedangkan kelemahannya ada pada saat laba yang dihasilkan rendah sehingga menyebabkan metode ini tidak diakui oleh pajak.

c. Metode *Average* (Rata-Rata)

Metode ini merupakan solusi penengah dari penggunaan FIFO dan LIFO. Pada *Average*, penentuan harga perolehan persediaan barang dagangan didasarkan pada harga diantara harga persediaan pertama dan harga persediaan terakhir yang masuk. Metode ini lebih mudah dalam perhitungan harga pokok karena tidak perlu memperhatikan stok barang pertama maupun terakhir, serta lebih mudah diterapkan. Kelemahannya adalah laba yang dihasilkan kurang akurat.

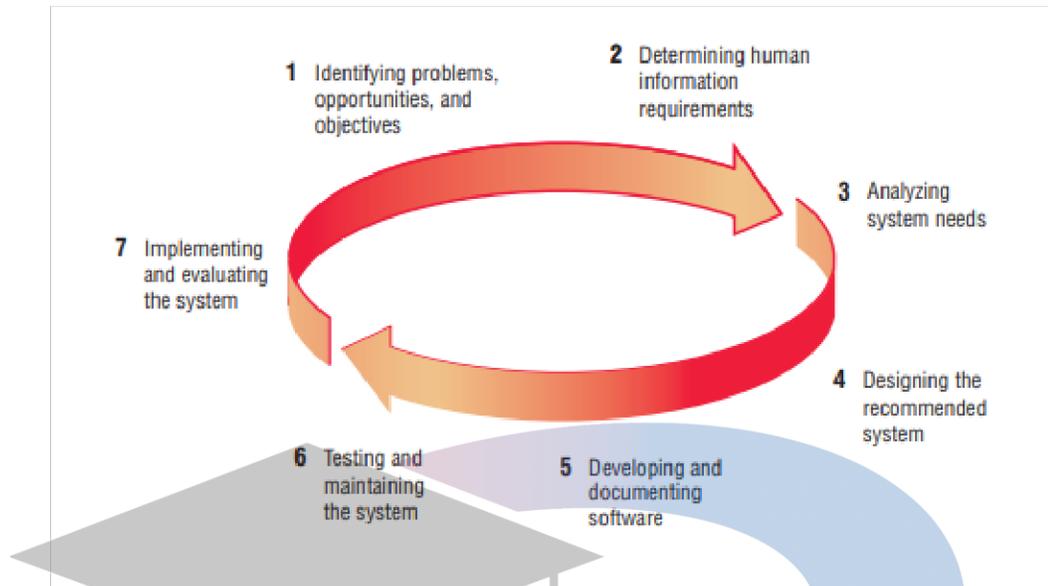
Metode biaya persediaan rata-rata kadang dinamakan dengan metode rata-rata tertimbang (*weighted average*). Metode *weighted average* atau rata-rata tertimbang adalah metode dengan menghitung harga beli barang yang pertama dibeli dan harga beli barang yang terakhir masuk ke gudang untuk kemudian dicari harga rata-rata pembelian barang guna menentukan harga jual atau harga pokok barang yang dijual. Apabila metode ini digunakan untuk menghitung harga pokok penjualan (HPP), maka biaya-biaya dibandingkan terhadap pendapatan sesuai dengan rata-rata per unit harga pokok penjualan. Biaya rata-rata tertimbang per unit yang sama digunakan dalam menentukan biaya persediaan barang dagang pada akhir periode. Metode rata-rata akan mendekati arus fisik barang. [16]

2.3. Siklus Hidup Pengembangan Sistem

Siklus hidup pengembangan sistem (*System Development Life Cycle*) adalah pendekatan bertahap untuk analisis dan desain yang menyatakan bahwa sistem paling baik dikembangkan melalui penggunaan siklus analisis dan pengguna tertentu. [16]

2.3.1. Tahapan dalam SDLC

Terdapat 7 tahap utama dalam siklus hidup pengembangan sistem adalah [16]:



Gambar 2.2 Tahapan-Tahapan Siklus Hidup Pengembangan Sistem

1. Mengidentifikasi Masalah, Peluang, dan Tujuan (*Identifying problems, opportunities, dan objectives*)

Dalam fase pertama SDLC, seorang analis perlu memperhatikan dan mengidentifikasi masalah, peluang, dan tujuan dengan benar. Fase pertama ini mengharuskan analis melihat secara jujur apa saja yang terjadi dalam bisnis. Kemudian, bersama dengan anggota organisasi lainnya, analis menunjukkan masalah. Seringkali orang lain akan mengemukakan masalah ini, dan itulah alasan analis awalnya dipanggil. Peluang adalah situasi yang menurut analis dapat ditingkatkan melalui penggunaan sistem informasi terkomputerisasi. Perebutan peluang memungkinkan bisnis memperoleh daya saing atau menetapkan standar industri. Pertama, analis harus menemukan kegiatan apa yang dilakukan oleh bisnis tersebut. Kemudian analis akan dapat melihat apakah beberapa aspek aplikasi sistem informasi dapat membantu bisnis mencapai tujuannya dengan mengatasi masalah atau peluang tertentu. Pengguna, analis, dan manajer sistem terlibat dalam proyek pada fase pertama ini. Kegiatan seperti mewawancarai manajemen pengguna, meringkas pengetahuan yang diperoleh, memperkirakan ruang lingkup proyek, dan mendokumentasikan hasilnya.

Output dari tahap ini adalah laporan kelayakan yang berisi definisi masalah dan merangkum tujuan. Manajemen harus membuat keputusan apakah akan melanjutkan usulan proyek

tersebut atau tidak. Keberhasilan proyek akan sangat bergantung pada tahap ini untuk mencegah waktu dan biaya yang terbuang karena penanganan masalah yang salah.

2. Menentukan Informasi yang Dibutuhkan Oleh Pengguna (*Determining human information requirements*)

Fase ini mengharuskan analis menentukan kebutuhan SDM dari pengguna yang terlibat dalam sistem, dengan menggunakan berbagai alat bantu untuk memahami bagaimana pengguna berinteraksi dalam konteks kerja dengan sistem informasi mereka. Metode interaktif seperti wawancara, pengambilan sampel dan investigasi data keras, menggunakan kuesioner, mengamati perilaku pengambil keputusan dan lingkungan kantor mereka, dan metode pembuatan prototipe.

Metode-metode ini digunakan untuk menjawab banyak pertanyaan tentang HCI (*Human-Computer Interaction*), seperti tentang kemudahan penggunaan sistem baru, kenyamanan penggunaan, serta kegunaan sistem dalam mendukung produktivitas kerja. Dalam fase persyaratan informasi SDLC, analis perlu untuk memahami informasi yang dibutuhkan pengguna untuk melakukan pekerjaan mereka. Pada titik ini, analis sedang memeriksa bagaimana cara untuk membuat sistem yang berguna bagi orang-orang yang terlibat. Bagaimana sistem dapat mendukung individu dengan lebih baik dan tugas-tugas apa saja yang perlu dilakukan? Tugas baru apa yang diaktifkan oleh sistem baru yang tidak dapat dilakukan tanpa sistem? Bagaimana sistem baru dapat dibuat untuk memperluas kemampuan pengguna di luar apa yang disediakan sistem lama? Bagaimana analis dapat membuat sistem yang bermanfaat untuk digunakan pekerja?

Analisis, pengguna, manajer operasi dan pekerja operasi terlibat pada fase tahap 2 ini. Analisis sistem perlu mengetahui detail fungsi sistem saat ini, seperti *who* (orang yang terlibat), *what* (aktivitas bisnis), *where* (di mana pekerjaan berlangsung), *when* (waktu), dan *how* (bagaimana prosedur saat ini dilakukan) dari bisnis yang diteliti. Analisis kemudian bertanya alasan bisnis menggunakan sistem saat ini. Mungkin ada alasan bagus untuk melakukan bisnis dengan menggunakan metode saat ini, dan ini harus dipertimbangkan ketika akan merancang sistem yang baru. Namun, jika alasan untuk operasi saat ini adalah "selalu dilakukan seperti itu", maka analisis mungkin akan memperbaiki prosedur. Pada penyelesaian fase ini, analis harus memahami bagaimana pengguna menyelesaikan

pekerjaan mereka dengan komputer dan mengetahui bagaimana membuat sistem baru yang lebih berguna. Analis juga harus tahu bagaimana bisnis berfungsi dan memiliki informasi lengkap tentang orang, tujuan, data, dan prosedur yang terlibat.

3. Menganalisis Kebutuhan Sistem (*Analyzing system needs*)

Dalam menganalisis kebutuhan sistem, terdapat alat dan teknik khusus untuk membantu analis dalam membuat penentuan persyaratan. Alat seperti diagram aliran data (DFD) digunakan untuk memetakan *input*, proses, dan *output* bisnis fungsi, atau diagram aktivitas/diagram urutan untuk menunjukkan urutan peristiwa, menggambarkan sistem dalam bentuk grafik yang terstruktur. Dari aliran data, urutan, atau diagram lainnya, kemudian dikembangkannya kamus data yang mencantumkan semua item data yang digunakan dalam sistem, serta spesifikasi mereka.

Selama fase ini, analis sistem juga menganalisis keputusan terstruktur yang dibuat. Keputusan terstruktur disini maksudnya dimana kondisi, alternatif kondisi, tindakan, dan aturan tindakan dapat ditentukan. Ada tiga alat utama untuk menganalisis keputusan terstruktur, yaitu Bahasa Inggris yang terstruktur (*structured English*), tabel keputusan (*decision tables*), dan pohon keputusan (*decision trees*). Pada titik ini, analis sistem menyiapkan proposal sistem yang meringkas apa yang telah ditemukan tentang pengguna, kegunaan, dan kegunaan sistem saat ini; menyediakan biaya analisis manfaat dari alternatif; dan membuat rekomendasi tentang apa (jika ada) yang harus dilakukan.

Jika salah satu rekomendasi dapat diterima oleh manajemen, analis akan melanjutkan proposal tersebut. Setiap masalah sistem itu unik, dan tidak pernah hanya satu solusi yang tepat. Cara di mana rekomendasi atau solusi dirumuskan tergantung pada kualitas individu dan pelatihan profesional masing-masing analis dan interaksi analis dengan pengguna dalam konteks lingkungan kerja mereka.

4. Merancang Sistem yang Direkomendasikan (*Designing the recommended system*)

Dalam fase ini, analis sistem menggunakan informasi yang dikumpulkan sebelumnya untuk menyelesaikan desain logis dari sistem informasi. Analis merancang prosedur bagi pengguna untuk membantu mereka memasukkan data secara akurat sehingga data yang masuk ke sistem informasi benar. Analis juga menggunakan teknik dengan *form* yang baik

dan desain halaman web atau tampilan layar pengguna untuk melengkapi masukan yang efektif untuk sistem informasi tersebut.

Bagian dari desain logis dari sistem informasi adalah merancang HCI. *User Interface*, yang menghubungkan pengguna dengan sistem, dirancang dengan bantuan pengguna untuk memastikan bahwa sistem dapat didengar, terbaca, dan aman, serta menarik dan menyenangkan untuk digunakan. Contohnya seperti *keyboard* (untuk mengetik pertanyaan dan jawaban), menu di layar (untuk mendapatkan perintah pengguna), dan berbagai *user interface* grafis (GUI) yang menggunakan mouse atau layar sentuh. Fase desain juga mencakup perancangan *database* yang akan menyimpan banyak data yang dibutuhkan oleh para pengambil keputusan dalam organisasi. Pengguna dapat memanfaatkan *database* yang terorganisir dengan baik dan logis dan menyesuaikannya dengan pekerjaan mereka. Pada fase ini, analis juga bekerja sama dengan pengguna untuk merancang *output* (baik di layar atau dicetak) yang memenuhi kebutuhan informasi mereka. Akhirnya, analis harus merancang kontrol dan prosedur cadangan untuk melindungi sistem dan data, serta untuk menghasilkan paket spesifikasi program untuk *programmer*.

5. Mengembangkan dan Mendokumentasikan Perangkat Lunak (*Developing and documenting software*)

Di fase ini, analis bekerja dengan *programer* untuk mengembangkan sumber asli perangkat lunak yang dibutuhkan. Selama fase ini analis bekerja dengan pengguna untuk mengembangkan *software* agar mendapatkan dokumentasi yang efektif, termasuk di dalamnya prosedur manual, bantuan *online*, dan situs web yang menampilkan pertanyaan yang sering diajukan. Tahap dokumentasi yang melibatkan pengguna, harus dijawab berdasarkan apa yang telah mereka ajukan dan selesaikan bersama dengan analis. Dokumentasi dilakukan untuk agar pengguna dapat mengetahui cara menggunakan perangkat lunak dan apa yang harus dilakukan jika terjadi masalah pada perangkat lunak. Pemrogram memiliki peran penting dalam fase ini karena mereka mendesain, membuat kode, dan menghilangkan kesalahan *syntax* dari program komputer. Untuk memastikan kualitas, *programmer* dapat melakukan sebuah desain atau panduan kode dan menjelaskan bagian kompleks dari program ke *programmer* lainnya.

6. Menguji dan Mempertahankan Sistem (*Testing and maintaining the system*)

Sebelum suatu sistem informasi dapat digunakan, maka harus dilakukan pengujian. Melakukan pengujian dapat mengurangi biaya sehingga dapat meminimalkan masalah sebelum sistem digunakan pengguna. Beberapa pengujian diselesaikan oleh *programer* sendiri dan beberapa di antaranya oleh analis sistem dalam hubungannya dengan *programer*. Serangkaian pengujian untuk menunjukkan masalah dijalankan pertama kali dengan data sampel dan akhirnya dengan data aktual dari sistem. Seringkali rencana uji dibuat di awal SDLC dan disempurnakan sebagai proyek.

Pemeliharaan sistem dan dokumentasinya dimulai pada fase ini dan dilakukan secara rutin sepanjang umur sistem informasi. Sebagian besar pekerjaan rutin *programer* adalah melakukan pemeliharaan yang menghabiskan banyak uang perusahaan. Beberapa pemeliharaan, seperti pembaruan program, dapat dilakukan secara otomatis melalui situs *vendor* di *Web*. Banyak prosedur sistematis yang digunakan analis di seluruh SDLC dapat membantu memastikan bahwa pemeliharaan dijaga seminimal mungkin.

7. Mengimplementasikan dan Mengevaluasi Sistem (*Implementing and evaluating the system*)

Pada tahap terakhir, analis melibatkan pelatihan pengguna untuk menangani sistem. *Vendor* melakukan beberapa pelatihan, dan pengawasan pelatihan. Selain itu, analis perlu merencanakan konversi yang lancar dari sistem lama ke sistem baru. Proses ini termasuk mengonversi *file* dari *format* lama ke *format* baru, membangun *database*, memasang peralatan, dan membawa sistem baru ke dalam produksi.

Evaluasi dimasukkan sebagai bagian dari fase akhir SDLC untuk kepentingan diskusi. Sebenarnya, evaluasi berlangsung pada setiap fase. Kriteria utama yang harus dipenuhi adalah apakah pengguna yang ditargetkan benar-benar menggunakan sistem atau tidak. Perlu dicatat bahwa kerja sistem seringkali bersifat siklus. Ketika seorang analis menyelesaikan satu fase pengembangan sistem dan melanjutkan ke fase berikutnya, penemuan masalah dapat memaksa analis untuk kembali ke fase sebelumnya dan memodifikasi pekerjaan yang telah dilakukan di sana.

2.4. Konsep Dasar *Database Management System* (DBMS)

2.4.1. Perancangan *Database*

Perancangan *database* adalah proses untuk menentukan isi dan pengaturan data yang dibutuhkan untuk mendukung berbagai rancangan sistem. Tahapan dalam perancangan *database* adalah [18]:

1. Perancangan Database Konseptual (*Conceptual database design*)

Conceptual database design adalah proses membangun model data yang digunakan di dalam suatu organisasi, seperti ERD (*Entity Relationship Diagram*). Tahap desain basis data konseptual yang dimulai dengan membuat model data konseptual dari organisasi dengan rincian implementasi seperti target DBMS, program aplikasi, bahasa pemrograman, *hardware platform*, *performance* dan segala pertimbangan fisik lainnya. Tahapan perancangan basis data konseptual adalah:

a. Identifikasi Tipe Entitas

Tahap pertama adalah mengidentifikasi tipe entitas. Cara untuk mengidentifikasi entitas adalah dengan memperhatikan kata-kata benda pada daftar kebutuhan pengguna. Dokumentasi dari tahap ini meliputi daftar entitas, keterangan entitas, dan nama lain entitas tersebut. Tujuan dari tahap ini adalah untuk menentukan tipe entitas yang dibutuhkan.

b. Identifikasi Tipe Rasional

Tujuan dari tahap ini adalah untuk menentukan hubungan-hubungan penting yang ada antara jenis-jenis entitas yang telah diidentifikasi sebelumnya. Tipe relasi dapat diidentifikasi melalui tahapan menentukan *multiplicit* dari tipe hubungan.

2. Perancangan *Database* Logikal

Perancangan *database* logikal adalah suatu proses membangun sebuah model dari informasi yang digunakan di perusahaan berdasarkan sebuah model data spesifik. Perancangan basis data logikal mempunyai tujuan untuk membangun dan memvalidasi relasi menggunakan normalisasi. Tujuan dari langkah ini adalah untuk meminimalkan kemungkinan terjadinya data rangkap, menghindari data yang tidak konsisten terutama bila dilakukan penambahan atau penghapusan data sebagai akibat karena adanya data yang rangkap, dan untuk menjamin bahwa identitas tabel secara tunggal sebagai determinan semua atribut.

3. Perancangan *Database* Fisikal

Perancangan *database* fisik merupakan proses pembuatan deskripsi dari implementasi *database* pada penyimpanan sekunder yang menjelaskan relasi dasar, *file organization*, dan indeks yang digunakan untuk mencapai akses yang efisien ke data, dan setiap *integrity constraint* yang saling berhubungan dan juga pengukuran keamanan (*security*).

2.4.2. Definisi *Database Management System* (DBMS)

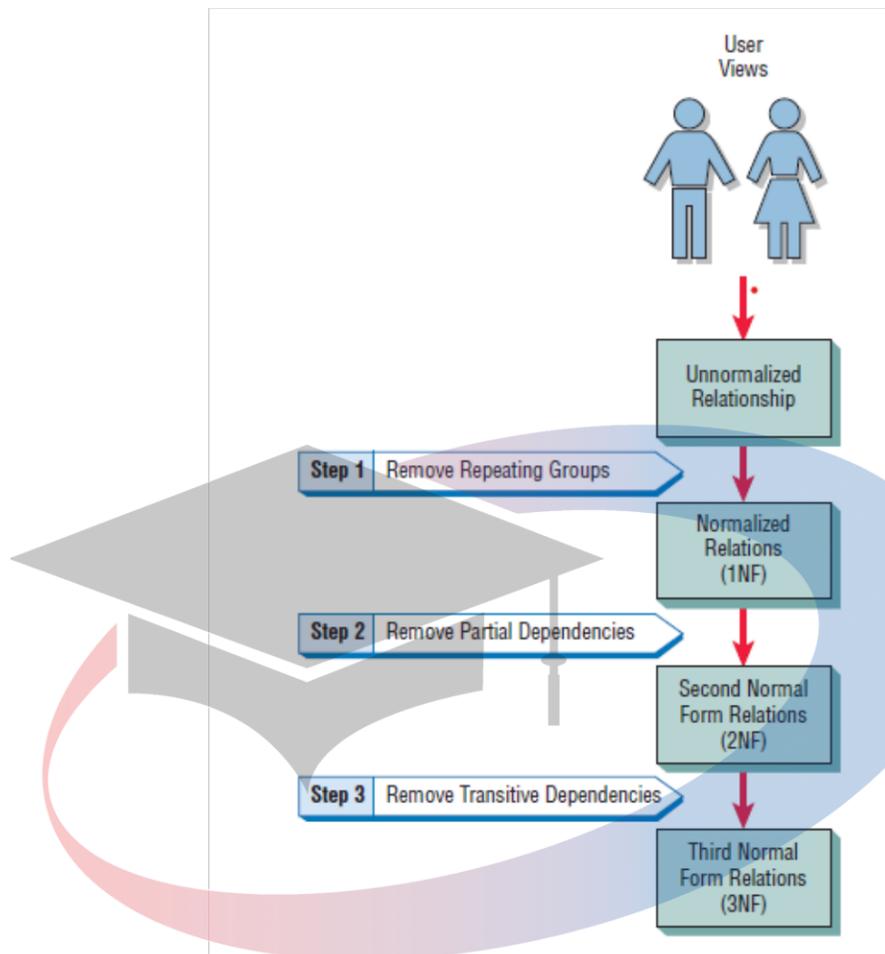
Database Management System (DBMS) merupakan perangkat lunak untuk mengendalikan pembuatan, pemeliharaan, pengolahan, dan penggunaan data yang berskala besar. Penggunaan DBMS saat ini merupakan hal yang sangat penting dalam segala aspek, baik itu dalam skala yang besar atau kecil. Orang yang memastikan bahwa *database* memenuhi tujuannya disebut *administrator database*. Adapun Bahasa dalam DBMS sebagai berikut: [19]

1. *Data Definition Language* (DDL)
2. *Data Manipulation Language* (DML)
3. *Data Control Language* (DCL)

2.4.3. Normalisasi

Normalisasi merupakan proses transformasi tampilan data pengguna yang kompleks dan penyimpanan data menjadi struktur data yang lebih kecil dan stabil sehingga lebih mudah dipelihara daripada struktur lainnya sehingga menghasilkan hubungan-hubungan yang sesuai dengan kebutuhan perusahaan. [17]

UNIVERSITAS
MIKROSKIL



Gambar 2.3 Tahapan-tahapan Normalisasi

Tahap Normalisasi dimulai dari tahap paling ringan (1NF) hingga paling ketat (5NF). Biasanya hanya sampai pada tingkat 3NF atau BCNF karena sudah cukup memadai untuk menghasilkan tabel-tabel yang berkualitas baik. Sebuah tabel dikatakan baik (efisien) atau normal jika memenuhi 3 kriteria sebagai berikut [20]:

1. Jika ada dekomposisi (penguraian) tabel, maka dekomposisinya harus dijamin aman (*Lossless-Join Decomposition*). Artinya, setelah tabel tersebut diuraikan /didekomposisi menjadi tabel-tabel baru, tabel-tabel baru tersebut bisa menghasilkan tabel semula dengan sama persis
2. Terpeliharanya ketergantungan fungsional pada saat perubahan data (*Dependency Preservation*)
3. Tidak melanggar *Boyce-Code Normal Form* (BCNF)

Tahapan-tahapan normalisasi adalah [20]:

1. *Unnormalization Form*

Bentuk yang tidak normal dimaksudkan suatu kumpulan data yang akan diolah yang diperoleh dari format-format yang beraneka ragam, masih terdapat duplikasi atau pengulangan data, bisa saja tidak sempurna atau tidak lengkap, dan sesuai fakta lapangan. Bentuk ini didapat dari dokumen yang ada di lapangan atau manual dengan atribut bukan nilai sederhana.

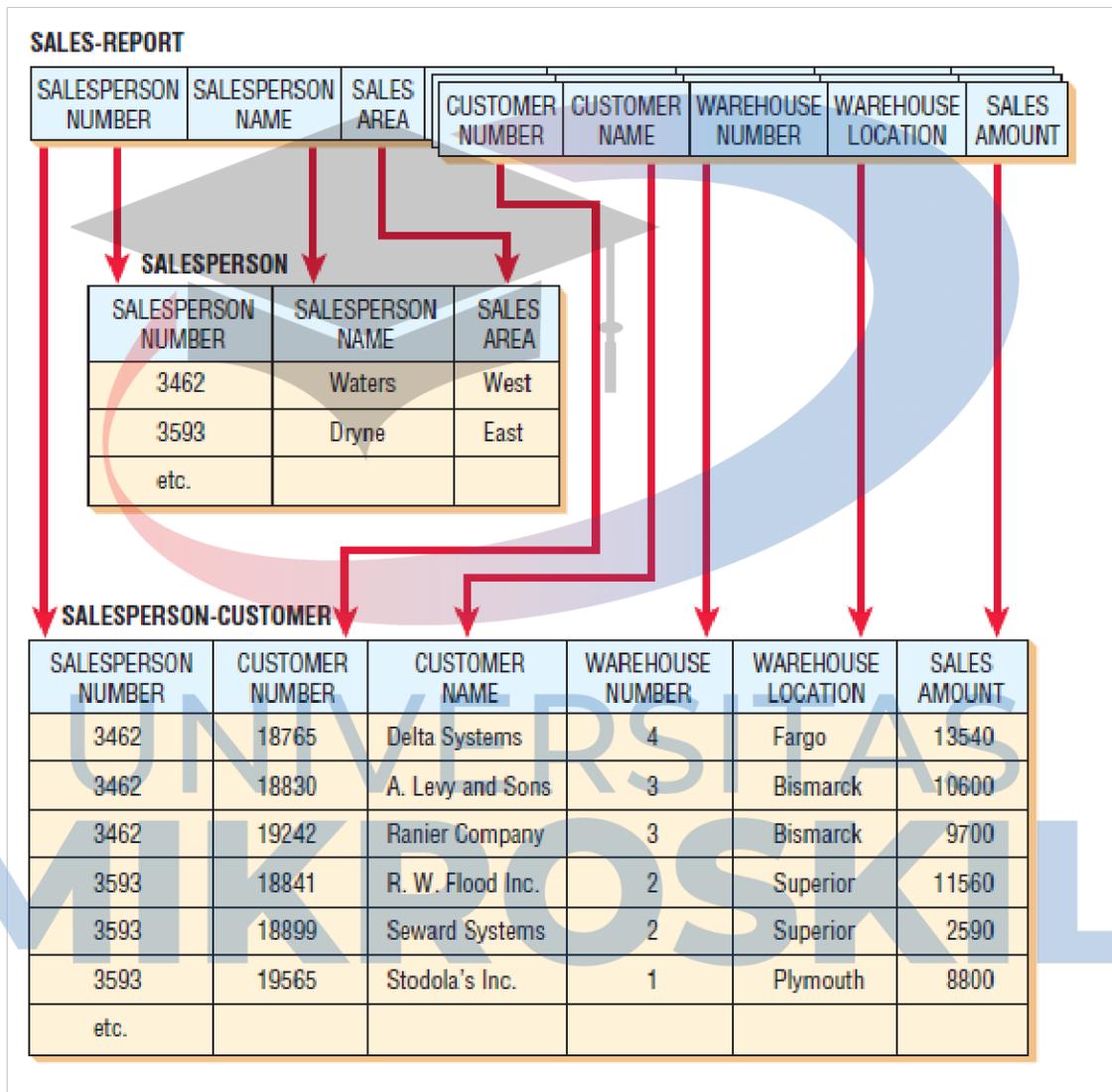
Tabel Mahasiswa							
nim	nama	prodi	kode mtk	nama mtk	id dosen	nama dosen	nilai
1234	Roma	TI	TI4801	Sistem Basis Data	SSD	Surya	A
			TI4815	Rekayasa Perangkat Lunak	RNW	Ronal	C
2345	Beni	SI	TI4801	Sistem Basis Data	SSD	Surya	B
			UN121	Kalkulus	WHY	Wahyu	B
			UN125	Bahasa Indonesia	SAB	Sabrina	A

Gambar 2.4 Tabel *Unnormalization form* (UNF)

2. Bentuk Normal Tahap Pertama (*1st Normal Form / 1NF*)

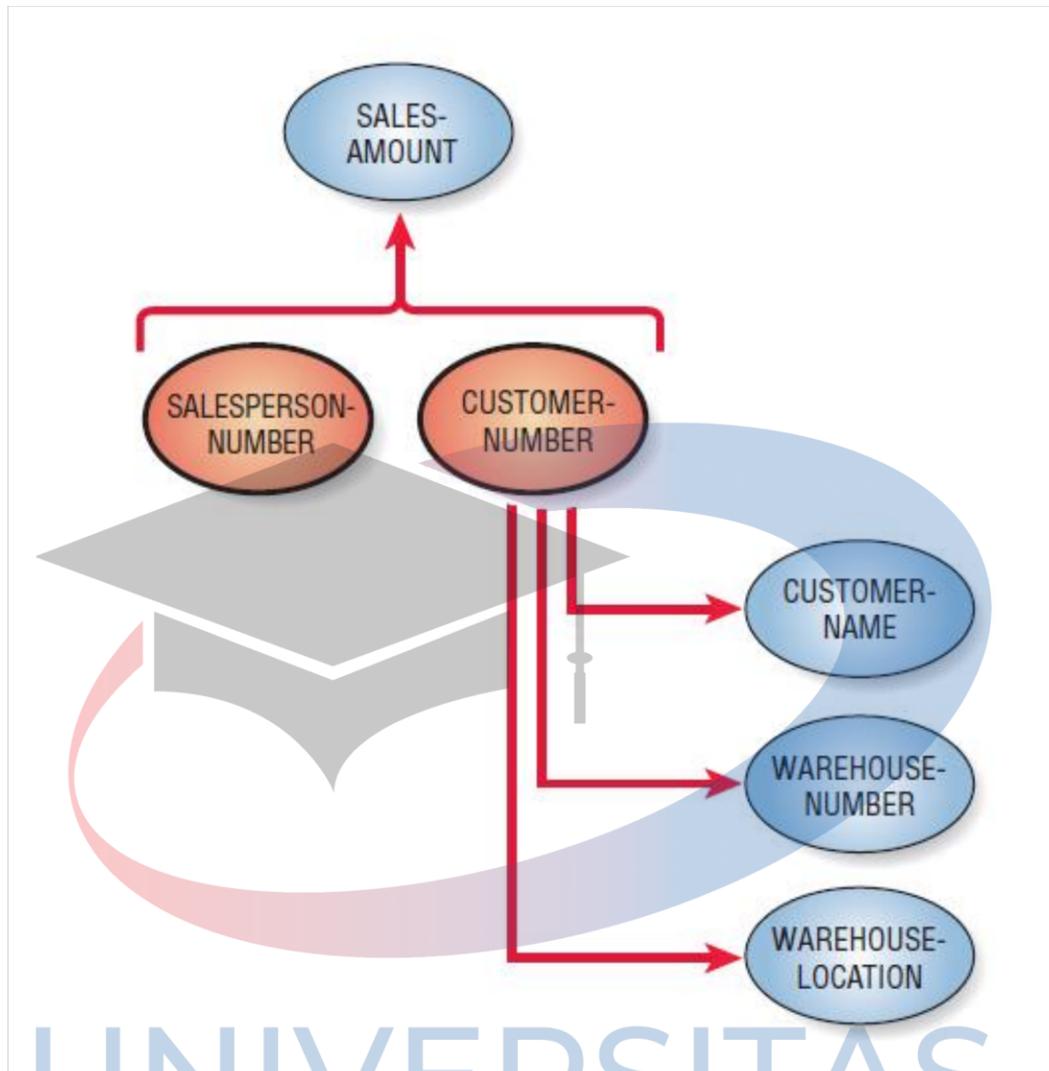
Langkah pertama adalah menghilangkan kelompok berulang. Misalnya seperti gambar dibawah ini. Relasi SALES-REPORT yang tidak dinormalisasi dipecah menjadi dua hubungan yang terpisah. Hubungan baru ini akan diberi nama SALESPERSON dan SALESPERSON-CUSTOMER. Relasi asli yang tidak dinormalisasi laporan penjualan dinormalisasi dengan memisahkan relasi menjadi dua relasi baru. Relasi pertama, SALESPERSON berisi kunci utama SALESPERSON-NUMBER dan semua atribut yang tidak berulang (SALESPERSON-NAME dan SALES-AREA). Relasi kedua, SALESPERSON-CUSTOMER, berisi kunci utama dari relasi SALESPERSON (kunci utama SALESPERSON adalah SALESPERSON-NUMBER), serta semua atribut yang merupakan bagian dari grup berulang (CUSTOMER NUMBER, CUSTOMER NAME, WAREHOUSE NUMBER, WAREHOUSE LOCATION, DAN SALES AMOUNT). Mengetahui SALESPERSON-NUMBER, kita harus menggunakan kunci gabungan (SALESPERSON-NUMBER dan CUSTOMER-NUMBER) untuk mengakses informasi lainnya. Hal ini dimungkinkan untuk menulis hubungan dalam notasi singkat sebagai berikut: SALESPERSON (SALESPERSON NUMBER, SALESPERSON NAME, SALES

AREA). Relasi SALESPERSON-CUSTOMER adalah relasi normal pertama, tetapi tidak ideal. Masalah muncul karena beberapa atribut tidak bergantung secara fungsional pada yang utama kunci (yaitu, SALESPERSON-NUMBER, CUSTOMER-NUMBER). Dengan kata lain, beberapa dari atribut bukan kunci hanya bergantung pada NOMOR PELANGGAN dan tidak pada gabungan kunci.



Gambar 2.5 Contoh 1NF

Diagram model data dibawah menunjukkan bahwa SALES-AMOUNT adalah dependen pada SALESPERSON-NUMBER dan CUSTOMER-NUMBER, tetapi tiga atribut lainnya hanya bergantung pada CUSTOMER-NUMBER.



Gambar 2.6 Diagram Model 1NF

Adapun ciri-ciri bentuk normal 1NF adalah :

- Jika sebuah tabel tidak memiliki atribut bernilai banyak (*multivalued attribute*), atau dengan kata lain harus bernilai tunggal
- Jika sebuah tabel tidak memiliki atribut *composite* atau kombinasinya dalam domain data yang sama. Setiap atribut dalam tabel tersebut harus bernilai *atomic* (tidak dapat dibagi-bagi lagi)
- Jika sebuah tabel tidak memiliki atribut turunan/*derivatied value*
- Jika sebuah tabel tidak memiliki *record* yang bernilai ganda/*redundancy*
- Atribut *composite* atau kombinasinya dalam domain data yang sama
- Setiap atribut dalam tabel tersebut harus bernilai *atomic* (tidak dapat dibagi-bagi lagi)

Tabel Mahasiswa							
nim	nama	prodi	kode_mtk	nama_mtk	id_dosen	nama_dosen	nilai
1234	Roma	TI	T14801	Sistem Basis Data	SSD	Surya	A
1234	Roma	TI	T14815	Rekayasa Perangkat Lunak	RNW	Ronal	C
2345	Beni	SI	T14801	Sistem Basis Data	SSD	Surya	B
2345	Beni	SI	UN121	Kalkulus	WHY	Wahyu	B
2345	Beni	SI	UN125	Bahasa Indonesia	SAB	Sabrina	A

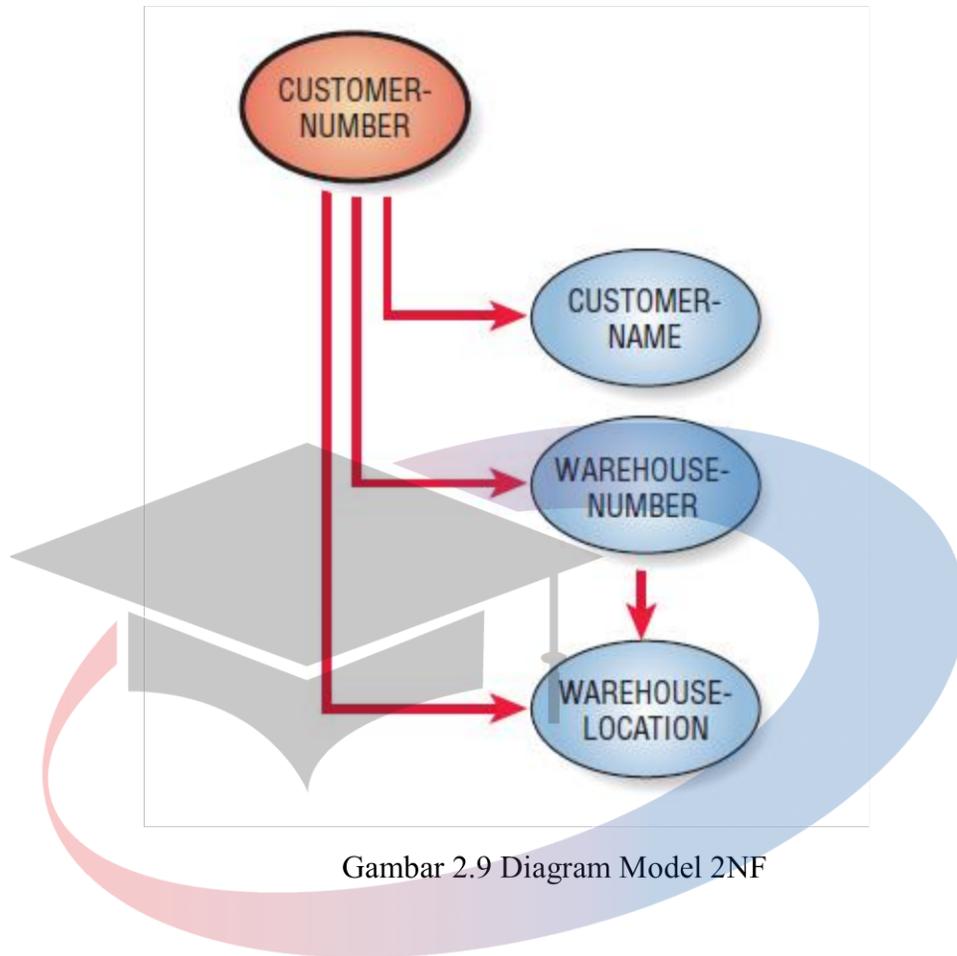
Gambar 2.7 Tabel *1st Normal Form* / 1NF

Dari gambar diatas, masih terdapat atribut yang muncul secara berulang, untuk itu harus melanjutkan ke tahap normalisasi kedua.

3. Bentuk Normal Tahap Kedua (*2nd Normal Form* / 2NF)

Di langkah kedua, semua atribut akan menjadi secara fungsional tergantung pada kunci utama. Oleh karena itu, langkah selanjutnya adalah menghapus semua sebagian atribut dependen dan menemukannya dalam relasi lain. Gambar di bawah ini menunjukkan bagaimana relasi SALESPERSON-CUSTOMER dibagi menjadi dua hubungan baru: SALES dan CUSTOMER-WAREHOUSE. Hubungan tersebut juga dapat dinyatakan sebagai berikut: SALES (SALESPERSON NUMBER, CUSTOMER NUMBER, SALES AMOUNT). Hubungan CUSTOMER-WAREHOUSE berada dalam bentuk normal kedua. Masih bisa disederhanakan lebih lanjut karena ada dependensi tambahan dalam relasi. Beberapa nonkey atribut bergantung tidak hanya pada kunci utama, tetapi juga pada atribut bukan kunci. Ketergantungan ini disebut ketergantungan transitif.

UNIVERSITAS
MIKROSKIL



Gambar 2.9 Diagram Model 2NF

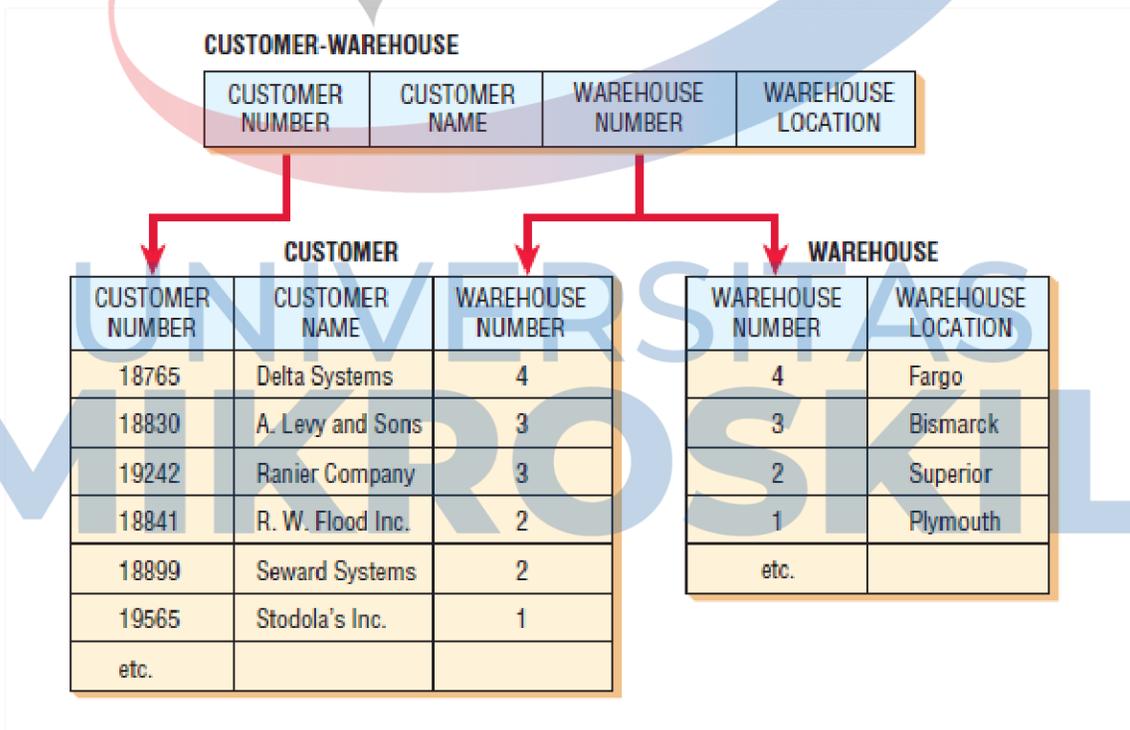
- a. Bentuk normal 2NF terpenuhi dalam sebuah tabel jika telah memenuhi bentuk 1NF, dan semua atribut selain *primary key*, secara utuh memiliki *functional dependency* pada *primary key*
- b. Sebuah tabel tidak memenuhi 2NF, jika ada atribut yang ketergantungannya (*functional dependency*) hanya bersifat parsial saja (hanya tergantung pada sebagian dari *primary key*)
- c. Jika terdapat atribut yang tidak memiliki ketergantungan terhadap *primary key*, maka atribut tersebut harus dipindah atau dihilangkan.

Tabel Kuliah			
<u>kode_mtk</u>	<u>nama_mtk</u>	<u>id_dosen</u>	<u>nama_dosen</u>
TI4801	Sistem Basis Data	SSD	Surya
TI4815	Rekayasa Perangkat Lunak	RNW	Ronal
UN121	Kalkulus	WHY	Wahyu
UN125	Bahasa Indonesia	SAB	Sabrina

Gambar 2.10 Tabel 2nd Normal Form / 2NF

4. Bentuk Normal Tahap Ketiga (*3rd Normal Form / 3NF*)

Suatu relasi yang dinormalisasi berada dalam bentuk normal ketiga jika semua nonkey atribut sepenuhnya bergantung secara fungsional pada kunci utama dan tidak ada transitif (nonkey) dependensi. Memisahkan relasi CUSTOMER-WAREHOUSE menjadi dua hubungan, seperti terlihat pada gambar dibawah. Kedua relasi baru tersebut disebut CUSTOMER dan WAREHOUSE, dan dapat ditulis sebagai berikut: CUSTOMER (CUSTOMER NUMBER, CUSTOMER NAME, WAREHOUSE NUMBER) dan WAREHOUSE (WAREHOUSE NUMBER, WAREHOUSE LOCATION). Kunci utama untuk relasi CUSTOMER adalah CUSTOMER NUMBER, dan kunci utama untuk relasi WAREHOUSE adalah WAREHOUSE NUMBER. Selain kunci utama ini, kita dapat mengidentifikasi WAREHOUSE NUMBER menjadi kunci asing dalam relasi CUSTOMER. Kunci asing adalah atribut yang bukan kunci dalam suatu relasi tetapi merupakan kunci utama dalam relasi lain. Relasi SALES-REPORT asli yang tidak dinormalisasi telah diubah menjadi empat hubungan 3NF.

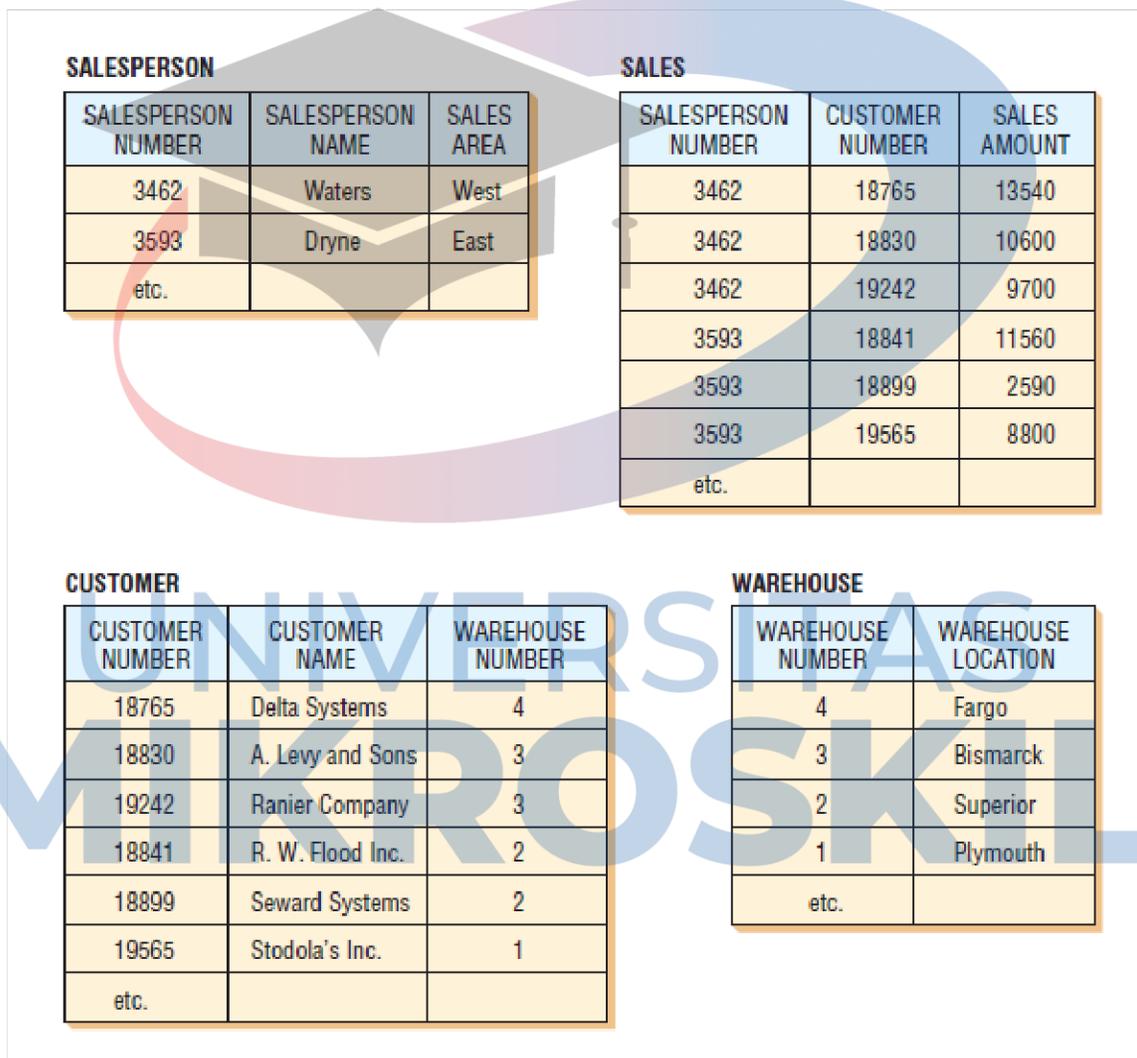


Gambar 2.11 Contoh 3NF

Dalam meninjau hubungan yang ditunjukkan pada gambar di bawah, kita dapat melihat bahwa single relasi SALES-REPORT diubah menjadi empat relasi berikut:

SALESPERSON (SALESPERSON NUMBER, SALESPERSON NAME, SALES AREA),
 SALES (SALESPERSON NUMBER, CUSTOMER NUMBER, SALES AMOUNT),
 CUSTOMER (CUSTOMER NUMBER, CUSTOMER NAME, WAREHOUSE
 NUMBER).

Bentuk normal ketiga (3NF) cukup untuk sebagian besar masalah desain database. Penyederhanaan yang didapat dari mengubah hubungan yang tidak dinormalisasi menjadi satu set hubungan 3NF cukup membantu saat menyisipkan, menghapus, dan memperbarui informasi dalam database.



Gambar 2.12 Diagram Model 3NF

Satu SALESPERSON melayani banyak CUTOMER, yang menghasilkan SALES dan menerima barang-barang mereka dari satu WAREHOUSE.

- a. Bentuk normal 3NF terpenuhi jika telah memenuhi bentuk 2NF, dan jika tidak ada atribut *non primary key* (biasa) yang memiliki ketergantungan terhadap atribut *non primary key* (biasa) yang lainnya
- b. Untuk setiap *Functional Dependency* dengan notasi $X \rightarrow A$, maka:
 - i. X harus menjadi *superkey* pada tabel tersebut
 - ii. Atau A merupakan bagian dari *primary key* pada tabel tersebut

Hal ini dapat dilihat pada gambar tabel-tabel dibawah ini, yakni tabel mahasiswa, tabel dosen, tabel matakuliah, dan tabel nilai.

Tabel Mahasiswa		
nim	nama	prodi
1234	Roma	TI
2345	Beni	SI

Tabel Dosen	
id dosen	nama dosen
SSD	Surya
RNW	Ronal
WHY	Wahyu
SAB	Sabrina

Tabel Matakuliah		
kode mtk	nama_mtk	id dosen
TI4801	Sistem Basis Data	SSD
TI4815	Rekayasa Perangkat Lunak	RNW
UN121	Kalkulus	WHY
UN125	Bahasa Indonesia	SAB

Tabel Nilai		
nim	kode mtk	nilai
1234	TI4801	A
1234	TI4815	C
2345	TI4801	B
2345	UN121	B
2345	UN125	A

Gambar 2.13 Tabel 3rd Normal Form / 3NF

Selanjutnya, langkah kelima dilakukan pengecekan *composite* dan *multivalue attribute* dengan cara melihat data yang mengandung tanda koma. Jika tidak ada data yang mengandung nilai koma, maka tabel yang dihasilkan tetap dan proses normalisasi selesai, dan tabel dapat diimplementasikan ke *database relational*.

2.5. Konsep Dasar Perancangan Sistem Informasi

Perancangan adalah satu langkah untuk memberikan gambaran secara umum kepada manusia atau pengguna tentang sistem yang diusulkan. Perancangan sistem atau desain secara umum mendefinisikan komponen-komponen sistem informasi pemetaan yang akan dirancang. Berdasarkan definisi di atas dapat disimpulkan bahwa perancangan merupakan suatu proses penerjemahan kebutuhan pemakai informasi yang

diperlukan oleh sistem yang ada serta untuk menunjang pengembangan sistem yang baru dengan tujuan untuk memecahkan masalah dan mengembangkan solusi terbaik bagi perusahaan. [21]

2.6. Teknik Pengembangan Sistem

Alar-alat yang dapat digunakan untuk mempermudah dalam pengembangan sistem informasi adalah:

2.6.1. *Fishbone Diagram*

Fishbone diagram / Ishikawa diagram / Cause-Effect Analysis adalah suatu tindakan dan langkah perbaikan yang lebih mudah dilakukan dengan mencari akar penyebab masalah. Manfaat *fishbone diagram* diantaranya mudah dibaca untuk diagram hubungan sebab akibat sehingga orang-orang lebih cenderung menggunakan metode ini, mengetahui penyebab masalah yang berpengaruh, produktivitas meningkat, dan meningkatkan komunikasi internal dan eksternal.

Langkah-langkah dalam mengerjakan *fishbone diagram* adalah [22]:

1. Tentukan Masalah

Masalah diinterpretasikan sebagai akibat. Setiap orang harus memahami dengan jelas sifat masalah dan proses atau produk yang dibahas.

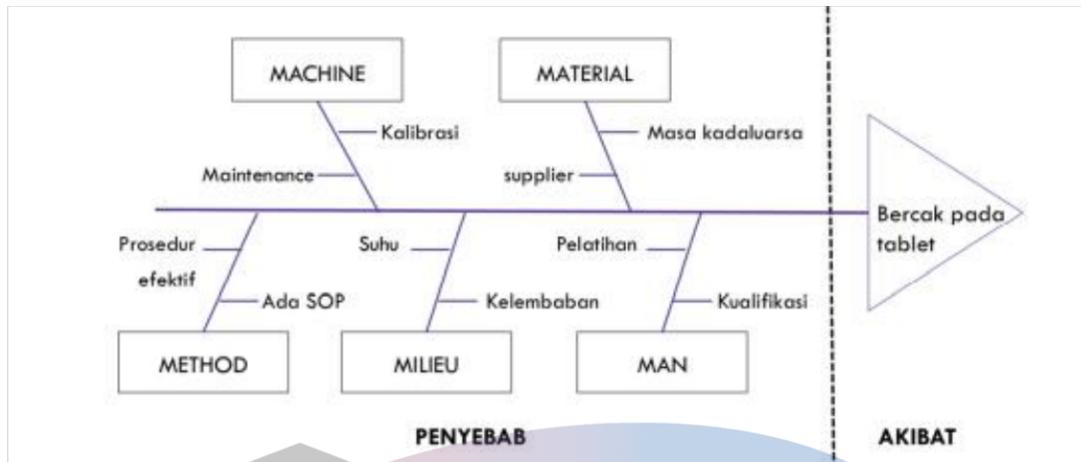
2. Tentukan kategori penyebab utama

Penyebab masalah dikelompokkan ke dalam kategori utama agar dalam menentukan akar penyebab masalah terstruktur.

3. Identifikasi terkait penyebab masalah dengan cara *brainstorming*

Setiap kategori utama memiliki sebab-sebab yang perlu diuraikan melalui *brainstorming*.

Contoh dalam menggunakan *fishbone diagram* sebagai berikut



Gambar 2.14 Contoh Penggunaan *Fishbone* Diagram di Industri Manufaktur

4. Analisis Diagram

Analisis membantu dalam mengidentifikasi penyebab yang memerlukan investigasi lebih lanjut. Jika terdapat banyak cabang dalam penyebab utama diperlukan investigasi lebih lanjut. Penyebab masalah yang muncul berulang kali berpotensi sebagai akar masalahnya.

2.6.2. PIECES

Analisis **PIECES** merupakan alat untuk menganalisis sistem informasi berbasis komputer, dimana terdiri dari poin-poin penting yang berguna untuk dijadikan pedoman dalam menganalisis sistem. Terdapat 6 kriteria pada analisis **PIECES** yaitu [23]:

1. *Performance* (Kinerja)

Kinerja merupakan variabel pertama dari **PIECES** dimana memiliki peranan penting untuk melihat sejauh mana dan seberapa handal suatu sistem informasi dalam mengelola data untuk menghasilkan informasi dan tujuan yang diharapkan.

2. *Information* (Informasi)

Informasi dan data yang dihasilkan sistem informasi ataupun dibutuhkan oleh perusahaan harus memiliki nilai yang berguna untuk pengambilan keputusan oleh manajemen perusahaan karena merupakan salah satu faktor penting untuk kemajuan perusahaan.

3. *Economics* (Ekonomi)

Ekonomi merupakan parameter untuk melihat apakah pengorbanan perusahaan untuk mengaplikasikan sistem informasi sepadan dengan hasil yang diperoleh perusahaan.

4. *Controls* (Kontrol)

Kontrol yang baik tentang hal-hal yang terkait pengendalian dan pengamanan sistem akan menjaga suatu sistem sehingga pihak dari luar sistem tidak akan mudah untuk masuk dan mengacaukan sistem tersebut.

5. *Efficiency* (Efisiensi)

Sistem informasi yang digunakan harus memiliki nilai keunggulan jika dibandingkan dengan penggunaan sistem secara manual. Keunggulan tersebut dapat terletak pada tingkat efisiensi saat sistem informasi tersebut beroperasi.

6. *Services* (Pelayanan)

Layanan merupakan aspek penting yang diberikan oleh sistem. Oleh sebab itu, kekurangan pada sistem yang perlu diperbaiki atau ditambahkan layanan baru harus disesuaikan dengan kebutuhan.

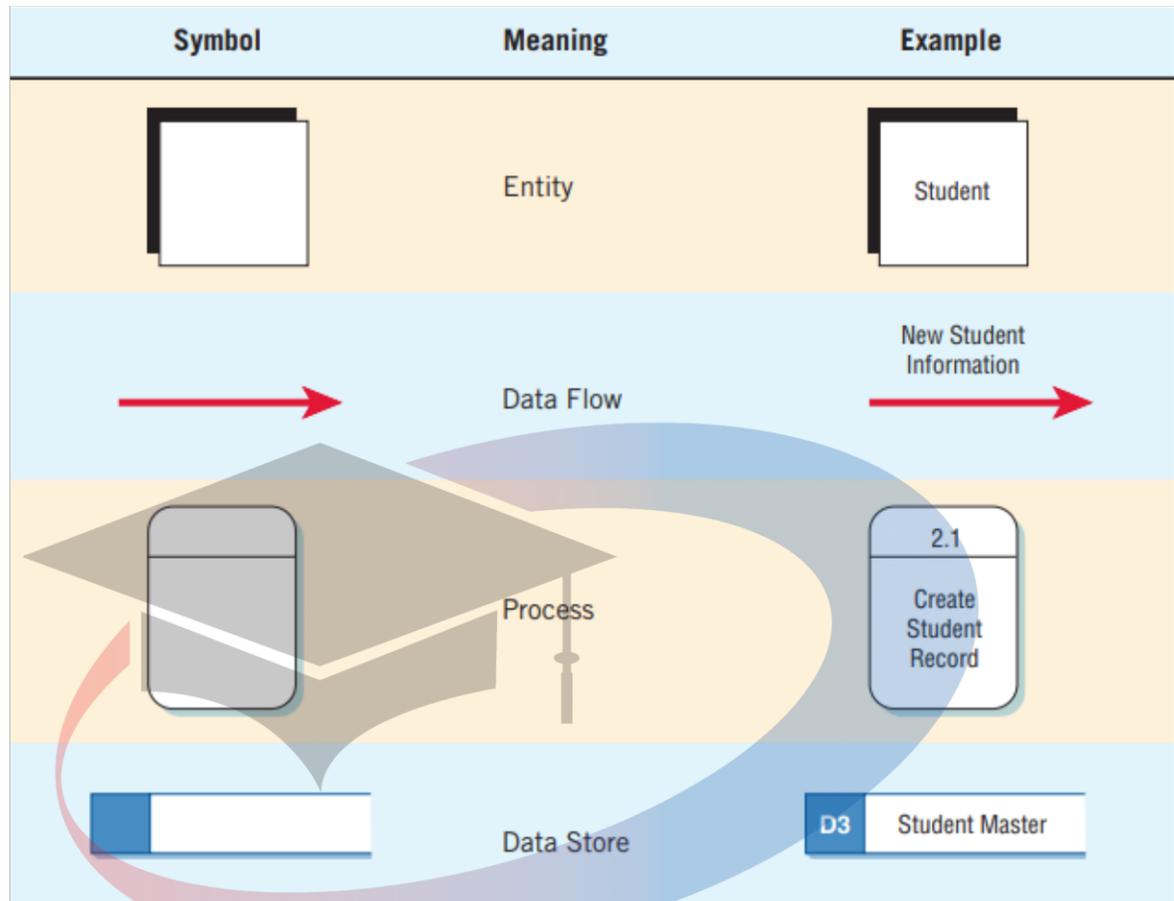
2.6.3. Diagram Aliran Data (*Data Flow Diagram*)

Data Flow Diagram (DFD) adalah diagram permodelan yang memberikan gambaran sistem sebagai hubungan proses yang dilihat berdasarkan fungsi dan terhubung oleh *data flow manual* atau dengan komputer. DFD terdiri dari dua: DFD fisik (*Physical DFD*) dan DFD logis (*Logical DFD*). DFD fisik yaitu DFD yang mengutamakan proses yang terjadi pada sistem yang diterapkan, DFD logis mengutamakan proses yang sudah terdapat pada sistem.

DFD memiliki tujuan yaitu [24]:

1. Mendeskripsikan fungsi yang mengubah aliran data
2. Menjelaskan sistem yang ada atau yang sedang dibuat secara logis, serta aliran data dari *input* ke level data *output*

Ada beberapa simbol DFD yang dipakai untuk menggambarkan data beserta proses transformasi data, antara lain: [17]



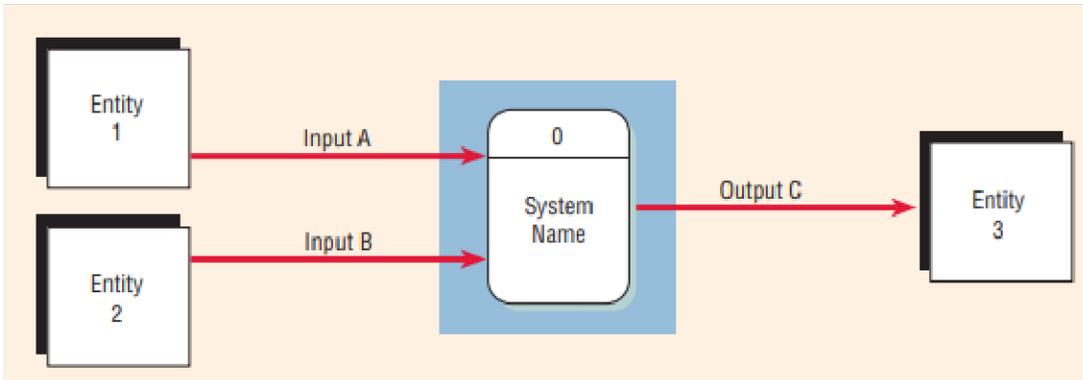
Gambar 2.15 Simbol-simbol Data Flow Diagram (DFD)

1. *Process* adalah simbol untuk proses pengolahan atau transformasi data
2. *Entity* adalah simbol untuk menggambarkan asal atau tujuan data
3. *Data Flow* adalah simbol untuk menggambarkan aliran data yang berjalan
4. *Data Store* adalah simbol untuk menunjukkan penyimpanan data yang memungkinkan pemeriksaan, penambahan, dan pengambilan data.

Dalam membuat DFD, terdapat beberapa langkah, yaitu [16]:

1. Membuat Diagram Konteks

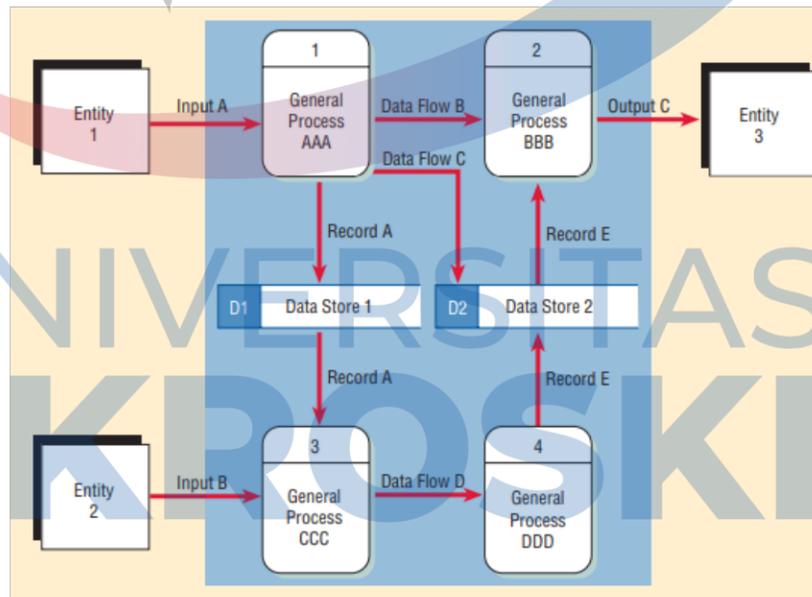
Diagram konteks adalah *level* tertinggi dalam DFD dan hanya berisi satu proses, mewakili keseluruhan sistem. Prosesnya diberi angka nol. Semua entitas eksternal ditampilkan pada diagram konteks, serta alur dari asal dan tujuan dari data tersebut. Contoh dari diagram konteks:



Gambar 2.16 Contoh Diagram Konteks

2. Menggambarkan Diagram Tingkat 0

Diagram tingkat 0 adalah diagram pecahan dari diagram konteks dan dapat mencakup hingga sembilan proses. Memasukkan lebih banyak proses pada level ini akan menghasilkan diagram yang berantakan yang sulit dipahami. Setiap proses diberi nomor dengan bilangan bulat, biasanya dimulai dari pojok kiri atas diagram dan diteruskan ke pojok kanan bawah. Contoh dari diagram tingkat 0:

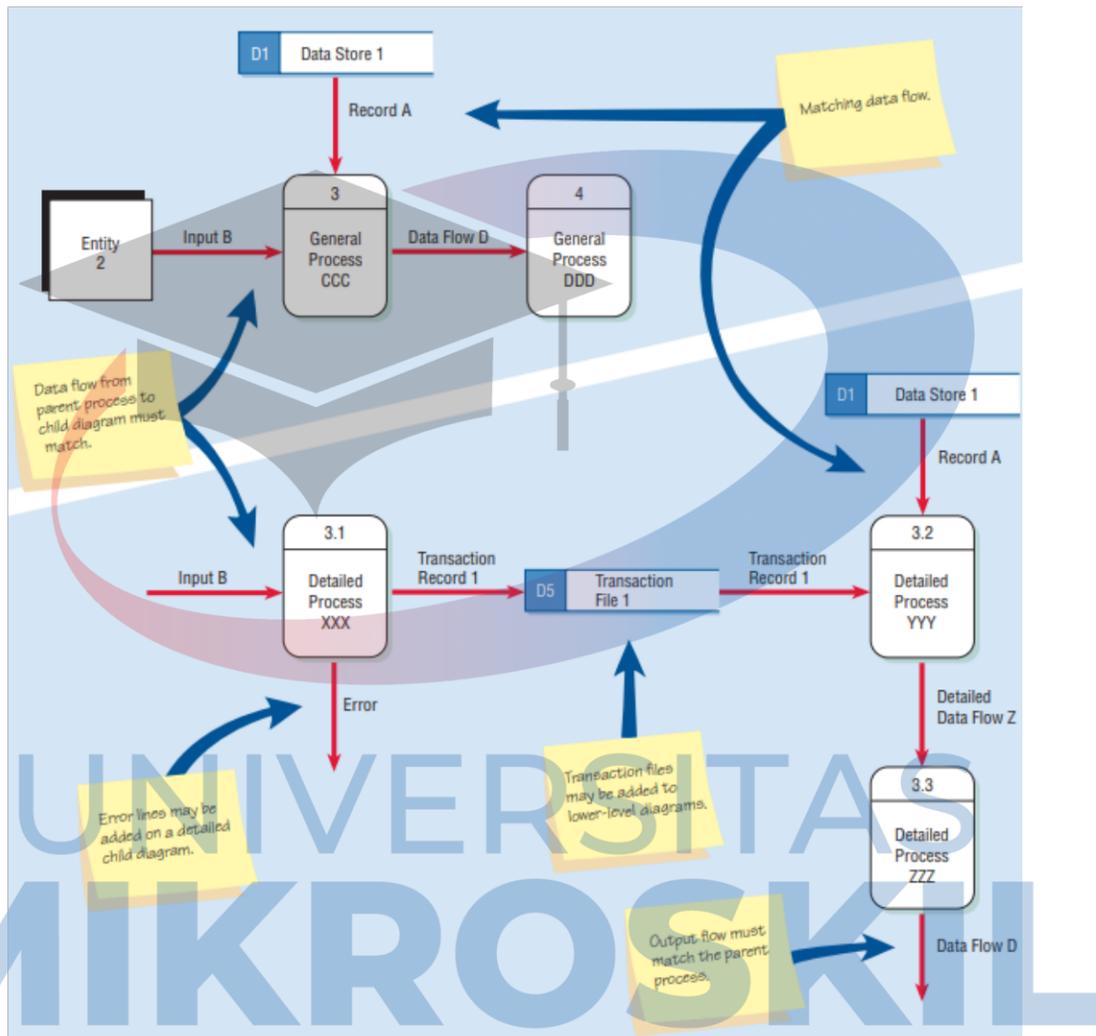


Gambar 2.17 Contoh Diagram Tingkat 0

3. Menciptakan Diagram Anak

Diagram tingkat 0 tersebut dapat dipecahkan lagi menjadi gambaran sistem yang lebih kecil yaitu diagram anak. Diagram anak diberi nomor yang sama dengan proses induknya pada diagram 0. Misalnya, proses 3 akan dipecahkan menjadi diagram 3. Proses pada diagram

anak diberi nomor menggunakan nomor proses induk, titik desimal, dan nomor unik untuk setiap proses anak. ada diagram 3, proses akan diberi nomor 3.1, 3.2, 3.3, dan seterusnya. Hal ini memungkinkan analisis untuk melacak serangkaian proses melalui berbagai tingkat pecahan. Jika Diagram 0 menggambarkan proses 1, 2, dan 3, diagram anak 1, 2, dan 3 semuanya berada pada level yang sama.



Gambar 2.18 Penjelasan Diagram Anak

2.6.4. Kamus Data

Kamus data adalah kumpulan dari referensi data tentang suatu metadata yang digunakan sebagai panduan oleh *system analyst* untuk menganalisis dan merancang. Sebuah kamus data adalah dokumen yang mengumpulkan dan mengoordinasikan istilah data tertentu, dan menegaskan apa istilah-istilah penting bagi orang yang berbeda dalam organisasi. Salah satu

alasan penting untuk memelihara kamus data adalah untuk menjaga kebersihan data atau dengan kata lain data harus konsisten. Kamus data menggantikan program yang sering berubah, dan itu mencegah program tidak berjalan karena perubahan belum diperbarui. Kamus data otomatis penting untuk perusahaan besar yang memiliki sistem yang menghasilkan beberapa ribu elemen data yang memerlukan katalogisasi dan referensi silang. [17]

Kamus data harus memuat informasi tentang hal-hal berikut yaitu *Data Flow*, *Data Structure*, *Data Elements*, *Data Stores*, dan *Data Processing*. Aliran data yang digambarkan dengan menggunakan DFD (*Data Flow Diagram*) dan diwakili dalam bentuk aljabar seperti simbol berikut ini [25]

Simbol	Keterangan
=	sama dengan atau terdiri dari atau terbentuk dari
+	dan
[]	pilih salah satu
{ }	iterasi atau pengulangan
()	pilihan (option)
*	komentar
	pemisah

Gambar 2.19 Notasi dalam Kamus Data

2.6.5. Teknik Spesifikasi Proses

Digunakan untuk mendeskripsikan spesifikasi dari setiap proses yang paling rendah (proses *atomic*) yang ada pada sistem dengan menggunakan notasi yang disebut *structured english* atau *pseudo-code*. Penulisannya cukup sederhana sehingga dapat digunakan sebagai media untuk mengkomunikasikan proses yang dilakukan sistem kepada pemakai. [25]