

BAB II

TINJAUAN PUSTAKA

2.1 Analisis dan Perancangan

2.1.1 Pengertian Analisis

Analisis sistem adalah penelitian atas sistem yang telah ada dengan tujuan untuk merancang sistem baru atau diperbaharui [1].

Analisis sistem adalah memeriksa sebuah masalah yang ada yang akan diselesaikan oleh perusahaan dengan menggunakan sistem informasi. Analisis sistem mencakup beberapa langkah yang harus dilakukan, yaitu:

1. Menentukan masalah.
2. Mengidentifikasi penyebab dari masalah tersebut.
3. Menentukan pemecahan masalahnya.
4. Mengidentifikasi kebutuhan informasi yang dibutuhkan untuk memecahkan masalah tersebut [2].

Dari teori-teori di atas dapat dikatakan bahwa analisis sistem adalah memeriksa, meneliti sistem yang sudah ada, dan mencari di mana letak kesalahan sistem itu, untuk kemudian diperbaiki dan diperbaharui atau digunakan untuk merancang sistem yang baru.

2.1.2 Pengertian Perancangan

Perancangan sistem adalah cara bagaimana sebuah sistem dapat memenuhi kebutuhan informasi yang telah ditentukan oleh analisa sistem [2].

Perancangan sistem adalah penentuan proses dan data yang diperlukan oleh sistem baru, jika sistem itu berbasis komputer, perancangan dapat menyertakan spesifikasi peralatan yang akan digunakan [1].

Dari teori-teori di atas dapat dikatakan bahwa perancangan sistem adalah penentuan bagaimana sebuah sistem yang akan dibuat sesuai dengan yang diinginkan oleh analisis sistem baik penentuan proses dan maupun datanya.

2.2 Konsep Sistem Informasi

2.2.1 Sistem

Sistem berasal dari bahasa Yunani yaitu "sistema" yang artinya kesatuan. Sebuah sistem merupakan suatu elemen atau perangkat kelembagaan yang dapat membantu dalam pencapaian tujuan ataupun sasaran. Dan itulah analisis sistem diperlukan sebagaimana yang telah diungkapkan sebelumnya. Untuk lebih jelasnya berikut para ahli mengungkapkan definisi sistem.

Sistem pada dasarnya adalah sekelompok unsur yang erat berhubungan satu dengan lainnya, yang berfungsi bersama-sama untuk mencapai tujuan tertentu [3].

Sistem adalah suatu kumpulan atau himpunan dari unsur, komponen atau variabel-variabel yang terorganisasi, saling berinteraksi, saling tergantung satu sama lain dan terpadu [4].

Sistem merupakan suatu kelompok dua atau lebih komponen yang saling berkaitan (*interrelated*) atau sub sistem yang bersatu untuk mencapai tujuan yang sama [5].

Sistem adalah suatu sistem di dalam organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan yang diperlukan [6].

Suatu sistem dibentuk oleh unsur-unsur yang disebut dengan parameter sistem sebagai berikut:

1. *Input*.

Input dalam bahasa Inggris diartikan sebagai "tenaga yang dimasukkan". Hal ini berarti bahwa *input* merupakan komponen penggerak atau pemberi tenaga dimana sistem itu dioperasikan. *Input* bisa berupa *input* perawatan (*maintenance input*) maupun *input* sinyal (*signal input*). *Maintenance input* merupakan energi yang dimasukkan supaya sistem dapat beroperasi, sedangkan *signal input* adalah energi yang diproses untuk didapatkan keluaran. Dalam suatu sistem pengambilan keputusan misalnya, *maintenance input*-nya adalah tim manajemen yang merupakan *personil* pengambilan keputusan sedangkan *signal input*-nya adalah informasi atau data-data yang menunjang kemudahan pengambilan keputusan.

2. Proses.

Proses merupakan langkah pengolahan *input* untuk mendapatkan hasil atau tujuan yang diinginkan. Disinilah dibutuhkan integrasi yang kuat antar subsistem atau dengan kata lain diperlukan keterpaduan antar subsistem agar proses interaksi untuk mencapai tujuan dapat berjalan lancar.

3. Output.

Output adalah hasil operasi dari suatu data yang telah diproses dalam suatu sistem, baik yang menjadi bentuk data maupun berbentuk informasi yang telah diolah yang berguna bagi pengguna sistem.

4. Batasan (Tujuan dan Kendala).

Setiap permasalahan yang akan dianalisis dalam bentuk sisten diperlukan batasan-batasan (*restriksi*). Batasan-batasan tersebut menyebabkan perlunya pembuatan pedoman atau petunjuk yang mengikat permasalahan. Batasan terdiri dari dua unsur, yaitu tujuan dan kendala. Tujuan adalah sasaran yang ingin dicapai, sedangkan kendala merupakan aturan atau batas-batas yang berlaku atas tujuan tersebut yang akan membuat tujuan menjadi lebih bermanfaat. Batasan tersebut mampu mengidentifikasi apa yang harus diantisipasi dalam tujuan. Dengan adanya batasan tersebut diatas, maka keberhasilan tujuan pengembalian keputusan akan menjadi lebih mudah untuk dilakukan.

2.2.2 Informasi

Informasi adalah data yang telah diklasifikasikan atau diolah atau diinterpretasikan untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya. Nilai informasi berhubungan dengan keputusan. Bila tidak ada pilihan atau keputusan maka informasi tidak diperlukan. Keputusan dapat berkisar dari keputusan berulang sederhana sampai keputusan strategis jangka panjang. Nilai informasi dilukiskan paling berarti dalam konteks pengambilan keputusan [4].

Berikut ini beberapa pengertian menurut para ahli:

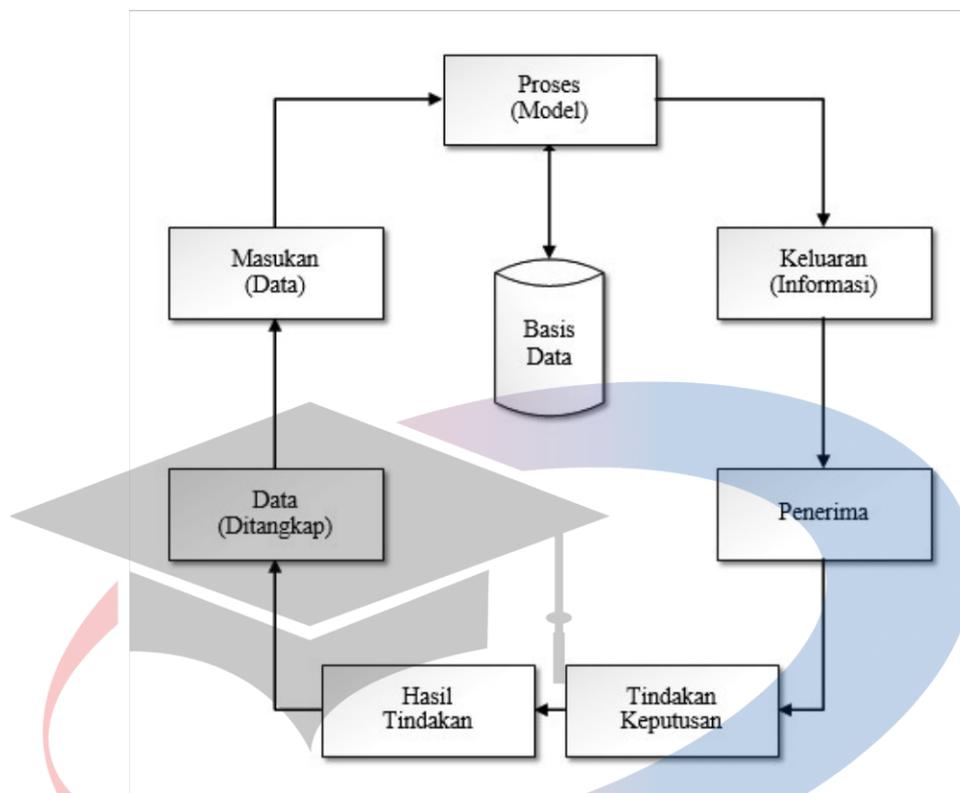
Menurut Davis dalam Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat bagi pengambilan keputusan saat ini atau saat mendatang [7].

Informasi merupakan kumpulan data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerima [8].

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya [6].

Fungsi utama informasi adalah menambah pengetahuan atau mengurangi ketidakpastian pemakai informasi. Informasi yang disampaikan kepada pemakai mungkin merupakan hasil dari data yang dimasukkan ke dalam pengolahan. Akan tetapi dalam kebanyakan pengambilan keputusan yang kompleks, informasi hanya dapat menambah kemungkinan kepastian atau mengurangi bermacam-macam pilihan [4].

Data merupakan bentuk mentah yang belum dapat bercerita banyak sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model agar menghasilkan informasi. Data diolah melalui suatu model informasi. Pengguna akan menerima informasi tersebut untuk membuat suatu keputusan dan melakukan tindakan yang akan mengakibatkan munculnya sejumlah data lagi. Data tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model dan seterusnya sehingga membentuk suatu siklus. Siklus inilah yang disebut sebagai Siklus Infomasi, perhatikan **Gambar 2.1:**



Gambar 2.1 Siklus Informasi

Sumber: Abdul Kadir, 2003

2.2.3 Sistem Informasi

Sistem Informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan [4].

Sistem Informasi dalam manajemen perusahaan adalah sistem yang terbuka dan sistem lingkaran tertutup. Sistem informasi mendapatkan *input* berupa data-data atau kejadian dalam perusahaan, diubah dengan pengolah informasi untuk memperoleh informasi. Pengolah informasi tersebut dapat berupa komputer, orang ataupun gabungan keduanya. Adapun yang menjalankan fungsi mekanisme pengendaliannya adalah para manajer dalam mengambil keputusan untuk memecahkan persoalan dalam perusahaan dan untuk mencapai target dan tujuan perusahaan. Selanjutnya keputusan-keputusan manajer diharapkan dapat membawa

perubahan sehingga pada akhirnya *output* sistem memenuhi harapan manajer. Jadi, suatu sistem informasi terdiri atas elemen data, informasi, pengolahan informasi dan manajer [9].

Suatu sistem informasi adalah suatu tipe khusus dari sistem kerja yang fungsi internalnya terbatas pada pemrosesan informasi dengan melakukan enam tipe operasi, menangkap (*capturing*), mentransmisikan (*transmitting*), menyimpan (*storing*), mengambil (*retrieving*), memanipulasi (*manipulating*), dan menampilkan (*displaying*) informasi. Suatu informasi ada karena digunakan untuk memproduksi informasi dan atau mendukung atau mengotomatiskan kerja yang dilakukan oleh sistem-sistem kerja. Definisi dan penjelasan sistem informasi yang merupakan satu dimensi di matrik pengukuran kesuksesan sistem oleh Seddon et al.

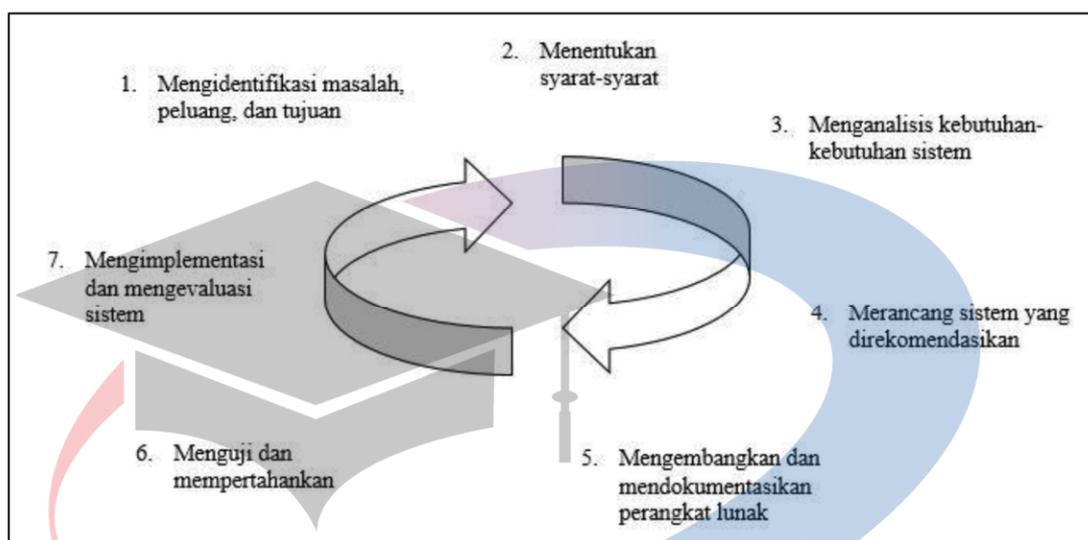
2.3 Siklus Hidup Pengembangan Sistem

Siklus Hidup Pengembangan Sistem (*SHPS*) atau *Sistem Development Life Cycle (SDLC)* adalah pendekatan melalui beberapa tahap untuk menganalisis dan merancang sistem yang dimana sistem tersebut telah dikembangkan dengan sangat baik melalui penggunaan siklus kegiatan menganalisis dan pemakaian secara spesifik.

Penganalisis tidak sepakat dengan beberapa banyaknya tahap yang ada didalam siklus hidup pengembangan sistem, namun mereka umumnya memuji pendekatan terorganisir mereka. Siklus ini dikembangkan dalam tujuh tahap seperti yang ditampilkan dalam gambar dibawah ini. Meskipun tahap-tahap ditampilkan secara terpisah, namun tidak pernah tercapai sebagai satu langkah terpisah. Melainkan beberapa aktifitas-aktifitas tersebut dilakukan secara berulang-ulang. Lebih berguna lagi memikirkan bahwa *SHPS* biasa dicapai dalam tahap-tahap (dengan aktifitas berulang yang saling tumpang tindih satu sama lainnya dan meninjau ke tujuan terakhir) dan tidak dalam langkah-langkah yang terpisah [11].

Siklus hidup pengembangan sistem adalah pendekatan melalui beberapa tahap untuk menganalisis dari merancang sistem, dimana sistem tersebut telah dikembangkan dengan sangat baik melalui penggunaan siklus kegiatan penganalisis dan pemakai secara spesifik [11].

Tahap-tahap didalam siklus hidup pengembangan sistem merupakan aktifitas yang dilakukan secara simultan dan berulang-ulang untuk menuju ke tujuan akhir seperti pada gambar berikut:



Gambar 2.2 Siklus Hidup Pengembangan Sistem

Sumber: Kenneth E. Kendall & Julie E. Kendall (1), 2003

Dari gambar diatas, siklus hidup pengembangan sistem dapat diuraikan dan dijelaskansebagai berikut [11]:

1. Mengidentifikasi masalah, peluang dan tujuan.

Penganalisis mengidentifikasi masalah, peluang dan tujuan-tujuan yang hendak dicapai. Tahap ini sangat penting bagi keberhasilan proyek, karena tidak seorangpun yang ingin membuang-buang waktu kalau tujuan masalah yang keliru. Penganalisis harus menemukan apa yang sedang dilakukan dalam bisnis lalu melihat beberapa aspek dalam aplikasi-aplikasi sistem informasi untuk membantu bisnis supaya mencapai tujuan-tujuannya dengan menyebut masalah atau peluang-peluang tertentu. Aktivitas dalam tahap ini meliputi wawancara terhadap manajemen pemakai, menyimpulkan pengetahuan yang diperoleh, mengestimasi cakupan proyek, dan mendokumentasikan hasil-hasilnya.

2. Menentukan syarat-syarat informasi.

Dalam tahap berikutnya, penganalisis memasukkan apa saja yang menentukan syarat-syarat informasi untuk para pemakai yang terlibat. Di antara perangkat-perangkat yang dipergunakan untuk menetapkan syarat-syarat informasi di dalam bisnis diantaranya adalah menentukan *sampel* dan memeriksa data mentah, wawancara, mengamati perilaku pembuat keputusan dan lingkungan kantor, dan *prototyping*.

Dalam tahap syarat-syarat informasi *SHPS*, penganalisis berusaha untuk memahami informasi apa yang dibutuhkan pemakai agar bisa ditampilkan dalam pekerjaan mereka. Anda dapat melihat bahwa beberapa metode untuk menentukan syarat-syarat informasi ini melibatkan interaksi secara langsung dengan pemakai. Tahap ini membentuk gambaran mengenai organisasi dan tujuan-tujuan yang dimiliki seorang penganalisa.

Orang-orang yang terlibat dalam tahap ini adalah penganalisis dan pemakai, biasanya manajer operasional dan pegawai operasional. Penganalisis sistem perlu tahu detail-detail fungsi-fungsi sistem yang ada, siapa (orang-orang yang terlibat), apa (kegiatan bisnis), dimana (lingkungan dimana pekerjaan itu dilakukan), kapan (waktu yang tepat), dan bagaimana (bagaimana prosedur yang harus dijalankan) dari bisnis yang sedang dipelajari. Kemudian penganalisis juga harus bertanya mengapa bisnis menggunakan sistem yang ada. Ada alasan yang bagus melakukan bisnis dengan menggunakan metode-metode yang ada, dan hal-hal seperti ini harus dipertimbangkan saat merancang sebuah sistem baru.

3. Menganalisis kebutuhan.

Perangkat dan teknik tertentu akan membantu analisis menentukan kebutuhan yaitu dengan menggunakan diagram aliran data untuk menyusun daftar *input*, proses, dan *output* fungsi bisnis dalam bentuk grafik terstruktur. Penganalisis sistem juga menganalisis keputusan terstruktur yang dibuat.

4. Merancang sistem yang direkomendasikan.

Dalam tahap desain dari siklus hidup pengembangan sistem, penganalisis sistem menggunakan informasi yang terkumpul sebelumnya untuk mencapai desain sistem informasi yang logik. Penganalisis merancang prosedur *data entry* sedemikian rupa sehingga data yang dimasukkan ke dalam sistem informasi benar-

benar akurat. Selain itu, penganalisis menggunakan teknik-teknik bentuk dan perancangan layar tertentu untuk menjamin keefektifan *input* sistem informasi.

Bagian dari perancangan sistem informasi yang logik adalah peralatan antarmuka pengguna. Antarmuka menghubungkan pemakai dengan sistem, jadi perannya benar-benar sangat penting. Tahap perancangan juga mencakup perancangan *file-file* atau basis data yang bisa menyimpan data-data yang diperlukan oleh pembuat keputusan. Basis data yang tersusun dengan baik adalah dasar bagi seluruh sistem informasi. Dalam tahap ini, penganalisis juga bekerja sama dengan pemakai untuk merancang *output* (baik pada layar maupun hasil cetakan).

Terakhir penganalisis harus merancang prosedur-prosedur *backup* dan kontrol untuk melindungi sistem dan data serta untuk membuat paket-paket spesifikasi program bagi pemrogram. Setiap paket bisa terdiri dari *layout input* dan *output*, spesifikasi *file*, dan detail-detail proses, serta pohon keputusan atau tabel, diagram aliran data, *flowchart* sistem, serta nama-nama dan fungsi-fungsi subprogram yang sudah tertulis.

5. Mengembangkan dan mendokumentasikan perangkat lunak.

Dalam tahap kelima dari siklus hidup pengembangan sistem, penganalisis bekerja bersama-sama dengan pemrogram untuk mengembangkan suatu perangkat lunak awal yang diperlukan. Beberapa teknik terstruktur untuk merancang dan mendokumentasikan perangkat lunak meliputi rencana terstruktur. Penganalisis sistem menggunakan salah satu semua perangkat ini untuk memprogram apa yang perlu diprogram.

Selama tahap ini, penganalisis juga bekerja sama dengan pemakai untuk mengembangkan dokumentasi perangkat lunak yang efektif, mencakup melakukan prosedur secara manual, bantuan *online*, dan *website*. Kegiatan dokumentasi menunjukkan kepada pemakai tentang cara penggunaan perangkat lunak dan apa yang harus dilakukan bila perangkat lunak mengalami masalah.

Pemrogram adalah pelaku utama dalam tahap ini karena mereka merancang, membuat kode dan mengatasi kesalahan-kesalahan dari program komputer. Bila programnya adalah untuk dijalankan dalam lingkungan *mainframe*, maka perlu diciptakan suatu *job control language (JCL)*. Untuk memastikan kualitasnya,

pemrogram bisa membuat perancangan dan kode program yang akan dijalankan, menjelaskan bagian-bagian kompleks dari program kepada tim pemrogram lainnya.

6. Menguji dan mempertahankan sistem.

Sebelum sistem informasi dapat digunakan, maka harus dilakukan pengujian terlebih dulu. Akan bisa menghemat biaya bila dapat menangkap adanya masalah sebelum sistem tersebut ditetapkan. Sebagian pengujian dilakukan oleh pemrogram sendiri, dan lainnya dilakukan oleh penganalisis sistem. Rangkaian pengujian ini pertama-tama dijalankan bersama-sama dengan data contoh serta dengan data aktual dari sistem yang telah ada.

Mempertahankan sistem dan dokumentasinya dimulai di tahap ini dan dilakukan secara rutin selama sistem informasi dijalankan. Sebagian besar kerja rutin pemrogram adalah melakukan pemeliharaan, dan bisnis menghabiskan banyak uang untuk kegiatan pemeliharaan. Sebagian besar prosedur sistematis yang dijalankan penganalisis selama siklus hidup pengembangan sistem membantu memastikan bahwa pemeliharaan bisa dijaga sampai tingkat minimum.

7. Mengimplementasikan dan mengevaluasi sistem.

Di tahap terakhir ini penganalisis membantu untuk mengimplementasikan sistem informasi. Tahap ini melibatkan pelatihan bagi pemakai untuk mengendalikan sistem. Sebagian pelatihan tersebut dilakukan oleh vendor, namun kesalahan pelatihan merupakan tanggung jawab penganalisis sistem. Selain itu, penganalisis perlu merencanakan konversi perlahan dari sistem lama ke sistem baru. Evaluasi yang ditunjukkan sebagai bagian dari tahap terakhir ini biasanya dimaksudkan untuk pembahasan. Sebenarnya, evaluasi dilakukan di setiap tahap. Kriteria utama yang harus dipenuhi ialah apakah pemakai yang dituju benar-benar menggunakan sistem.

2.4 Internet

2.4.1 Sejarah Internet

Sebelum *Internet* ada, *ARPAnet*(*US Defense Advanced Research Projects Agency*) atau Departemen Pertahanan Amerika pada tahun 1969 membuat jaringan komputer yang tersebar untuk menghindarkan terjadinya informasi terpusat, yang apabila terjadi perang dapat mudah dihancurkan. Jadi bila satu bagian dari sambungan *network* terganggu dari serangan musuh, jalur yang melalui sambungan itu secara otomatis dipindahkan ke sambungan lainnya. Setelah itu *Internet* digunakan oleh kalangan akademis (*UCLA*) untuk keperluan penelitian dan pengembangan teknologi. Dan baru setelah itu Pemerintah Amerika Serikat memberikan ijin ke arah komersial pada awal tahun 1990.

2.4.2 *E-mail (Electronic Mail)*

Electronic Mail adalah salah satu fasilitas atau aplikasi yang paling banyak digunakan di *Internet*. Hal ini karena *E-mail* merupakan alat komunikasi yang paling murah dan cepat. Dengan *E-mail* kita dapat berhubungan dengan siapa saja yang terhubung ke *Internet* di seluruh dunia dengan biaya pulsa lokal (bila kita menggunakan *line* telepon atau *ISDN*). *E-mail* menggeser penggunaan telepon dan *fax* dimasa kini.

Konsep *E-mail* adalah seperti kita mengirim surat dengan pos biasa, di mana kita mengirimkan ke kantor pos dengan dibubuhi alamat yang kita tuju. Dari Kantor Pos tersebut akan disampaikan ke Kantor Pos yang terdekat dengan alamat yang dituju dan akhirnya sampai ke alamat tersebut. Dan si penerima hanya membuka kotaknya saja yang ada di depan rumah. Di sini si pengirim tidak tahu apakah si orang yang dituju tersebut sudah menerima surat tersebut, sampai surat itu dibalas.

Dengan *E-mail* data dikirim secara elektronik sehingga sampai di tujuan dengan sangat cepat. Juga kita dapat mengirim *file-file* berupa program, gambar, grafik dan lain sebagainya. Kita juga dapat mengirim ke lebih dari satu orang sekaligus pada saat bersamaan.

2.4.3 *WWW (World Wide Web)*

WWW adalah aplikasi yang paling menarik di *Internet* dan seperti *E-mail* aplikasi ini sangat penting dan banyak digunakan. Aplikasi ini kadang disebut "*The killer application*" atau "*the world is in your fingertip*" karena sedemikian mudahnya kita dapat mendapatkan informasi tidak hanya teks tetapi gambar (*images*), maupun *multimedia*.

Dalam aplikasi ini banyak fasilitas yang dapat dilakukan seperti:

1. Memesan atau membeli suatu barang secara *online*.
2. Mendaftar secara *online*.
3. Mengakses *multimedia*, dan sebagainya.

Informasi yang diletakkan di *WWW* disebut "*HomePage*" dan setiap *homepage* mempunyai alamatnya masing-masing. Untuk dapat menarik perhatian *user* sehingga *homepage* dapat sering dikunjungi *user*, maka kita harus *men-design* semenarik mungkin dan banyak terdapat informasi yang jelas. Dalam hal ini bidang seni sangat dibutuhkan, sehingga dunia periklanan dan dunia bisnis semakin semarak.

2.4.4 FTP (File Transfer Protocol)

File Transfer Protocol (FTP) adalah salah satu aplikasi di *Internet* untuk mengambil (*download*) dan meletakkan (*upload*) suatu *file* di *FTP server*. Dengan hal ini kita dapat bertukar *file* dengan cepat. Pada saat ini banyak program atau *software* yang bebas untuk di *download* dari manapun di *Internet*.

2.4.5 TCP / IP (Transfer Control Protocol / Internet Protocol)

TCP/IP merupakan protokol jaringan komputer terbuka dan bisa terhubung dengan berbagai jenis perangkat keras dan lunak. *TCP* terdiri beberapa *layer* atau lapisan yang memiliki fungsi tertentu dalam komunikasi data. Setiap fungsi dari *layer* selain dapat bekerjasama dengan *layer* pada tingkat lebih rendah atau lebih tinggi, juga bisa berkomunikasi dengan *layer* sejenis pada *remote host (peering)*. *IP* adalah jantung *TCP/IP* memiliki peran sebagai pembawa data yang independen. Semua dokumen *TCP/IP* dalam bentuk *public* dokumen *IEN (Internet Engineering Notes)* dan *RFC (Request for Comments)*. *IP* dibagi atas kelas *network A, B, dan C*. Sedangkan kelas *D* untuk keperluan membalikkan (*reverse*) *IP* yang boleh diabaikan. *IP* ditulis dalam

bilangan desimal dari 0 sampai 255. Data yang mengalir antar *layer* atau antar *host* dienkapsulasi dan diberi judul agar tiap *layer* bisa memprosesnya. Sebuah *host* tidak tahu alamat *IP gateway* di jaringan lain, tetapi data mengalir ke *host* tujuan di jaringan lain melalui *gateway* jaringannya setelah diberi penentuan *routing* alamat *IP*. *TCP* singkatan dari *transfer control protocol* dan *IP* singkatan dari *Internet Protocol*. *TCP/IP* menjadi satu nama karena fungsinya selalu bergandengan satu sama lain dalam komunikasi data. *TCP/IP* saat ini dipergunakan dalam banyak jaringan komputer lokal (*LAN*) yang terhubung ke *Internet*.

2.5 E-Commerce

2.5.1 Definisi E-Commerce

E-Commerce merupakan satu set dinamis teknologi, aplikasi dan proses bisnis yang menghubungkan perusahaan, konsumen dan komunitas tertentu melalui transaksi elektronik dan perdagangan barang, pelayanan dan informasi yang dilakukan secara elektronik [10].

2.5.2 Jenis E-Commerce

Secara umum *E-Commerce* bisa diklarifikasikan menjadi dua jenis, yaitu [10]:

1. *Business to Business (B2B)*, dengan karakteristiknya:
 - *Trading partners* yang sudah saling mengetahui dan antara mereka sudah terjalin hubungan yang berlangsung cukup lama. Pertukaran informasi hanya berlangsung di antara mereka dan karena sudah saling mengenal, maka pertukaran informasi tersebut dilakukan atas dasar kebutuhan dan kepercayaan.
 - Pertukaran data dilakukan secara berulang-ulang dan berkala dengan format data yang telah disepakati. Jadi layanan yang digunakan antar kedua sistem tersebut sama dan menggunakan standar yang sama pula.
 - Salah satu pelaku tidak harus menunggu *partner* mereka lainnya untuk mengirimkan data.
 - Model yang umum digunakan adalah *peer-to-peer*, di mana *processing intelligence* dapat didistribusikan dikedua pelaku bisnis.
2. *Business to Consumer (B2C)*, dengan karakteristiknya:
 - Terbuka untuk umum, di mana informasi disebarakan secara umum pula.

- Layanan yang dilakukan juga bersifat umum, sehingga mekanismenya dapat digunakan oleh orang banyak. Sebagai contoh, karena sistem *web* sudah umum di kalangan masyarakat, maka sistem yang digunakan adalah sistem *web* pula.
- Layanan yang diberikan berdasarkan permintaan. Konsumen berinisiatif sedangkan produsen harus siap memberikan respon terhadap inisiatif konsumen tersebut.
- Sering digunakan sistem pendekatan *client-server*, di mana konsumen dari pihak *client* menggunakan sistem yang minimal (berbasis *web*) dan penyedia barang atau jasa (*business procedure*) berada pada pihak *server*.

2.5.3 Keuntungan *E-Commerce*

Dengan adanya *E-Commerce* kita dapat mengambil keuntungan. Keuntungan-keuntungan yang dapat diambil dengan adanya *E-Commerce* sebagai berikut [10]:

1. *Revenue stream* baru yang mungkin lebih menjanjikan, yang tidak bisa ditemui di sistem transaksi tradisional.
2. Dapat meningkatkan *market exposure*.
3. Menurunkan biaya (*operating cost*).
4. Melebarkan jangkauan (*global reach*).
5. Meningkatkan kesetiaan konsumen (*consumer loyalty*).
6. Meningkatkan *supplier management* dan *Value Chain*.
7. Mempersingkat waktu produksi (*production time*).

2.5.4 Kekurangan *E-Commerce*

Dari segi pandangan bisnis apabila terjadi penyalahgunaan dan kegagalan dalam sistem maka akibat yang dapat terjadi adalah sebagai berikut [10]:

1. Kehilangan segi keuangan (*financial*) secara langsung karena kecurangan.
Penipuan berasal dari dalam/luar dengan mentransfer sejumlah uang dari rekening satu ke rekening lainnya.
2. Pencurian informasi rahasia yang berharga.
Gangguan yang timbul dapat menyingkap semua informasi rahasia tersebut kepada pihak-pihak yang tidak berhak dan dapat mengakibatkan kerugian yang besar bagi korban.

3. Kehilangan kesempatan bisnis karena gangguan keamanan.

Bergantung pada pelayanan elektronik dapat mengakibatkan gangguan selama periode waktu yang tidak dapat diperkirakan.

4. Penggunaan akses ke sumber oleh pihak-pihak yang tidak berwenang.

Pihak luar mendapatkan akses yang sebenarnya bukan menjadi haknya dan dia menggunakan hal tersebut untuk kepentingan pribadi.

5. Kehilangan kepercayaan dari para konsumen.

Kepercayaan konsumen terhadap sebuah perusahaan atau lembaga tertentu dapat hilang karena berbagai macam faktor, seperti usaha yang dilakukan dengan sengaja, ketidak-jujuran praktek bisnis yang tidak benar dan bisa juga kesalahan-kesalahan fatal yang dilakukan oleh perusahaan yang terkait, yang dapat mengakibatkan kepercayaan konsumen berkurang.

6. Kerugian-kerugian yang tak terduga.

Gangguan terhadap transaksi bisnis yang disebabkan oleh gangguan dari luar yang dilakukan dengan sengaja ketidak-jujuran, praktek bisnis yang tidak benar, kesalahan dari faktor manusia, atau kesalahan sistem elektronik dapat mengakibatkan kerugian transaksi bisnis tidak dapat dihindarkan, terutama dari segi *financial* (keuangan).

2.6 Diagram Aliran Data (*Data Flow Diagram*)

Data flow diagram (DFD) adalah suatu *network* yang menggambarkan suatu sistem komputerisasi, manualisasi atau gabungan dari keduanya, yang penggambarannya disusun dalam bentuk kumpulan komponen sistem yang saling berhubungan sesuai dengan aturan mainnya. Keuntungan dari *DFD* adalah memungkinkan untuk menggambarkan sistem dari *level* yang paling tinggi kemudian menguraikannya menjadi *level* yang lebih rendah (dekomposisi), sedangkan kekurangan dari *DFD* adalah tidak menunjukkan proses pengulangan (*looping*), proses keputusan dan proses perhitungan [4].

1. Diagram Konteks.

Diagram konteks adalah tingkatan tertinggi dalam *DFD* dan hanya memuat satu proses menunjukkan sistem secara keseluruhan. Proses tersebut diberi nomor

nol. Semua entitas eksternal yang ditunjukkan pada diagram konteks berikut aliran data utama menuju dan dari sistem.

2. Data Flow Diagram.

Data Flow Diagram (DFD) merupakan diagram yang sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir (misalnya lewat telepon, surat dan sebagainya) atau lingkungan fisik dimana data tersebut akan disimpan (misalnya disket, *hardisk*, *file* kartu dan sebagainya) yang digambarkan dalam bentuk simbol-simbol.

Adapun simbol-simbol yang dapat digunakan untuk menggambarkan *DFD* dari sistem yang akan dirancang atau sistem yang sedang berjalan di suatu perusahaan, antara lain dapat dilihat pada tabel berikut ini:

Tabel 2.1 Empat simbol dasar Data Flow Diagram

Simbol	Arti	Contoh
	Mahasiswa	
	Aliran Data	
	Proses	
	Penyimpanan Data	

Sumber: Kenneth E. Kendall & Julie E. Kendall, 2003:40

Empat simbol dasar yang digunakan untuk menetapkan gerakan diagram aliran data adalah sebagai berikut:

1. Entitas.

Kotak rangkap dua digunakan untuk menggambarkan suatu sistem 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 luar batas-batas sistem. Entitas-entitas tersebut harus diberi nama dengan sebuah 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. Aliran Data.

Tanda panah menunjukkan perpindahan data dari satu titik ke titik yang lain, dengan kepala tanda panah mengarah ke tujuan data. Aliran data yang muncul secara simultan bisa digambarkan hanya dengan menggunakan tanda panah paralel. Karena sebuah tanda panah menunjukkan seseorang, tempat, atau sesuatu, maka harus digambarkan dalam kata benda.

3. Proses.

Bujur sangkar dengan sudut membulat digunakan untuk menunjukkan adanya proses transformasi. Proses-proses tersebut selalu menunjukkan suatu perusahaan di dalam atau perubahan data. Jadi, aliran 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.

4. Penyimpanan Data.

Simbol dasar terakhir yang digunakan dalam diagram aliran data adalah bujur sangkar dengan ujung terbuka, yang menunjukkan penyimpanan data. Bujur sangkar yang digambarkan dengan dua garis paralel yang tertutup oleh sebuah garis pendek di sisi sebelah kanan. Simbol-simbol ini digambarkan hanya dengan lebar secukupnya saja sehingga memungkinkan menandai bentuk huruf-huruf diantara garis-garis paralel yang ada. Dalam diagram aliran data logika, jenis penyimpanan fisik (sebagai contoh: *tape*, disket) tidak ditetapkan. Pada poin ini simbol

penyimpanan data menunjukkan tempat penyimpanan untuk data-data yang memungkinkan penambahan dan perolehan data [11].

Langkah-langkah dalam membuat diagram *DFD* adalah sebagai berikut [11]:

1. Menciptakan Diagram Konteks.

Diagram konteks adalah tingkatan tertinggi dalam diagram aliran data dan hanya memuat satu proses, menunjukkan sistem secara keseluruhan. Proses tersebut diberi nomor. 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 aliran data menuju dari sistem diketahui analisa sistem dari wawancara dengan pengguna dan sebagai hasil analisis dokumen.

2. Menggambar Diagram Level 0 (Level berikutnya).

Lebih mendetail dibanding diagram konteks yang diperbolehkan, bisa dicapai dengan “mengembangkan diagram”. Masukan dan keluaran yang ditetapkan dalam diagram yang pertama tetap konstan dalam semua diagram sub-urutannya. Sisa diagram asli dikembangkan ke dalam gambar terperinci yang melibatkan tiga sampai sembilan proses yang menunjukkan penyimpanan data dan aliran data baru pada level yang lebih rendah.

3. Menciptakan Diagram Anak (Tingkatan yang lebih mendetail).

Setiap proses dalam diagram 0 bisa dikembangkan untuk menciptakan diagram anak yang lebih mendetail. Proses pada diagram 0 yang dikembangkan itu disebut *parent process* (proses induk) dan diagram yang dihasilkan disebut *child diagram* (diagram anak). Aturan utama untuk menciptakan diagram anak, keseimbangan vertikal, menyatakan bahwa suatu diagram anak tidak bisa menghasilkan keluaran atau menerima masukan di mana proses induknya juga tidak menghasilkan atau menerima. Semua aliran data yang menuju atau keluar dari proses induk harus ditunjukkan mengalir ke dalam atau keluar dari diagram anak.

Aturan-aturan dalam *DFD* adalah sebagai berikut [11]:

1. Diagram aliran data harus memiliki sedikitnya satu proses dan tidak boleh mempunyai objek-objek yang berdiri sendiri atau objek yang terkoneksi ke dirinya sendiri.
2. Sebuah proses harus menerima sedikitnya satu aliran data dan membuat sedikitnya satu aliran data. Proses-proses dengan seluruh masukan atau seluruh keluaran tidak boleh terjadi.
3. Sebuah penyimpanan data harus terkoneksi sedikitnya ke satu proses.
4. Entitas-entitas eksternal tidak boleh saling dikoneksikan satu sama lain. Meskipun mereka berkomunikasi secara independen, komunikasi tersebut bukanlah bagian sistem yang sedang dirancang.
5. Aliran data yang keluar dari sebuah proses tidak boleh bercabang bila akan menjadi aliran data masuk untuk proses-proses yang lain. Aliran data yang keluar dari sebuah proses tidak boleh menjadi aliran data masuk untuk proses itu sendiri.

2.7 Diagram Sebab-Akibat (*Fishbone Diagram*)

Fishbone diagram (diagram tulang ikan karena bentuknya seperti tulang ikan) sering juga disebut *Cause and Effect Diagram* atau *Ishikawa Diagram* diperkenalkan oleh Dr. Kaoru Ishikawa, seorang ahli pengendalian kualitas dari Jepang, sebagai satu dari tujuh alat kualitas dasar (*7 basic quality tools*). *Fishbone diagram* digunakan ketika kita ingin mengidentifikasi kemungkinan penyebab masalah dan terutama ketika sebuah *team* cenderung jatuh berpikir pada rutinitas (Tague, 2005, p. 247) [14].

Fishbone diagram akan mengidentifikasi berbagai sebab potensial dari satu efek atau masalah, dan menganalisis masalah tersebut melalui sesi *brainstorming*. *Brainstorming* adalah teknik kreativitas kelompok dimana upaya dilakukan untuk menemukan kesimpulan atas masalah tertentu dengan mengumpulkan daftar gagasan yang secara spontan disumbangkan oleh anggotanya. Istilah itu dipopulerkan oleh Alex Faickney Osborn dalam buku *Imej Imajinasi* tahun 1953. Masalah akan dipecah menjadi sejumlah kategori yang berkaitan, mencakup manusia, material, mesin, prosedur, kebijakan, dan sebagainya. Setiap kategori mempunyai sebab-sebab yang perlu diuraikan melalui sesi *brainstorming* [14].

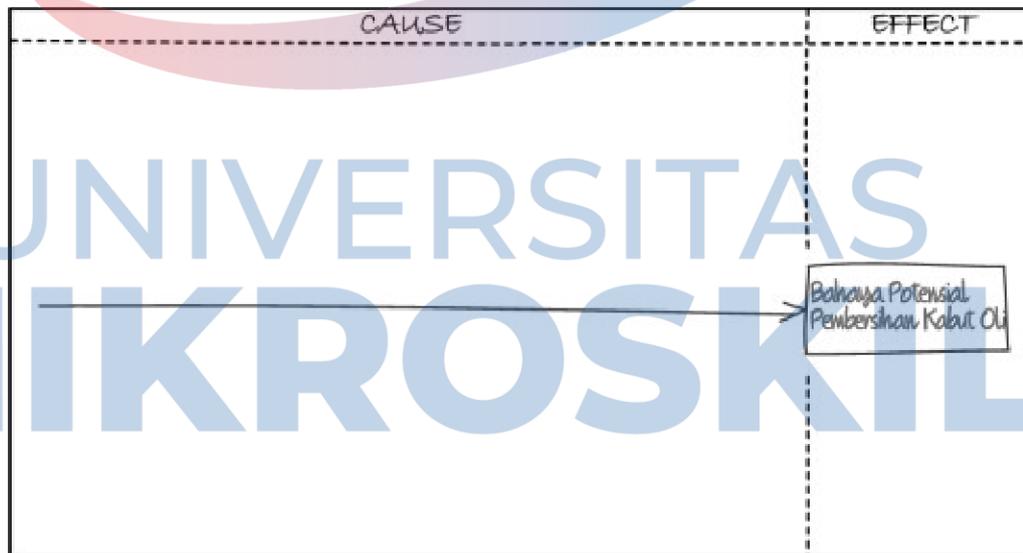
Untuk lebih jelasnya, akan diuraikan prosedur atau langkah-langkah pembuatan *fishbone diagram* di bawah ini [14].

Langkah-langkah pembuatan *Fishbone Diagram* [14]:

Pembuatan *fishbone diagram* kemungkinan akan menghabiskan waktu sekitar 30-60 menit dengan peserta terdiri dari orang-orang yang kira-kira mengerti / paham tentang masalah yang terjadi, dan tunjuklah satu orang pencatat untuk mengisi *fishbone diagram*. Alat-alat yang perlu disiapkan adalah: *flipchart* (kertas besar) atau *whiteboard* dan *marking pens* atau *spidol*.

1. Langkah I: Menyepakati Pernyataan Masalah.

- Sepakati sebuah pernyataan masalah (*problem statement*). Pernyataan masalah ini diinterpretasikan sebagai “effect”, atau secara visual dalam *fishbone* seperti “kepala ikan”.
- Tuliskan masalah tersebut di tengah *whiteboard* di sebelah paling kanan, misal: “Bahaya Potensial Pembersihan Kabut Oli”.
- Gambarkan sebuah kotak mengelilingi tulisan pernyataan masalah tersebut dan buat panah horizontal panjang menuju ke arah kotak (lihat **Gambar 2.3**).

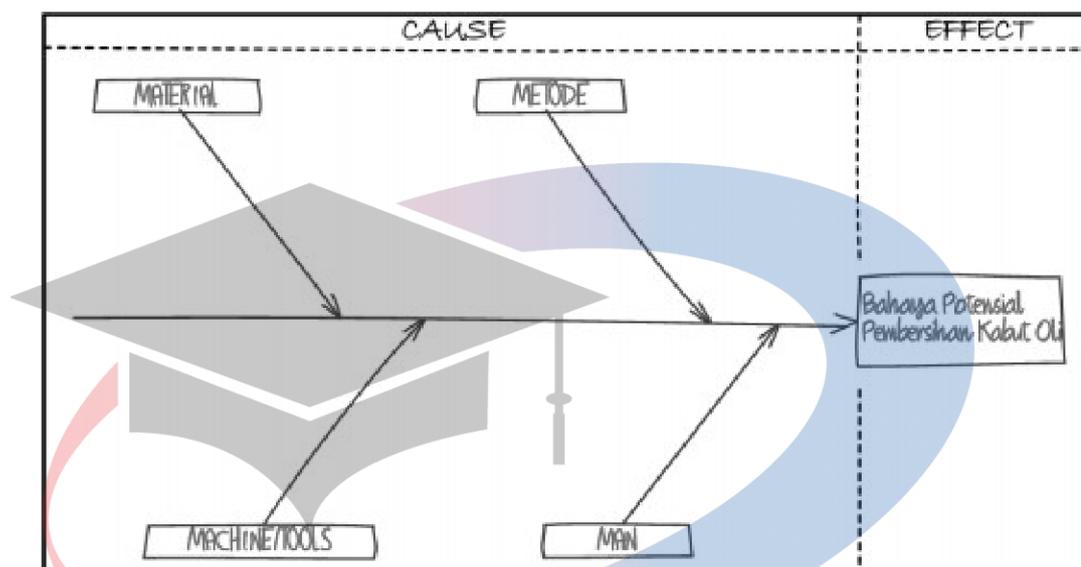


Gambar 2.3 Pembuatan *Fishbone Diagram* Menyepakati Pernyataan Masalah

2. Langkah II: Mengidentifikasi Kategori-Kategori.

- Dari garis horisontal utama, buat garis diagonal yang menjadi “cabang”. Setiap cabang mewakili “sebab utama” dari masalah yang ditulis. Sebab ini diinterpretasikan sebagai “*cause*”, atau secara visual dalam *fishbone* seperti “tulang ikan”.
- Kategori sebab utama mengorganisasikan sebab sedemikian rupa sehingga masuk akal dengan situasi. Kategori-kategori ini antara lain:
 - a. Kategori 6M yang biasa digunakan dalam industri manufaktur:
 1. *Machine* (mesin atau teknologi)
 2. *Method* (metode atau proses)
 3. *Material* (termasuk *raw material*, *consumption*, dan informasi),
 4. *Man Power* (tenaga kerja atau pekerjaan fisik) / *Mind Power* (pekerjaan pikiran: saran),
 5. *Measurement* (pengukuran atau inspeksi), dan
 6. *Milieu / Mother Nature* (lingkungan).
 - b. Kategori 8P yang biasa digunakan dalam industri jasa:
 1. *Product* (produk atau jasa)
 2. *Price* (harga),
 3. *Place* (tempat),
 4. *Promotion* (promosi atau hiburan),
 5. *People* (orang),
 6. *Process* (proses),
 7. *Physical Evidence* (bukti fisik), dan
 8. *Productivity & Quality* (produktivitas dan kualitas).
 - c. Kategori 5S yang biasa digunakan dalam industri jasa:
 1. *Surroundings* (lingkungan)
 2. *Suppliers* (pemasok),
 3. *Systems* (sistem),
 4. *Skills* (keterampilan), dan
 5. *Safety* (keselamatan).

Kategori di atas hanya sebagai saran, kita bisa menggunakan kategori lain yang dapat membantu mengatur gagasan-gagasan. Jumlah kategori biasanya sekitar 4 sampai dengan 6 kategori. Kategori pada contoh ini (lihat **Gambar 2.4**).



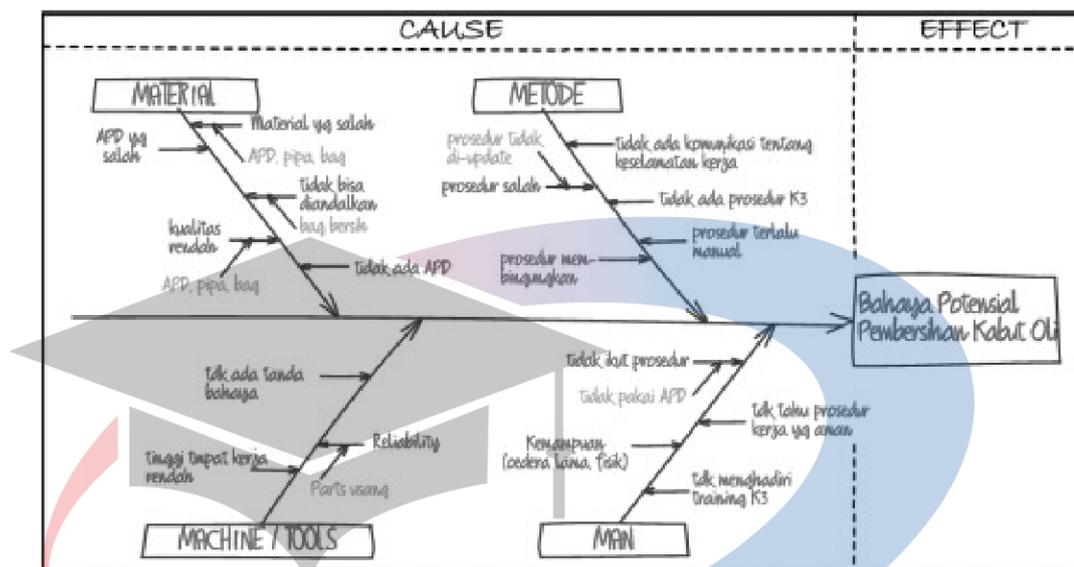
Gambar 2.4 Pembuatan *Fishbone Diagram* Menyepakati Pernyataan Masalah

3. Langkah III: Menemukan sebab-sebab potensial dengan cara *brainstorming*.

- Setiap kategori mempunyai sebab-sebab yang perlu diuraikan melalui sesi *brainstorming*.
- Saat sebab-sebab dikemukakan, tentukan bersama-sama di mana sebab tersebut harus ditempatkan dalam *fishbone diagram*, yaitu tentukan di bawah kategori yang mana gagasan tersebut harus ditempatkan, misal: “Mengapa bahaya potensial? Penyebab: Karyawan tidak mengikuti prosedur!” Karena penyebabnya karyawan (manusia), maka diletakkan di bawah “Man”.
- Sebab-sebab ditulis dengan garis horisontal sehingga banyak “tulang” kecil keluar dari garis diagonal.
- Pertanyakan kembali “Mengapa sebab itu muncul?” sehingga “tulang” lebih kecil (sub-sebab) keluar dari garis horisontal tadi, misal: “Mengapa karyawan disebut tidak mengikuti prosedur? Jawab: karena tidak memakai APD” (lihat **Gambar 2.5**).

Gambar 2.5).

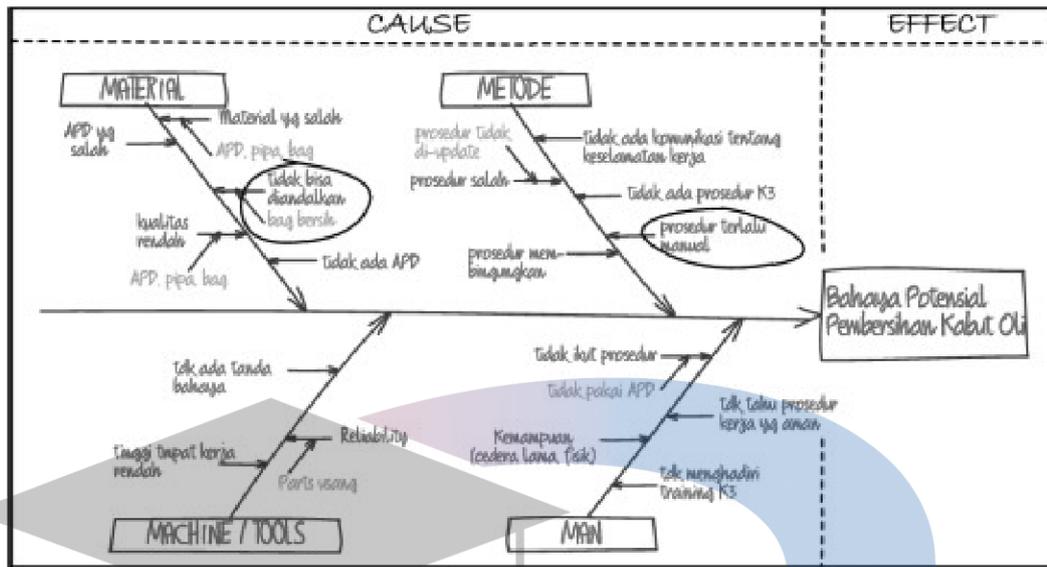
- Satu sebab bisa ditulis di beberapa tempat jika sebab tersebut berhubungan dengan beberapa kategori.



Gambar 2.5 Pembuatan *Fishbone Diagram* Menemukan Sebab-Sebab Potensial

4. Langkah IV: Mengkaji dan menyepakati sebab-sebab yang paling mungkin.

- Setelah setiap kategori diisi carilah sebab yang paling mungkin di antara semua sebab-sebab dan sub-subnya.
- Jika ada sebab-sebab yang muncul pada lebih dari satu kategori, kemungkinan merupakan petunjuk sebab yang paling mungkin.
- Kaji kembali sebab-sebab yang telah didaftarkan (sebab yang tampaknya paling memungkinkan) dan tanyakan, “Mengapa ini sebabnya?”
- Pertanyaan “Mengapa?” akan membantu kita sampai pada sebab pokok dari permasalahan teridentifikasi.
- Tanyakan “Mengapa?” sampai saat pertanyaan itu tidak bisa dijawab lagi. Kalau sudah sampai ke situ sebab pokok telah teridentifikasi.
- Lingkarilah sebab yang tampaknya paling mungkin *pada fishbone diagram* (lihat **Gambar 2.6**).



Gambar 2.6 Pembuatan *Fishbone Diagram* Melingkari Sebab yang Paling Mungkin

Diskusi selama sesi *brainstorming* hendaknya dirangkum, seperti terlihat pada **Tabel 2.2** di bawah ini.

Tabel 2.2 Rangkuman diskusi pada sesi *brainstorming fishbone diagram*

Possible Root Cause	Discussion	Root Cause?
MAN		
Kemampuan karyawan melakukan tugas (cedera lama, fisik)	Cedera personil teridentifikasi saat briefing K3*. Pelaksanaan tugas tidak tergantung pada fisik.	N
Tidak tahu prosedur K3	Awareness training di OJT sudah disediakan	N
Tidak mengikuti prosedur K3	Karyawan baru di-briefing K3 dan sistem penalty	N
Tidak menghadiri training K3	Pelatihan K3 diberikan dalam orientasi dan OJT	N

MACHINE / TOOLS		
Tinggi tempat kerja rendah	Bukan akar masalah jika metode dapat diubah	N
Part sudah usang	Tidak ada part usang menyebabkan insiden	N
Tidak ada tanda bahaya	Tanda bahaya sudah ada	N
METHOD		
Prosedur tidak diperbaharui	Review prosedur rutin setahun sekali	N
Tidak ada prosedur K3	Prosedur meliputi prosedur K3 untuk semua kegiatan	N
Prosedur K3 salah	Prosedur sudah ditinjau oleh supervisor, manajer, dept. head	N
Prosedur K3 membingungkan	Prosedur sudah ditinjau oleh supervisor, manajer, dept. head	N
Prosedur terlalu manual	Bag dipegang operator, perlu memastikan tidak ada kebocoran oli, dll.	Y
Tidak ada komunikasi K3	Disertakan dalam OJT	N
MATERIAL		
APD** yang salah	Verifikasi dengan vendor sebelum membeli	N
Material yang tidak bisa diandalkan bahan (bag kimia	Bag plastik rentan robek bila menyentuh objek tajam	Y

Kualitas rendah (pipa, APD, bag kimia)	Verifikasi dengan vendor sebelum membeli	N
Material yang digunakan salah (pipa, APD, bag kimia)	Verifikasi dengan vendor sebelum membeli	N
Tidak ada APD yang disediakan	APD sudah disediakan untuk semua aktivitas berbahaya	N

*) K3 = Kesehatan dan Keselamatan Kerja

***) APD = Alat Pelindung Diri

Dari contoh di atas, *fishbone diagram* dapat menemukan akar permasalahan, yaitu kabut oli selama ini dibersihkan dengan ditampung di *bag* plastik yang rentan robek dan selama tidak ada *bag* plastik ada kemungkinan oli menetes jika kran rusak, solusi bisa dengan menambahkan *containment tray* atau *safety cabinet* yang permanen menempel pada pipa.

Jika masalah rumit dan waktunya memungkinkan, kita bisa meninggalkan *fishbone diagram* di dinding selama beberapa hari untuk membiarkan ide menetas dan membiarkan orang yang lalu lalang turut berkontribusi. Jika *fishbone diagram* terlihat timpang atau sempit, kita bisa mengatur ulang *fishbone diagram* dengan kategori sebab utama yang berbeda. Kunci sukses *fishbone diagram* adalah terus bertanya “Mengapa?”, lihatlah diagram dan carilah pola tanpa banyak bicara, dan libatkan orang-orang di “grass root” yang terkait dengan masalah karena biasanya mereka lebih mengerti permasalahan di lapangan.

2.8 Kamus Data (*Data Dictionary*)

Kamus data adalah suatu aplikasi khusus dari jenis kamus-kamus yang digunakan sebagai referensi kehidupan setiap hari. Kamus data merupakan hasil referensi data mengenai data, suatu data yang disusun oleh penganalisis sistem untuk membimbing mereka selama melakukan analisis dan design. Sebagai suatu dokumen,

kamus data mengumpulkan dan mengkoordinasikan istilah-istilah data tertentu, dan menjelaskan apa arti setiap istilah yang ada [11].

Berikut adalah simbol-simbol yang digunakan dalam kamus data:

Tabel 2.3 Simbol-simbol kamus data

Notasi	Keterangan
=	Terdiri dari
+	Dan
{ }	Elemen-elemen repetitif (kelompok berulang)
[]	Salah satu dari dua situasi tertentu
()	Pilihan (boleh dikosongkan)

Sumber: Kenneth E. Kendall & Julie E. Kendall, 2003, Jilid 1, 338

Kamus data otomatis sangat berguna karena memiliki kapasitas dalam hal referensi silang *item-item* data, dengan demikian memungkinkan dilakukannya perubahan-perubahan program terhadap semua program yang berbagi suatu elemen biasa. Fitur ini menggantikan pengubahan program serampangan atau mencegah penundaan sampai program tidak bisa berjalan karena perubahan tersebut tidak diimplementasikan terhadap program yang telah diperbaharui. Jelasnya kamus data otomatis menjadi sangat penting untuk sistem-sistem besar karena mampu menghasilkan ribuan elemen data yang dikatalogkan dan dibuat referensi silang [11].

Sebagai Tambahan untuk dokumentasi serta mengurangi redundansi, kamus data bisa digunakan untuk [11]:

1. Melakukan validasi diagram aliran data dalam hal kelengkapan dan keakuratan.
2. Menyediakan suatu titik awal untuk mengembangkan layar dan laporan-laporan.
3. Menentukan muatan data yang disimpan di *file*.

Dalam kamus data dikenal adanya struktur data. Struktur data adalah bagian struktur yang biasanya terdiri dari beberapa *item* tertentu yang memuat informasi-informasi mengenai elemen-elemen pada kamus data. Struktur data biasanya digambarkan dengan menggunakan notasi aljabar. Berikut ini adalah simbol notasi struktur data dalam kamus data:

1. Tanda sama dengan (=), artinya "terdiri dari".
2. Tanda plus (+), artinya "dan".
3. Tanda kurung kurawal { }, artinya menunjukkan elemen-elemen repetitif, juga disebut kelompok berulang atau tabel-tabel.
4. Tanda kurung siku [], artinya menunjukkan salah satu dari dua situasi tertentu. Satu elemen bisa ada sedangkan elemen lainnya juga bisa ada tetapi tidak bisa keduanya ada secara bersamaan. Elemen-elemen yang ada di dalam tanda kurung ini saling terpisah satu sama lain.
5. Tanda kurung (), merupakan suatu elemen yang bersifat pilihan. Elemen-elemen yang bersifat pilihan ini bisa dikosongkan pada layar masukan atau bisa juga dengan memuat spasi atau nol untuk *field-field* numerik pada struktur *file*.

2.9 Basis Data

Database merupakan kumpulan data dan informasi dalam jumlah yang tidak sedikit. Oleh karena itu *database* harus disusun sebagai kriteria terpola dengan jelas sejak dari awalnya. *Database* akan diubah ke dalam bentuk *table* dalam *software Database Manajemen System* dengan memperhatikan: *File Table*, *Field*, *Primary Key* dan sebagainya yang merupakan bagian dari *database*. Singkatnya langkah awal pengolahan suatu unit data itu dilakukan di dalam *database*.

Pembuatan *database* dalam *Database Manajemen System* mencakup 2 (dua) tindakan utama, yakni: tindakan penentuan dari struktur dari *database* yang mencakup penentuan bagian-bagian dari *database* dan tindakan pengisian data ke dalamnya dengan melakukan pengetikan data ke dalam struktur tersebut.

Sebagai suatu jenis program aplikasi yang berguna untuk mengorganisasikan sejumlah data yang ada dengan model *relational*, *Database Manajemen System* diimplementasikan dalam bentuk tabel yang terdiri dari baris dan kolom. Baris pada *Database Manajemen System* dikenal dengan istilah *record* yang merupakan

kumpulan informasi yang ada di dalam *field*. Sementara kolom yang di kenal dengan istilah *Field* merupakan tempat di mana informasi ditampung. Setiap *Field* dalam *database* akan berelasi dengan *field* lainnya atau dengan data lainnya.

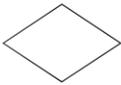
Sistem basis data adalah suatu sistem menyusun dan mengelola *record-record* menggunakan komputer untuk menyimpan atau merekam serta memelihara data operasional lengkap sebuah organisasi/perusahaan sehingga mampu menyediakan informasi yang optimal yang diperlukan pemakai untuk proses mengambil keputusan [12].

Tujuan basis data yang efektif yaitu [11]:

1. Memastikan bahwa data dapat dipakai diantara pemakai untuk berbagai aplikasi.
2. Memelihara data baik keakuratan maupun kekonsistenannya.
3. Memastikan bahwa semua data yang diperlukan untuk aplikasi sekarang dan yang akan datang akan disediakan dengan cepat.
4. Membolehkan basis data untuk berkembang dan kebutuhan pemakai untuk berkembang.
5. Membolehkan pemakai untuk membangun pandangan personalnya tentang data tanpa memperhatikan cara data disimpan secara fisik.

Berikut adalah simbol hubungan entitas beserta penjelasan dan artinya:

Tabel 2.4 Simbol hubungan entitas dan artinya

Simbol	Penjelasan Resmi	Arti Sebenarnya
	Entitas	Sekelompok orang, tempat atau sesuatu
	Entitas Terhubung	Digunakan untuk menghubungkan dua entitas
	Entitas Terhubung	Digunakan untuk kelompok terulang
	Entitas Attribut	Tepat satu
	Kebanyak hubungan	Satu atau lebih
	Ke 0 atau 1 hubungan	Hanya satu atau nol
	Ke lebih dari satu hubungan	Lebih besar dari satu

Sumber: Kenneth E. Kendall & Julie E. Kendall, 2003, Jilid 2, 133

2.10 Normalisasi

Proses normalisasi merupakan proses pengelompokan elemen data menjadi tabel-tabel yang menunjukkan entitas dan relasinya. Proses ini selalu diuji pada beberapa kondisi. Apakah ada kesulitan pada saat menambah (*insert*), menghapus (*delete*), mengubah (*update*), membaca (*retrieve*) pada satu *database*. Bila ada kesulitan pada pengujian tersebut maka relasi dapat dipecah pada beberapa tabel lagi. Dengan kata lain perancangan belum mendapatkan *database* yang optimal. Ada beberapa konsep yang harus diketahui lebih dahulu seperti *field* atau atribut kunci dan ketergantungan kunci (*functional dependency*) [4].

Setiap *file* selalu memiliki kunci yang berupa satu *field* atau satu set *field* yang dapat mewakili *record*. Misalnya nomor pegawai yang merupakan kunci dari tabel pegawai di suatu perusahaan. Untuk melakukan pencarian dapat dilakukan dengan menyebut nomor pegawai tersebut, maka kemudian akan diketahui nama, alamat dan atribut lainnya mengenai pegawai tersebut. Ada beberapa macam kunci (*key function*) yang digunakan untuk proses pencarian, penyaringan, penghapusan dan lainnya, yaitu sebagai berikut:

1. Kunci Kandidat (*Candidate Key*).

Kunci kandidat adalah satu atribut atau satu set minimal atribut yang mengidentifikasi secara unik suatu kejadian yang spesifik dari suatu entitas. Satu minimal set dari atribut menyatakan secara tak langsung dimana tidak dapat membuang beberapa atribut dalam satu set tanpa merusak kepemilikan unik. Jika satu kunci kandidat berisi lebih dari satu atribut, biasanya disebut sebagai *composite key* (kunci gabungan).

2. Kunci Primer (*Primary Key*).

Kunci primer adalah satu atribut atau satu set minimal atribut yang tidak hanya mengidentifikasi secara unik suatu kejadian yang spesifik, akan tetapi juga dapat mewakili setiap kejadian dari suatu entitas. Setiap kunci kandidat punya peluang menjadi *primary key*, akan tetapi sebaiknya dipilih satu saja yang dapat mewakili entitas yang ada secara menyeluruh.

3. Kunci Alternatif (*Alternative Key*).

Kunci alternatif adalah kunci kandidat yang tidak dipakai sebagai *primary key*. Kerap kali kunci alternatif ini dipakai sebagai kunci pengurutan dalam pembuatan laporan.

4. Kunci Tamu (*Foreign Key*).

Kunci tamu adalah satu atribut atau satu set atribut yang melengkapi satu *relationship* (hubungan) yang menunjukkan ke induknya. Kunci tamu ditempatkan pada entitas anak dan sama dengan *primary key* induk direlasikan. Hubungan antara entitas induk dengan anak adalah hubungan satu lawan banyak (*one to many relationship*).

Teknik normalisasi juga merupakan suatu teknik yang menstrukturkan data dalam cara tertentu untuk membantu mengurangi atau mencegah timbulnya masalah yang berhubungan dengan pengolahan data dalam *database*. Proses normalisasi selalu diuji pada beberapa tabel lagi atau dengan kata lain perancangan belum mendapat *database* yang optimal. Pada proses normalisasi ini perlu dikenal dahulu definisi dari tahap normalisasi, yaitu sebagai berikut:

1. Bentuk Tidak Normal (*Unnormalized Form*).

Bentuk ini merupakan kumpulan data yang akan direkam, tidak ada keharusan untuk mengikuti suatu format tertentu, dapat saja data tidak lengkap atau terduplikasi. Data dikumpulkan apa adanya sesuai dengan kedatangannya.

2. Bentuk Normal Pertama (*First Normal Form*).

Bentuk normal pertama mempunyai ciri yaitu bahwa setiap data dibentuk dalam *flat file* (*file* datar atau rata), data dibentuk dalam satu *record* demi *record* dan nilai dari *field* berupa *atomic value*. Tidak ada set atribut yang berulang atau atribut bernilai ganda (*multi value*).

3. Bentuk Normal Kedua (*Second Normal Form*).

Bentuk normal kedua mempunyai syarat, yaitu bentuk data telah memenuhi kriteria bentuk normal pertama. Atribut bukan kunci haruslah bergantung secara fungsi pada kunci utama. Dengan demikian untuk membentuk normal kedua haruslah sudah ditentukan kunci *field*. Kunci *field* haruslah unik dan dapat mewakili atribut lain yang menjadi anggotanya.

4. Bentuk Normal Ketiga (*Thrid Normal Form*).

Untuk menjadi bentuk normal ketiga maka relasi haruslah dalam bentuk normal kedua dan semua atribut bukan *primer* tidak punya hubungan yang transitif. Dengan kata lain, setiap atribut bukan kunci haruslah bergantung hanya pada *primary key* dan pada *primary key* secara menyeluruh.

UNIVERSITAS
MIKROSKIL