

BAB II

KAJIAN LITERATUR

1.1 Konsep Sistem Informasi

Secara umum, konsep dari sistem informasi disusun oleh 2 bagian yaitu sistem dan informasi.

2.1.1 Sistem

Sistem adalah suatu kumpulan dari prosedur-prosedur, saling berhubungan dan membentuk jaringan kerja untuk mencapai tujuan tertentu. Secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu [4].

Sistem merupakan kumpulan dari beberapa bagian yang memiliki keterkaitan dan saling bekerja sama serta membentuk suatu kesatuan untuk mencapai tujuan dari sistem tersebut [5].

Sistem adalah sekumpulan unsur/elemen yang saling berkaitan dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan [6].

Suatu sistem untuk mencapai tujuan tertentu harus memiliki sifat-sifat dasar yang menjadikannya suatu karakter. Suatu sistem untuk mencapai tujuan tertentu harus memiliki sifat-sifat dasar yang menjadikannya suatu karakter. Karakteristik suatu sistem terdiri dari komponen-komponen (*components*), batas sistem (*boundary*), lingkungan luar sistem (*environments*), penghubung (*interface*), masukan (*input*), keluaran (*output*), pengolah (*process*), dan sasaran (*objectives*) atau tujuan (*goal*) [4].

1. Komponen Sistem

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu sub sistem. Setiap sub sistem mempunyai sifat-sifat dari sistem yang menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan.

2. Batas Sistem

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem lainnya atau lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan.

3. Lingkungan Luar Sistem

Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem, dapat bersifat menguntungkan maupun merugikan. Lingkungan luar yang menguntungkan merupakan energi dari sistem yang harus dijaga dan dipelihara. Sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, karena akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung Sistem

Penghubung merupakan media penghubung antara satu sub sistem dengan sub sistem lainnya. Keluaran (*output*) dari satu sub sistem akan menjadi masukan (*input*) untuk sub sistem yang lainnya dengan melalui penghubung.

5. Masukan Sistem

Masukan sistem adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran.

6. Keluaran Sistem

Keluaran sistem adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisi pembuangan. Keluaran dapat merupakan masukan untuk sub sistem yang lain atau kepada super sistem.

7. Pengolah Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. Pengolah yang akan mengubah masukan menjadi keluaran.

8. Sasaran sistem

Suatu sistem pasti mempunyai sasaran (*objectives*) atau tujuan (*goal*), kalau tidak mempunyai sasaran maka operasi sistem tidak akan ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

Berdasarkan informasi dari berbagai sumber tersebut, dapat disimpulkan bahwa sistem adalah suatu entitas terorganisir yang terdiri dari berbagai komponen atau unsur yang saling

berinteraksi dan bekerja sama untuk mencapai tujuan tertentu. Sistem memiliki karakteristik dasar yang mencakup komponen sistem, batas sistem, lingkungan luar sistem, penghubung sistem, masukan sistem, keluaran sistem, pengolah sistem, dan sasaran sistem.

2.1.2 Informasi

Informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang [7].

Informasi adalah data yang telah diproses, diorganisasikan atau diringkas menjadi sesuatu yang berguna [5].

Informasi adalah aset yang sangat vital dalam sebuah organisasi dan wajib diproteksi karena informasi merupakan aset yang berharga bagi organisasi [8].

Informasi harus memiliki nilai jika dapat mengubah tindakan yang diambil. Perubahan ini penting untuk dipahami karena data sebenarnya tidak memiliki nilai dalam pengambilan keputusan, hanya informasi yang memiliki nilai, dalam arti informasi memudahkan manajer dalam mengambil keputusan. Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Sumber informasi adalah data. Data nyata akan menggambarkan suatu kejadian-kejadian dan kesatuan nyata [7].

Data yang diolah untuk menghasilkan informasi menggunakan model proses yang tertentu. Data yang diolah melalui suatu model menjadi informasi, kemudian penerima menerima informasi tersebut, yang berarti menghasilkan suatu keputusan dan melakukan tindakan yang lain akan membuat sejumlah data kembali. Data tersebut akan di *input*, diproses kembali lewat suatu model yang disebut dengan siklus informasi (*information cycle*). Siklus ini juga disebut dengan siklus pengelolaan data (*data processing cycles*). Kualitas informasi terdiri dari 3 hal yaitu [4]:

1. Informasi harus akurat (*accurate*) berarti informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan.
2. Tepat pada waktunya (*time lines*) merupakan informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah asing tidak akan memiliki nilai lagi.

3. Relevan (*relevance*) berarti informasi tersebut mempunyai manfaat untuk pemakaiannya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda.

Berdasarkan informasi dari berbagai sumber tersebut, dapat disimpulkan bahwa informasi adalah hasil pengolahan data yang telah diproses, diorganisir, atau diringkas sehingga menjadi sesuatu yang memiliki nilai dan kegunaan bagi penerimanya. Informasi memiliki peran penting sebagai dasar dalam pengambilan keputusan, baik secara langsung pada saat itu juga maupun secara tidak langsung pada masa mendatang.

Lebih lanjut, informasi dianggap sebagai aset vital dalam sebuah organisasi dan perlu diproteksi karena nilainya yang berharga. Proses transformasi data menjadi informasi melibatkan suatu model proses, yang sering disebut sebagai siklus informasi atau siklus pengelolaan data. Data yang diolah melalui siklus ini menjadi informasi, yang kemudian diterima oleh penerima, menghasilkan keputusan, dan memicu tindakan yang dapat menghasilkan sejumlah data kembali.

Kualitas informasi juga menjadi hal krusial, dengan tiga elemen utama, yaitu keakuratan (*accurate*), ketepatan waktu (*time lines*), dan relevansi (*relevance*). Informasi harus bebas dari kesalahan dan tidak menyesatkan (akurat), harus disampaikan pada waktu yang tepat tanpa terlambat (tepat pada waktunya), dan harus memiliki manfaat atau relevansi bagi pemakainya.

Dengan demikian, informasi bukan hanya sekadar data yang diolah, tetapi merupakan output yang bernilai tinggi, mampu membantu pengambilan keputusan, dan memiliki dampak yang signifikan pada tindakan organisasi. Oleh karena itu, perlindungan dan manajemen kualitas informasi menjadi suatu keharusan dalam lingkungan organisasi modern.

2.1.3 Sistem Informasi

Sistem informasi merupakan sekumpulan *hardware*, *software*, *brainware*, prosedur, dan atau aturan yang diorganisasikan secara integral untuk mengolah data menjadi informasi yang bermanfaat guna untuk memecahkan masalah dan pengambilan keputusan. Sistem informasi adalah kesatuan data olahan yang terintegrasi dan saling melengkapi yang menghasilkan *output* baik dalam bentuk gambar, suara maupun tulisan [4].

Sistem informasi adalah sebuah sistem yang terdiri dari entitas-entitas yang saling berhubungan dan bekerjasama untuk mengumpulkan, mengolah, menyimpan, mengelola, dan mendistribusikan informasi guna mendukung pengambilan keputusan, pengawasan, analisis, dan pengelolaan suatu organisasi atau bisnis [8].

Sistem informasi adalah suatu sistem yang menyediakan informasi untuk mengambil keputusan dan juga menjalankan suatu bisnis yang ada di perusahaan [9].

Sistem informasi juga dapat dikatakan sebagai kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi. Pendapat tersebut mengemukakan, bahwa sistem informasi merupakan kumpulan kegiatan yang diintegrasikan antara program kerja, informasi ke dalam suatu *server database* sehingga keinginan suatu organisasi dalam mencapai tujuan bisa terwujud [4].

Komponen sistem informasi terdiri dari [4]:

1. *Hardware* (perangkat keras), terdiri dari komputer, printer dan jaringan.
2. *Software*, kumpulan perintah yang ditulis dengan aturan untuk memerintah komputer melaksanakan tugas tertentu.
3. *Data*, merupakan komponen dasar dari informasi yang akan diproses lebih lanjut untuk menghasilkan informasi
4. Manusia, yang terlibat dalam komponen seperti operator dan pimpinan.
5. Prosedur, dokumentasi proses sistem buku penuntun operasional (aplikasi) dan teknis.

Sistem Informasi memiliki tujuan di antaranya adalah suatu sistem informasi yang baik haruslah relevan agar dapat memperoleh hasil yang bagus dalam menentukan suatu keputusan dalam sebuah kelompok. Seluruh komponen sistem informasi dan data-data, harus memberikan hasil yang bermanfaat sekurang-kurangnya sebanyak pengeluaran yang telah dilakukan [10].

Peranan sistem informasi yang sering kita jumpai dalam kehidupan sehari-hari, contoh di antaranya yaitu [10]:

1. Sistem akademis berbasis *website*, yang dapat mempermudah kita untuk mengakses data terkait akademisi.
2. Sistem pemerintahan berbasis *website*, atau *e-government*, yang dapat memberikan layanan pemerintahan melalui internet.
3. Sistem pertukaran data digital, dengan menggunakan komputer kita dapat bertukar data secara cepat.
4. Sistem biometrik yaitu sistem yang dikembangkan untuk keamanan informasi dengan menggunakan pola wajah dan sidik jari agar tidak sembarang orang melakukan akses.

Berdasarkan informasi dari berbagai sumber tersebut, dapat disimpulkan bahwa sistem informasi adalah suatu kumpulan komponen yang terintegrasi, termasuk *hardware*, *software*,

brainware, prosedur, dan aturan, yang bekerja bersama untuk mengolah data menjadi informasi yang bermanfaat. Sistem informasi tidak hanya mencakup teknologi, tetapi juga melibatkan manusia dan prosedur dalam rangka mencapai tujuan suatu organisasi atau bisnis.

Sistem informasi memiliki beberapa karakteristik, di antaranya adalah kemampuannya untuk mengumpulkan, mengolah, menyimpan, mengelola, dan mendistribusikan informasi. Tujuan utama dari sistem informasi adalah untuk mendukung pengambilan keputusan, pengawasan, analisis, dan pengelolaan suatu organisasi atau bisnis. Komponen-komponen sistem informasi melibatkan *hardware* (perangkat keras), *software* (program perintah), data, manusia (seperti operator dan pimpinan), dan prosedur (dokumentasi proses operasional dan teknis).

Selain itu, sistem informasi juga memiliki peran dalam kehidupan sehari-hari, seperti sistem akademis berbasis website, sistem pemerintahan berbasis website (e-government), sistem pertukaran data digital, dan sistem biometrik untuk keamanan informasi. Pentingnya relevansi dalam sistem informasi diutarakan sebagai salah satu tujuan utama, di mana hasil yang dihasilkan haruslah bermanfaat setidaknya sebanding dengan pengeluaran yang telah dilakukan.

Secara keseluruhan, sistem informasi merupakan suatu entitas kompleks yang memainkan peran krusial dalam mendukung kegiatan operasional, pengambilan keputusan, dan pencapaian tujuan organisasi atau bisnis.

1.2 Analisis dan Perancangan Sistem Informasi

Analisis dan perancangan sistem informasi adalah penjabaran seluruh informasi menjadi beberapa bagian untuk mengidentifikasi dan menilai berbagai masalah dan kebutuhan berbagai masalah dan kebutuhan yang diinginkan untuk membuat saran perbaikan. Analisis sistem informasi yang diterapkan dalam penelitian ini adalah identifikasi *stakeholder* dan pengguna, analisis masalah, identifikasi karakteristik, identifikasi kebutuhan sistem dan identifikasi kategori produk, serta pemodelan *use case* [11].

Analisis dan perancangan sistem informasi adalah proses mengubah analisis menjadi rencana desain perangkat lunak. Desain menciptakan representasi untuk model perangkat lunak, tetapi tidak seperti ketentuan (yang berfokus pada deskripsi informasi, fungsionalitas, dan perilaku yang diperlakukan), model desain memberikan informasi tentang arsitektur perangkat lunak, struktur data, antarmuka, dan komponen yang diperlukan untuk penerapan sistem [12].

Analisis dan perancangan sistem informasi adalah proses pencarian solusi atau pemecahan dari suatu masalah baik secara terstruktur, maupun berorientasi objek. Pengembangan secara terstruktur biasanya lebih menekankan pembuatan sistem berdasarkan proses kerja/prosedur yang telah ditetapkan [13].

Berdasarkan definisi dari berbagai sumber tersebut dapat disimpulkan analisis dan perancangan sistem informasi adalah suatu proses holistik yang melibatkan dekonstruksi informasi menjadi elemen-elemen terpisah. Langkah-langkah dalam proses ini melibatkan identifikasi stakeholder dan pengguna, analisis masalah, identifikasi karakteristik, identifikasi kebutuhan sistem, dan identifikasi kategori produk. Pemahaman mendalam terhadap elemen-elemen ini membantu dalam memberikan gambaran yang komprehensif terkait masalah dan kebutuhan yang perlu diatasi.

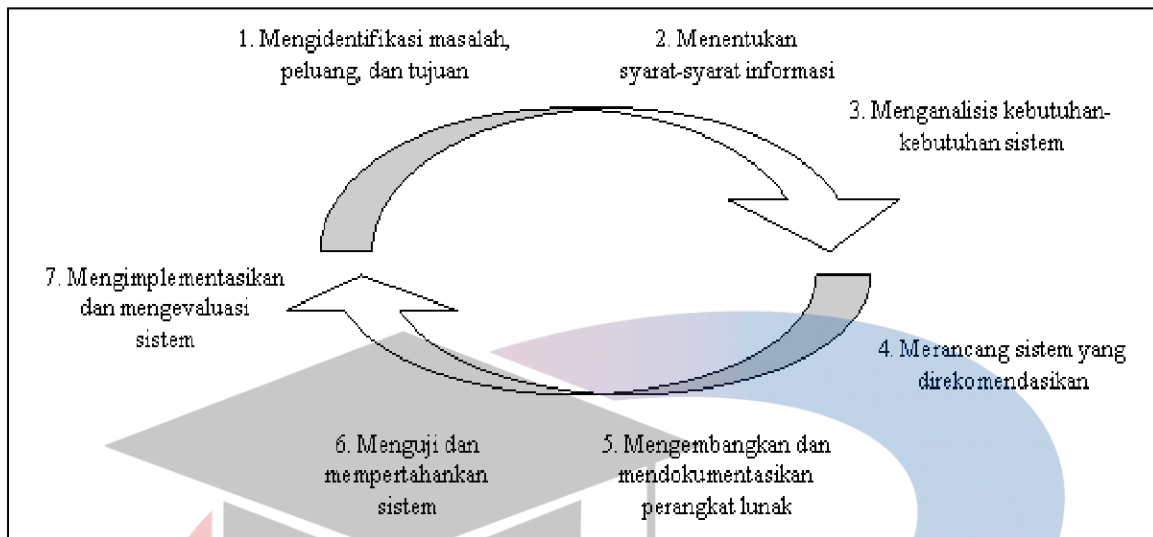
1.3 *System Development Life Cycle (SDLC)*

System Development Life Cycle (SDLC) atau sering disebut dengan siklus hidup pengembangan sistem merupakan pendekatan yang dilakukan melalui beberapa tahap untuk menganalisis dan merancang sistem yang juga telah dikembangkan dengan sangat baik melalui penggunaan siklus kegiatan penganalisis dan pemakai secara spesifik. Dari tersebut dapat diartikan bahwa siklus hidup pengembangan sistem merupakan rangkaian proses dalam menganalisis suatu sistem dalam mencapai suatu tujuan [14].

Systems Development Life Cycle (SDLC) merupakan pendekatan bertahap untuk analisis dan desain berdasarkan asumsi bahwa sistem lebih baik dikembangkan melalui penggunaan siklus aktivitas analisis dan pengguna tertentu [15].

Systems Development Life Cycle (SDLC) adalah suatu pendekatan sistematis yang digunakan dalam pengembangan perangkat lunak. Pendekatan ini membantu tim pengembangan perangkat lunak dalam mengatur proses pembuatan, pengembangan, pengujian, dan pemeliharaan perangkat lunak secara efisien dan terstruktur. Tujuan dari SDLC adalah untuk menghasilkan perangkat lunak berkualitas tinggi yang memenuhi kebutuhan pengguna, mengikuti tenggat waktu, dan berada dalam batas anggaran yang ditentukan. Model SDLC menyediakan panduan teori untuk pengembangan software. Model-model SDLC sangat penting untuk evolusi perangkat lunak secara sistematis sehingga dapat dikerjakan dalam batas waktu tertentu dengan kualitas yang baik [16].

Tahapan utama dari siklus hidup pengembangan sistem dibagi menjadi 7 fase seperti yang ditunjukkan seperti pada Gambar 2.1 berikut [14].



Gambar 2.1 Siklus Hidup Pengembangan Sistem

Berikut ini akan dijelaskan tahapan-tahapan dari siklus hidup pengembangan sistem, yaitu [14]:

1. Mengidentifikasi masalah, peluang, dan tujuan

Pada fase pertama dari siklus hidup pengembangan sistem, analis mengidentifikasi masalah, peluang, dan tujuan yang ingin dicapai. Analis memiliki pandangan yang jujur tentang apa yang terjadi dalam bisnis. Kemudian, bersama dengan anggota organisasi lainnya, analis mengidentifikasi masalah. Peluang adalah situasi di mana analis percaya bahwa perbaikan dapat dilakukan melalui penggunaan sistem informasi terkomputerisasi. Mengukur peluang yang memungkinkan bisnis memperoleh keunggulan kompetitif atau menetapkan standar industri. Analis harus mencari tahu apa yang terjadi di perusahaan. Hanya dengan demikian analis dapat melihat beberapa aspek aplikasi sistem informasi untuk membantu perusahaan mencapai tujuannya dengan mengatasi masalah atau peluang tertentu.

2. Menentukan syarat-syarat informasi

Pada langkah berikutnya, penganalisis mencakup segala sesuatu yang menentukan kebutuhan informasi dari pengguna yang terpengaruh. Selama fase kebutuhan informasi dari siklus pengembangan sistem, analis mencoba memahami informasi apa yang dibutuhkan pengguna agar muncul dalam pekerjaan mereka. Analis sistem perlu mengetahui detail fungsi sistem

yang ada: siapa (orang yang terlibat), apa (bisnis), dan di mana (lingkungan tempat bekerja), kapan (waktu yang baik), dan bagaimana (bagaimana prosedur bekerja).

3. Menganalisis kebutuhan sistem

Perangkat dan teknik–teknik tertentu akan membantu penganalisis menentukan kebutuhan yaitu penggunaan diagram aliran data untuk menyusun daftar input, proses dan output fungsi bisnis dalam bentuk grafik terstruktur. Dari diagram aliran data, dikembangkan suatu kamus data berisikan daftar seluruh item data yang digunakan dalam sistem, berikut spesifiknya apakah berupa alphanumeric atau teks, serta berapa banyak spasi yang dibutuhkan saat dicetak.

4. Merancang sistem yang direkomendasikan

Analisis merancang proses entri data sehingga data yang dimasukkan ke dalam sistem informasi benar-benar akurat. Selain itu, analisis menggunakan jenis desain dan teknik penyempurnaan tertentu untuk memastikan efektivitas input sistem informasi. Tahap desain juga mencakup perancangan file atau *database* yang dapat menyimpan data yang dibutuhkan oleh pengambil keputusan. *Database* yang terorganisir dengan baik adalah dasar dari semua sistem informasi. Pada tahap ini, analisis juga bekerja dengan pengguna untuk merancang *output* (baik di layar maupun di cetak). Terakhir, analisis harus merancang prosedur pencadangan dan pengendalian untuk melindungi sistem dan data, dan membuat paket spesifikasi program untuk pemrogram.

5. Mengembangkan dan mendokumentasikan perangkat lunak

Analisis bekerja dengan *programmer* untuk mengembangkan perangkat lunak asli yang dibutuhkan. Beberapa teknik terstruktur untuk merancang dan membuat profil perangkat lunak termasuk perencanaan terstruktur, grafik Nassi-Shneiderman, dan pseudocode. Penganalisis sistem menggunakan salah satu perangkat ini untuk memprogram apa yang perlu diprogram.

6. Menguji dan mempertahankan sistem

Sebelum sistem informasi digunakan, terlebih dahulu harus diuji. Beberapa pemeriksaan dilakukan oleh pemrogram itu sendiri dan beberapa oleh penganalisis sistem. Rangkaian pengujian ini pertama kali dilakukan dengan data sampel dan data aktual dari sistem yang ada.

Pemeliharaan sistem dan dokumentasinya dimulai pada tahap ini dan dilakukan sesering sistem informasi berjalan.

7. Mengimplementasikan dan mengevaluasi sistem

Di tahap terakhir dari pengembangan sistem, 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. Proses ini mencakup pengubahan *file-file* dari format lama ke format baru atau membangun suatu basis data, meng-*install* peralatan dan membawa sistem baru untuk diproduksi. Evaluasi yang ditunjukkan sebagai bagian dari tahap terakhir dari siklus hidup pengembangan sistem biasanya dimaksudkan untuk pembahasan. Kriteria utama yang harus dipenuhi ialah apakah pemakai yang dituju benar-benar menggunakan sistem.

1.4 Alat Bantu Perancangan Sistem

Ada beberapa alat bantu yang perlu digunakan untuk mempermudah dalam menggambarkan komponen yang ada, proses dan pemecahan masalah yang terjadi pada saat analisis sistem dan perancangan sistem.

2.3.1 *Fishbone Diagram*

Diagram tulang ikan atau *fishbone diagram* adalah alat analisis yang mewakili kemungkinan penyebab masalah sebagai garis besar grafis. Saat menggunakan diagram tulang ikan, seorang analis pertama-tama menyatakan masalahnya dan menggambar tulang utama dengan sub-tulang yang mewakili kemungkinan penyebab masalah. Di setiap area, analis mengidentifikasi kemungkinan penyebab dan menggambarkannya sebagai sub-tulang horizontal. Tujuan utama dari diagram tulang ikan adalah untuk menggambarkan secara grafik cara hubungan antara akibat dan semua faktor yang berpengaruh pada akibat tersebut [17].

Fishbone Diagram adalah bagan yang berbentuk seperti ikan yang digunakan untuk mengidentifikasi berbagai penyebab atau faktor utama yang mempengaruhi pengendalian kualitas masalah yang mencapai tujuan tertentu. Penyebab atau faktor utama tersebut dapat diuraikan menjadi banyak kategori terkait, termasuk orang, material, mesin, prosedur dan kebijakan [18].

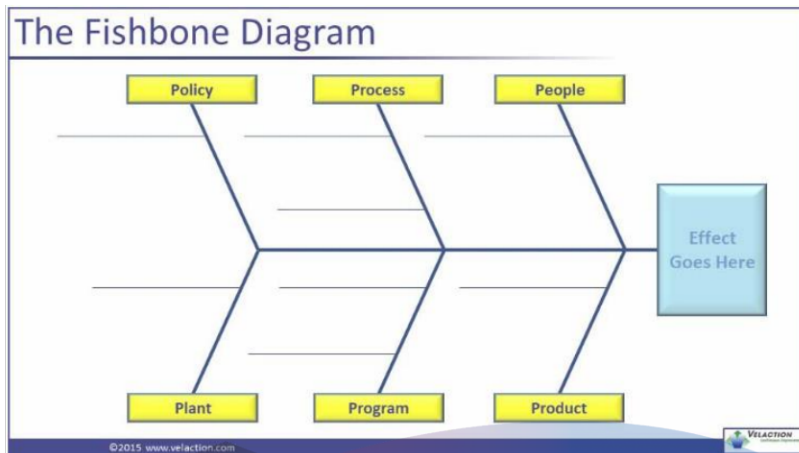
Langkah-langkah dalam membuat *fishbone diagram* yaitu [17]:

1. Identifikasi masalah
2. Deskripsi masalah di sisi kanan diagram. Ini mungkin masalahnya itu sendiri atau suatu gejala, tetapi pada tahap ini tidak ada yang tahu pasti.
3. Menggambar garis horizontal yang mengarah ke kotak yang menggambarkan masalah. Panah ini akan berfungsi sebagai tulang punggung, ini adalah titik awal dalam mengidentifikasi dan mengelompokkan penyebab utama dan penyebab.
4. Mengidentifikasi penyebab potensial dengan mengelompokkannya ke dalam beberapa kategori utamanya adalah orang, proses, material, peralatan, dan lingkungan. Kategori utama diidentifikasi dengan menggunakan teknik curah pendapat, mencoba membuat daftar semuanya. Selama representasi grafis pada diagram, harus berhati-hati untuk membuat ruang yang cukup di antara kategori-kategori untuk kemudian menambahkan penyebab individual. Setiap kategori utama ini akan dieksplorasi secara rinci.
5. Melanjutkan kegiatan curah pendapat setiap kategori penyebab melalui analisis rinci untuk setiap kategori utama yang teridentifikasi, menulis penyebab rinci pada garis miring yang terhubung ke kategori utama.

Adapun manfaat dari *fishbone* yaitu membantu menentukan akar penyebab masalahnya, membantu menghasilkan ide untuk memecahkan masalah, membantu penyelidikan lebih lanjut atau penemuan fakta, menentukan tindakan yang menentukan cara membuat hasil yang diinginkan, mendiskusikan subjek dengan lengkap dan rapi, dan menciptakan ide-ide baru [17].

Keunggulan *fishbone* adalah dapat memperluas setiap masalah yang terjadi, dan setiap orang yang terlibat dapat memberikan saran yang dapat menimbulkan masalah. Dan kelemahan dari *fishbone* adalah prespektif berbasis alat dan disain akan membatasi kemampuan tim untuk mendeskripsikan masalah menggunakan metode "*level why*" yang dalam, kecuali kertas yang digunakan benar-benar cukup untuk memenuhi kebutuhan ini [17].

Berikut ini pada Gambar 2.2 ditunjukkan kerangka dari *Fishbone Diagram* dengan model 6P [19].



Gambar 2.2 *Fishbone Diagram*

Mekanisme penggunaan diagram *Fishbone* dilakukan untuk menganalisis faktor-faktor yang memengaruhi rencana rancangan sistem informasi usulan pada sekolah Perguruan Kristen Hosana. Selanjutnya akan ditarik kesimpulan dan rangkuman untuk memudahkan pembahasan. Pendekatan diagram *Fishbone Analysis* yang digunakan dalam penelitian ini menggunakan metode pendekatan 6P yaitu identifikasi masalah menggunakan kategori *Policy*, *Process*, *People*, *Plant*, *Program*, dan *Product* sebagai kategori identifikasi masalah.

2.3.2 *Data Flow Diagram (DFD)*





Data Flow Diagram (DFD) adalah suatu bentuk model yang memungkinkan sistem analisa menggambarkan suatu sistem sebagai suatu jaringan proses dan fungsi yang dihubungkan satu dengan yang lain oleh suatu penghubung. DFD digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa memperhatikan lingkungan fisik dimana data tersebut mengalir atau disimpan [20].

Data Flow Diagram (DFD) adalah menggambarkan suatu pandangan mengenai suatu masukan, proses, dan keluaran dari sistem. Penganalisis sistem mencoba untuk memahami kebutuhan informasi pengguna, yaitu harus mampu mengkonseptualisasikan bagaimana data bergerak, transformasi yang dialami data dan apa hasilnya [15].

Terdapat dua teknik dasar penggambaran simbol DFD yang umum dipakai dan pada penelitian ini yang digunakan adalah model Gane dan Sarson. Yourdon dan De Marco menggunakan lambang lingkaran untuk menggambarkan *process* dan menggunakan lambang garis sejajar untuk menggambarkan *data store*. Sedangkan untuk simbol *external entity* dan simbol *data flow* kedua teknik tersebut menggunakan lambang yang sama yaitu: segi empat untuk

melambangkan *external entity* dan anak panah untuk melambangkan *data flow*. Berikut ini pada Tabel 2.1 ditunjukkan simbol dari DFD [21].

Tabel 2.1 Simbol DFD Gane dan Sarson

Keterangan	Gane dan Sarson Simbol
<i>External Entity</i>	
<i>Process</i>	
<i>Data Store</i>	
<i>Data Flow</i>	

Berikut ini, akan diuraikan fungsi dari masing-masing simbol-simbol yang terdapat pada DFD antara lain [20]:

1. Simbol Terminator atau *External Entity*

Terminator atau *External Entity* adalah bagian sistem yang dapat men-*supply input* ke dalam sistem dan menggunakan *output* dari sistem. *External Entity* dapat berupa orang atau sekelompok orang atau bagian/departemen/pemakai sistem.

2. Simbol Proses

Menggambarkan bagaimana suatu ditransformasikan menjadi output. Proses menggambarkan apa yang dilakukan sistem.

3. *Data Store* atau Penyimpanan Data

Menggambarkan media penyimpanan data dalam sistem yang berisi data yang digunakan sistem, digunakan sebagai sarana mengumpulkan dan membaca data. *Data store* dapat merupakan *file/database*, arsip/catatan manual, lemari/kotak, agenda/buku.

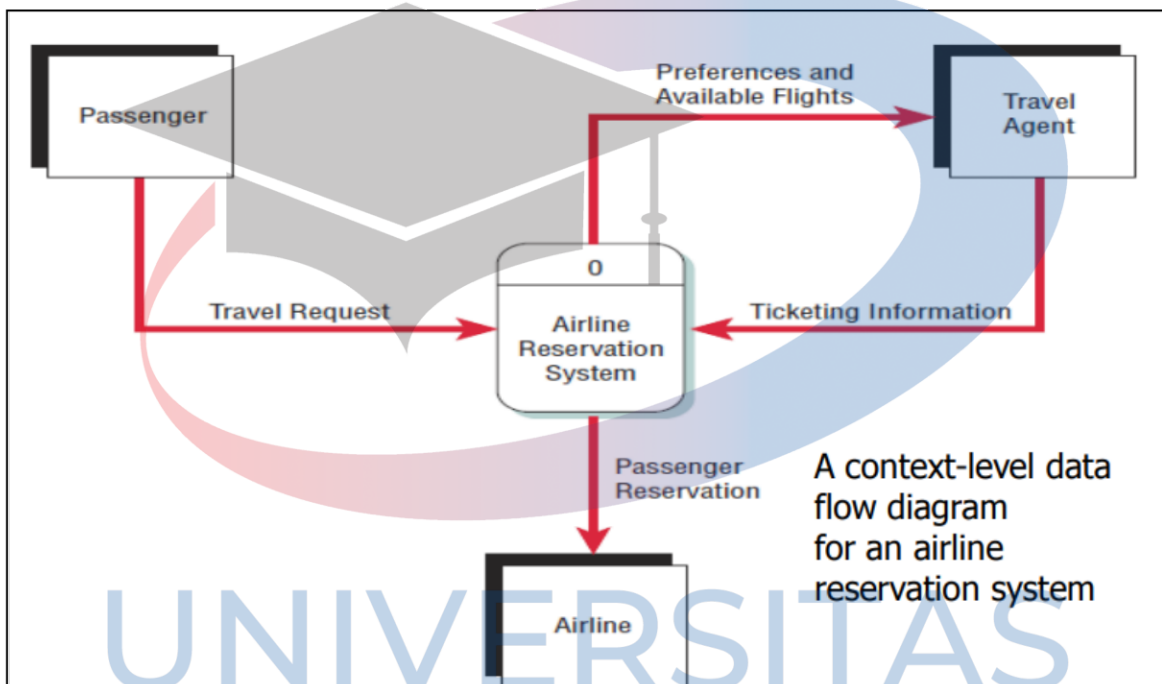
4. *Data Flow* atau Alur Data

Menggambarkan serangkaian paket data/informasi.

Terdapat beberapa jenis-jenis dari DFD yang biasa digunakan dalam perancangan sistem yaitu [20]:

1. Context Diagram

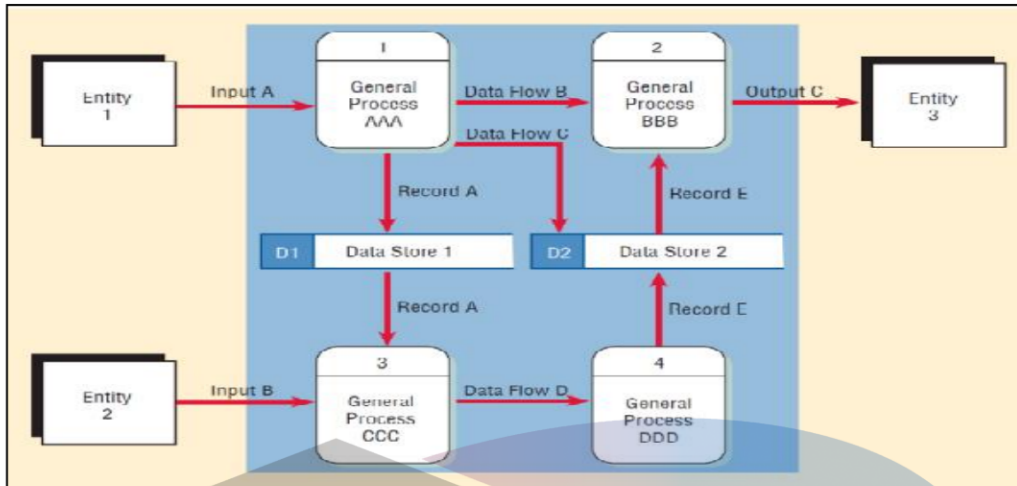
Context Diagram adalah diagram yang menggambarkan bagaimana proses dokumentasi data. *Context Diagram* terdiri atas sebuah lingkaran proses transformasi, *data sources*, dan *data destination* yang menerima maupun mengirim data secara langsung dari proses transformasi. Tujuan utama dari pembuatan *Context Diagram* adalah untuk menjadi jembatan komunikasi dengan *stakeholders*, tidak untuk menjelaskan *data flows* kepada *developers*.



Gambar 2.3 Contoh *Context Diagram* Versi Gane dan Sarson

Pada Gambar 2.3 merupakan sebuah contoh dari *Context Diagram* versi Gane dan Sarson dari sistem pemesanan tiket pesawat dimana terdapat 3 *external entity* yaitu penumpang, agen *travel*, dan maskapai penerbangan. Agen *travel* akan memberikan alur data keluaran yang masuk sistem berupa informasi tiket pesawat serta mendapatkan alur data masukan dari sistem berupa informasi preferensi dan penerbangan yang tersedia dari sistem. Selanjutnya, penumpang akan memberikan alur data keluaran yang masuk ke sistem berupa *request* pemesanan tiket. Di sisi maskapai penerbangan mendapatkan alur data masukan dari sistem berupa reservasi dari penumpang.

2. DFD Level 0

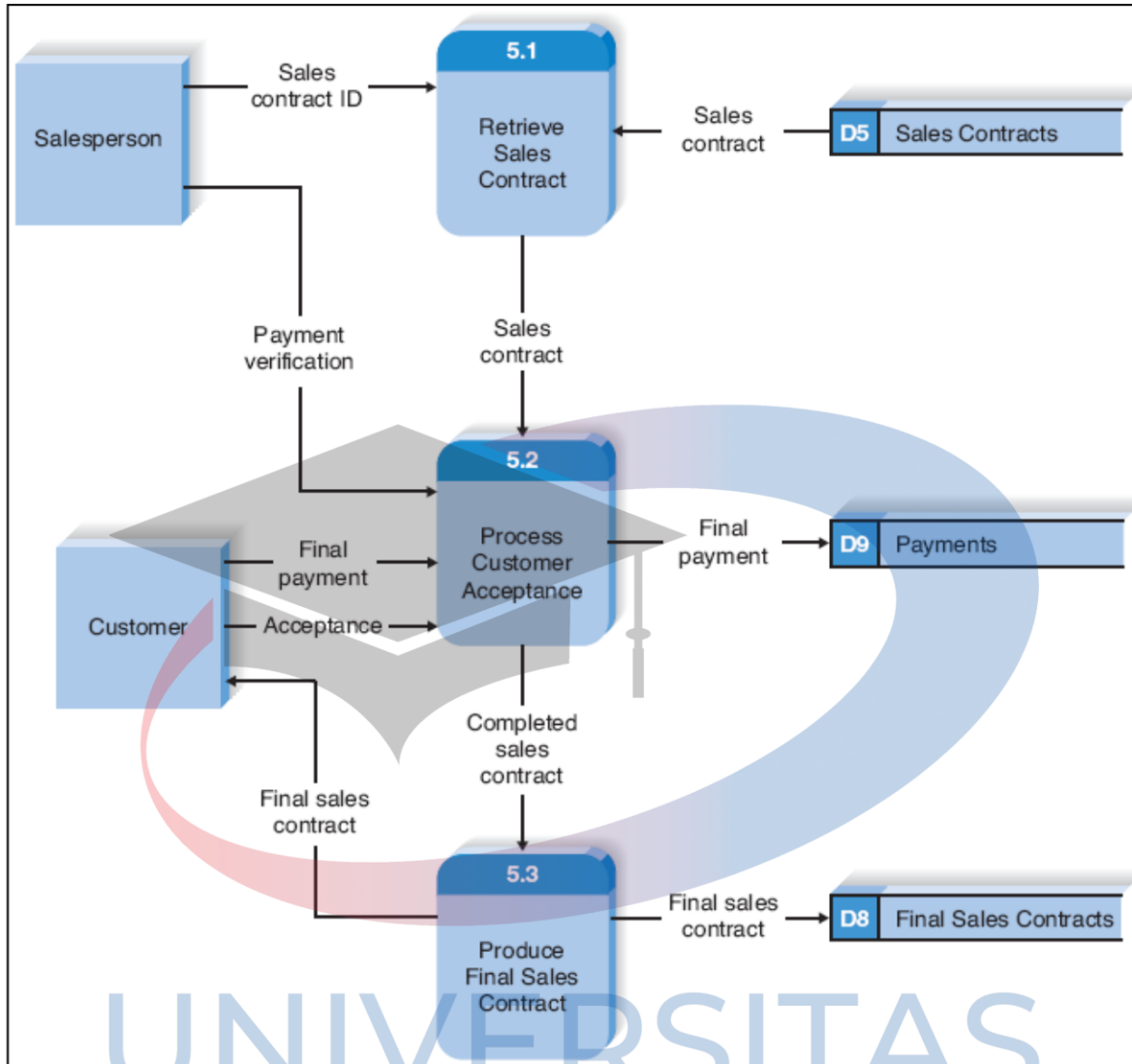


Gambar 2.4 Contoh DFD Level 0 Versi Gane dan Sarson

Pada Gambar 2.4 merupakan sebuah contoh dari DFD Level 0 yang menunjukkan secara detail alur keluar masuknya data secara lebih detail dan data *store* yang digunakan.

3. DFD Level 1,2, dan seterusnya.

UNIVERSITAS MIKROSKIL



Gambar 2.5 Contoh DFD Level 1 Versi Gane dan Sarson

Pada Gambar 2.5 merupakan sebuah contoh dari DFD Level 1 yang menunjukkan secara detail alur keluar masuknya data secara lebih detail dan data *store* yang digunakan. Pada DFD Level 1, gambaran alur data lebih jelas mencakup hingga kolom data yang dikirim dan akan tersimpan pada basis data sistem.

Terdapat beberapa langkah-langkah dalam proses penggambaran sebuah DFD antara lain [20][21]:

1. Identifikasi *external entity*.
2. Identifikasi semua *input* dan *output* yang terlibat dengan kesatuan luar.

3. Gambarlah terlebih dahulu suatu Diagram Konteks (*Context Diagram*)=*top level* dimana *Context Diagram* selalu mengandung satu dan hanya satu proses saja.
4. Gambarlah bagan berjenjang (*hierarchy chart*) untuk mempersiapkan penggambaran DFD ke level-level lebih bawah lagi.
5. Gambarlah sketsa DFD untuk *overview diagram* (level 0).
6. Gambarlah DFD untuk level-level berikutnya (1,2, dst).

Pada penelitian ini, jenis DFD yang akan digunakan adalah DFD versi Gane dan Sarson.

2.3.3 Kamus Data

Kamus data atau *data dictionary* adalah suatu daftar data elemen yang terorganisir dengan definisi yang tetap dan sesuai dengan sistem sehingga *user* dan analis sistem mempunyai pengertian yang sama tentang *input*, *output*, dan komponen *data store*. Kamus data ini sangat membantu analis sistem dalam mendefinisikan data yang mengalir di dalam sistem, sehingga pendefinisian data itu dapat dilakukan dengan lengkap dan terstruktur [15].

Kamus Data (KD) atau *Data Dictionary* (DD) atau disebut juga dengan istilah *System Data Dictionary*, digunakan untuk memperjelas aliran data yang digambarkan pada *Data Flow Diagram* (DFD). Kamus Data (KD) adalah kumpulan daftar elemen data yang mengalir pada sistem perangkat lunak sehingga masukan (*input*) dan keluaran (*output*) dapat dipahami secara umum (memiliki standar cara penulisan). Kamus data dalam implementasi program dapat menjadi parameter masukan atau keluaran dari sebuah fungsi atau prosedur [22].

Kamus data adalah aplikasi khusus dari jenis-jenis kamus yang digunakan sebagai referensi dalam kehidupan sehari-hari. Kamus data adalah karya referensi data tentang data (yaitu metadata), satu yang disusun oleh analis sistem untuk memandu mereka melalui analisis dan desain. Sebagai dokumen, kamus data mengumpulkan dan mengordinasikan istilah-istilah data tertentu, dan itu menegaskan apa setiap istilah berarti bagi orang yang berbeda dalam organisasi. Salah satu alasan penting untuk memelihara kamus data adalah untuk menjaga data tetap bersih [15].

Selain menyediakan dokumentasi dan menghilangkan redundansi, kamus data dapat digunakan untuk [15]:

1. Validasi diagram aliran data untuk kelengkapan dan akurasi.
2. Berikan titik awal untuk mengembangkan layar dan laporan.

3. Menentukan isi data yang disimpan dalam file.
4. Mengembangkan logika untuk proses diagram aliran data.
5. Buat *Extensible Markup Language* (XML).

Tabel 2.2 Simbol Kamus Data

No	Simbol	Uraian
1	=	Terdiri dari
2	+	Dan
3	{ }	Elemen-elemen repetitif (kelompok berulang)
4	[]	Salah satu dari dua situasi tertentu
5	()	Pilihan (boleh dikosongkan)
6		Pemisah sejumlah alternatif pilihan antara simbol []

Hal-hal yang harus dimuat didalam kamus data adalah sebagai berikut [15]:

1. Nama arus data yaitu kamus data dibuat berdasarkan arus data yang mengalir di DAD (Diagram Alir Data), serta nama arus data juga harus dicatat di dalam kamus data.
2. Alias, perlu ditulis karena data yang sama mempunyai nama yang berbeda untuk orang atau departemen satu dengan yang lainnya.
3. Tipe data atau bentuk data, digunakan untuk mengelompokkan kamus data ke dalam kegunaannya sewaktu perancangan sistem.
4. Arus data, menunjukkan dari mana data mengalir dan kemana data menuju keterangan, arus data ini perlu dicatat dikamus data untuk memudahkan mencari arus data di DAD (Diagram Alir Data).
5. Penjelasan, dapat diisi dengan keterangan-keterangan tentang arus data tersebut.

Berikut ini ditunjukkan contoh dari pembuatan kamus data pada data pelanggan dan data penjualan antara lain [15]:

1. Data Pelanggan = Nomor Pelanggan + Nama Pelanggan + Alamat + [Status Pelanggan]
 Status Pelanggan = [Aktif | Tidak Aktif]

Seperti dapat dilihat kamus data pelanggan terdapat simbol [] pada status pelanggan yang artinya kamus data status pelanggan merupakan pilihan yang wajib dipilih salah satu aktif atau tidak aktif.

2. Data Penjualan = Nomor Invoice + Kode Pelanggan + Nama Pelanggan + {Kode Produk + Nama Produk + Kuantiti + Harga + Subtotal} + Total Penjualan

Seperti dapat dilihat kamus data penjualan terdapat simbol {} pada status kode produk, nama produk, kuantiti, harga, dan subtotal yang artinya kamus data tersebut merupakan perulangan dikarenakan dalam data penjualan pasti terdapat isi penjualan yang berupa produk-produk yang dijual kepada pelanggan.

2.3.4 Normalisasi

Normalisasi adalah proses dimana dapat mendekomposisi atau membagi relasi menjadi lebih dari satu relasi untuk menghilangkan anomali dalam database relasional. Ini adalah proses langkah demi langkah dan setiap langkah dikenal sebagai bentuk normal [23].

Salah satu alasan utama normalisasi adalah untuk mengatur data sehingga dapat mengurangi data yang berlebihan. Tujuan normalisasi adalah mengkonstruksi relasi tanpa redundansi. Untuk melakukan ini diperlukan pendefinisian dalam terminologi relasi normal. Pada proses normalisasi dibagi dalam tiga tahap, yaitu tahap normalisasi pertama, tahap normalisasi kedua, tahap normalisasi ketiga [15].

Istilah penting dalam teknik normalisasi [23].

1. *Field*/atribut kunci

Setiap *file* selalu terdapat kunci dari file berupa satu field atau satu field yang dapat mewakili record.

2. *Candidate key*

Kumpulan atribut minimal yang secara unik mengidentifikasi sebuah baris yang fungsinya sebagai calon *primary key*.

3. *Composite key*

Kunci kandidat yang berisi lebih dari satu atribut.

4. *Primary key*

Candidate key yang dipilih untuk mengidentifikasi baris secara unik.

5. *Alternate key*

Candidate key yang tidak dipilih sebagai *primary key*.

6. Foreign key

Kunci di tabel lain yang terhubung dengan *primary key* pada sebuah tabel.

Ketergantungan (*dependency*) merupakan konsep yang mendasari normalisasi. Dalam basis data *dependency* lebih sering disebut *Functional Dependency* atau Ketergantungan Fungsional yang digunakan untuk menggambarkan hubungan, batasan, keterkaitan antara atribut-atribut dalam relasi. Atau lebih jelasnya nilai dari suatu atribut dapat menentukan nilai dari atribut yang lain. *Dependency* akan mencari acuan untuk pendekomposisi data ke dalam bentuk yang paling efisien. Sebagai contoh yaitu untuk NIM dan Nama_Mhs. NIM secara fungsional menentukan Nama_Mhs, karena untuk setiap NIM yang sama maka nilai Nama_Mhs nya sama. *Dependency* dapat dibagi ke dalam tiga jenis, yaitu [23]:

1. Full Dependency (Ketergantungan Penuh)

Menunjukkan jika terdapat atribut A dan B dalam suatu relasi, dimana B memiliki ketergantungan fungsional secara penuh pada A, tapi B tidak memiliki ketergantungan terhadap subset A. Contoh *Full Dependency* dapat dilihat pada Tabel 2.3.

Tabel 2.3 Contoh *Full Dependency*

KodeBarang	NamaBarang	NoFaktur	Kts
B001	Sabun	F001	3
B002	Shampoo	F002	5
B003	Pasta Gigi	F003	16

KodeBarang → NamaBarang

KodeBarang, NoFaktur → Kts (Kuantitas)

Untuk setiap nilai KodeBarang yang sama, maka nilai NamaBarang juga pasti sama. Begitu juga untuk setiap nilai KodeBarang yang terdapat pada NoFaktur yang sama, maka nilai Kuantitasnya juga sama.

2. Partial Dependency (Ketergantungan Sebagian)

Ketergantungan parsial atau sebagian memiliki 2 atribut dari A untuk menentukan B, namun untuk menentukan B tidak harus 2 atribut artinya jika salah satu atribut A yang menentukan B dapat dihilangkan namun tidak merubah arti relasi dan masih tetap berelasi ketergantungan. Contoh *Partial Dependency* dapat dilihat pada Tabel 2.4.

Tabel 2.4 Contoh *Partial Dependency*

No Faktur	Kode Barang	Nama Barang	Kode Pelanggan	Nama Pelanggan	Kts
F001	B001	Sabun	P001	Selly	3
F002	B002	Shampoo	P002	Andi	5
F003	B003	Pasta Gigi	P003	Citra	16

NamaBarang, KodePelanggan dan NamaPelanggan hanya bergantung pada field KodeBarang, bukan pada NoFaktur.

3. *Transitive Dependency* (Ketergantungan Transitif)

Transitive dependency biasanya terjadi pada tabel hasil relasi, atau kondisi dimana terdapat tiga atribut A,B,C. Kondisinya adalah A tergantung terhadap B dan B tergantung terhadap C. Maka C dikatakan sebagai *transitive dependency* terhadap A melalui B. Contoh *Transitive Dependency* dapat dilihat pada Tabel 2.5.

Tabel 2.5 Contoh *Transitive Dependency*

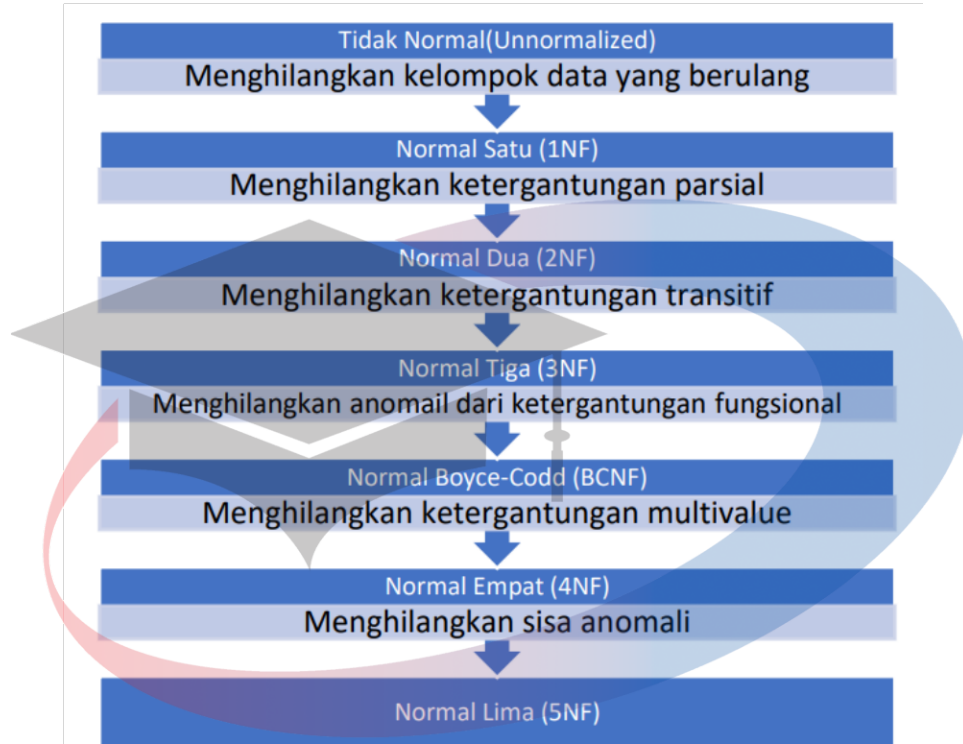
KodeBarang	NamaBarang	KodeKategori	Kategori
B001	Sabun	K001	Kamar Mandi
B004	Panci	K002	Alat Dapur
B005	Boneka	K003	Mainan

Terdapat beberapa manfaat dilakukannya normalisasi pada rancangan basis data sistem usulan yaitu [23]:

1. Menghasilkan tabel yang lebih kecil dengan baris yang lebih kecil.
2. Pencarian, pengurutan, dan membuat indeks lebih cepat, karena tabel lebih sempit, dan lebih banyak baris cocok pada halaman data.
3. Menghasilkan lebih banyak tabel dengan memecah tabel asli. Dengan demikian bisa ada lebih banyak indeks berkerumun dan karenanya ada lebih banyak fleksibilitas dalam menyetel kueri.
4. Pencarian indeks umumnya lebih cepat karena indeks cenderung lebih sempit dan lebih pendek.
5. Semakin banyak tabel memungkinkan penggunaan segmen yang lebih baik untuk mengontrol penempatan fisik data.
6. Ada lebih sedikit indeks per tabel dan karenanya perintah modifikasi data lebih cepat.
7. Ada sejumlah kecil nilai nol dan redundansi. Ini membuat basis data lebih kompak.
8. Anomali modifikasi data berkurang.

- Secara konseptual lebih bersih dan lebih mudah untuk dipertahankan dan diubah seiring perubahan kebutuhan.

Selanjutnya, ditunjukkan tahap-tahapan yang dilakukan dalam melakukan normalisasi basis seperti yang ditunjukkan Gambar 2.6 [23].



Gambar 2.6 Tahapan Normalisasi Basis Data

- Tidak Normal (*Unnormalized*)

Pada tahap ini, diambil seluruh data yang ada dan diperlukan dalam *database* itu sendiri.

Misalnya pada contoh tabel nilai mahasiswa, datanya terdiri dari NIM, nama mahasiswa, alamat mahasiswa, ID Dosen, nama dosen, status dosen, nilai basis data, nilai pemrograman, nilai pengantar ilmu komputer, nilai interaksi manusia dan komputer.

Tabel 2.6 Tabel Mentah

NIM	Nama_Mhs	Alamat_Mhs	ID_Dosen	Nama_Dosen	Status_Dosen	Basis Data	Pemrograman	PIK	IMK
072134321	Andi	Medan	10010	Raissa	Tetap	78		92	
072120089	Budi	Medan	10087	Andini	Honoror		85	66	78
072129262	Iwan	Binjai	10091	Ulfayani	Tetap	56	83		
087201231	Sari	Medan	10091	Ulfayani	Tetap		96		83
082324321	Putri	Medan	10003	Laylan	Honoror	89			77
079909032	Melisa	Tj.Morawa	10003	Laylan	Honoror	50		41	

Tabel mentah di atas memerlukan suatu analisa dikarenakan tabel yang ada di soal mempunyai suatu keanehan. Perhatikan pada kolom Basis Data, Pemrograman, PIK, dan IMK. Setiap

nama kolom yang sudah disebutkan memiliki kode tersendiri yang bisa merepresentasikan nama dari mata kuliah. Dengan begitu bisa dikumpulkan kode mata kuliah menjadi satu kolom baru yang bisa diberi nama Kode_Matkul. Kemudian nama dari mata kuliah juga bisa dibuat menjadi satu kolom baru yang diberi nama Nama_Matkul sehingga bentuk tabel seperti yang terlihat pada Tabel 2.7 berikut.

Tabel 2.7 Tabel *Unnormalized*

NIM	Nama_Mhs	Alamat_Mhs	ID_Dosen	Nama_Dosen	Status_Dosen	Kode_Matkul	Nama_Matkul	Nilai
072134321	Andi	Medan	10010	Raissa	Tetap	MK001	Basis Data	78
						MK003	PIK	92
072120089	Budi	Medan	10087	Andini	Honoror	MK002	Pemrograman	85
						MK003	PIK	66
						MK004	IMK	78
072129262	Iwan	Binjai	10091	Ulfayani	Tetap	MK001	Basis Data	56
						MK002	Pemrograman	83
087201231	Sari	Medan	10091	Ulfayani	Tetap	MK002	Pemrograman	96
						MK004	IMK	83
082324321	Putri	Medan	10003	Laylan	Honoror	MK001	Basis Data	89
						MK004	IMK	77
079909032	Melisa	Tj.Morawa	10003	Laylan	Honoror	MK001	Basis Data	50
						MK003	PIK	41

2. Normal Satu (1NF)

Pada tahap ini, bagi seluruh data yang diperlukan menjadi beberapa bagian berdasarkan jenis data tersebut. Sebuah tabel bisa dikategorikan sebagai tabel 1NF jika di setiap baris *record* hanya memiliki 1 *value*. Hasil transformasi dari *Unnormalized Form* ke 1NF bisa dilihat pada Tabel 2.8 berikut.

Tabel 2.8 Tabel Normal 1 (1NF)

NIM	Nama_Mhs	Alamat_Mhs	ID_Dosen	Nama_Dosen	Status_Dosen	Kode_Matkul	Nama_Matkul	Nilai
072134321	Andi	Medan	10010	Raissa	Tetap	MK001	Basis Data	78
072134321	Andi	Medan	10010	Raissa	Tetap	MK003	PIK	92
072120089	Budi	Medan	10087	Andini	Honoror	MK002	Pemrograman	85
072120089	Budi	Medan	10087	Andini	Honoror	MK003	PIK	66
072120089	Budi	Medan	10087	Andini	Honoror	MK004	IMK	78
072129262	Iwan	Binjai	10091	Ulfayani	Tetap	MK001	Basis Data	56
072129262	Iwan	Binjai	10091	Ulfayani	Tetap	MK002	Pemrograman	83
087201231	Sari	Medan	10091	Ulfayani	Tetap	MK002	Pemrograman	96
087201231	Sari	Medan	10091	Ulfayani	Tetap	MK004	IMK	83
082324321	Putri	Medan	10003	Laylan	Honoror	MK001	Basis Data	89
082324321	Putri	Medan	10003	Laylan	Honoror	MK004	IMK	77
079909032	Melisa	Tj.Morawa	10003	Laylan	Honoror	MK001	Basis Data	50
079909032	Melisa	Tj.Morawa	10003	Laylan	Honoror	MK003	PIK	41

3. Normal Dua (2NF)

Pada tahap ini, bagi berdasarkan jenis dan memberikan *primary key* pada masing-masing tabel. Sebuah tabel bisa dikategorikan sebagai tabel 2NF jika tabel sudah dalam keadaan 1NF dan setiap kolom didalam tabel itu *Functional Dependency* kepada semua *key* (*Full Dependency*), bukan hanya kepada salah satu *key* (*Partial Dependency*). Berarti harus ditentukan terlebih dahulu *key* yang ada di dalam tabel. Dari tabel di atas bisa ditentukan bahwa ada dua *candidate key* yang tersedia dalam tabel diatas, yaitu NIM + Kode_Matkul, dan NIM + Nama_Matkul. Setelah dipilih dari kedua *candidate key*, maka yang paling efisien untuk menjadi *Primary Key* adalah NIM + Kode_Matkul. Setelah itu, tentukan *Functional Dependency* yang ada di dalam tabel seperti yang ditunjukkan pada Tabel 2.9.

Tabel 2.9 *Functional Dependency*

Full Dependency	NIM + Kode_Matkul	Nama_Matkul Nilai_Matkul
Partial Dependency	NIM	Nama_Mhs Alamat_Mhs ID_Dosen Nama_Dosen Status_Dosen
Partial Dependency	Kode_Matkul	Nama_Matkul
Transitive Dependency	ID_Dosen	Nama_Dosen Status_Dosen

Setelah mendapatkan semua *Functional Dependency* di dalam tabel, maka untuk mengubah tabel 1NF menjadi 2NF harus dipindahkan kolom yang bergantung hanya kepada salah satu dari *key* (*Partial Dependency*) ke dalam tabel baru sehingga saat ini tabel terpisah seperti terlihat pada Tabel 2.10 hingga 2.12 berikut.

Tabel 2.10 Tabel Mahasiswa 2NF

NIM	Nama_Mhs	Alamat_Mhs	ID_Dosen	Nama_Dosen	Status_Dosen
072134321	Andi	Medan	10010	Raissa	Tetap
072120089	Budi	Medan	10087	Andini	Honorar
072129262	Iwan	Binjai	10091	Ulfayani	Tetap
087201231	Sari	Medan	10091	Ulfayani	Tetap
082324321	Putri	Medan	10003	Laylan	Honorar
079909032	Melisa	Tj.Morawa	10003	Laylan	Honorar

Tabel 2.11 Tabel Mata Kuliah 2NF

Kode_Matkul	Nama_Matkul
MK001	Basis Data
MK002	Pemrograman
MK003	PIK
MK004	IMK

Tabel 2.12 Tabel Nilai 2NF

NIM	Kode_Matkul	Nilai
072134321	MK001	78
072134321	MK003	92
072120089	MK002	85
072120089	MK003	66
072120089	MK004	78
072129262	MK001	56
072129262	MK002	83
087201231	MK002	96
087201231	MK004	83
082324321	MK001	89
082324321	MK004	77
079909032	MK001	50
079909032	MK003	41

4. Normal Tiga (3NF)

Pada tahap ini, bagi menjadi lebih terperinci untuk menghindari terjadinya redundansi. Sebuah tabel bisa dikategorikan sebagai tabel 3NF jika tabel sudah dalam 2NF dan setiap kolom yang bukan *key* harus *Functional Dependency* dengan *primary key* nya. Bisa dilihat di dalam Tabel Nilai dan Tabel Mata Kuliah sudah memenuhi persyaratan 3NF. Tetapi didalam Tabel Mahasiswa terdapat kolom yang bergantung bukan kepada *key* nya (*Transitive Dependency*) yaitu Nama_Dosen dan Status_Dosen. Ini menunjukkan bahwa Tabel Mahasiswa belum memenuhi persyaratan menjadi 3NF. Nama_Dosen dan Status_Dosen itu *Functional Dependency* kepada ID_Dosen yang diketahui bahwa ID_Dosen itu bukanlah sebuah *primary key*. Maka untuk membuat Tabel Nilai kedalam bentuk 3NF, harus ditempatkan kolom yang

Functional Dependency bukan kepada *key* nya (*Transitive Dependency*) kedalam tabel yang baru sehingga bentuk tabel menjadi seperti Tabel 2.13 hingga 2.16 berikut.

Tabel 2.13 Tabel Mahasiswa 2NF

NIM	Nama_Mhs	Alamat_Mhs
072134321	Andi	Medan
072120089	Budi	Medan
072129262	Iwan	Binjai
087201231	Sari	Medan
082324321	Putri	Medan
079909032	Melisa	Tj.Morawa

Tabel 2.14 Tabel Dosen 2NF

ID_Dosen	Nama_Dosen	Status_Dosen
10010	Raissa	Tetap
10087	Andini	Honorar
10091	Ulfayani	Tetap
10003	Laylan	Honorar

Tabel 2.15 Tabel Mata Kuliah 2NF

Kode_Matkul	Nama_Matkul
MK001	Basis Data
MK002	Pemrograman
MK003	PIK
MK004	IMK

Tabel 2.16 Tabel Nilai 2NF

NIM	Kode_Matkul	Nilai
072134321	MK001	78
072134321	MK003	92
072120089	MK002	85
072120089	MK003	66
072120089	MK004	78
072129262	MK001	56
072129262	MK002	83
087201231	MK002	96
087201231	MK004	83
082324321	MK001	89
082324321	MK004	77
079909032	MK001	50
079909032	MK003	41

2.3.5 Basis Data

Basis data tidak hanya merupakan kumpulan *file*. Lebih dari itu, basis data adalah pusat sumber data yang caranya dipakai oleh banyak pemakai untuk berbagai aplikasi. Inti dari basis data adalah *Database Management (DBMS)*, yang membolehkan pembuatan, modifikasi dan pembaharuan basis data, mendapatkan kembali data, dan membangkitkan laporan [15].

Basis data merupakan kumpulan informasi yang ada selama periode waktu yang lama, seringkali bertahun-tahun. Basis data merupakan hal yang sangat penting untuk semua bisnis. Basis data berada di belakang layar perusahaan besar maupun perusahaan kecil. Perusahaan menyimpan setiap data – data penting mereka ke dalam basis data. Kekuatan basis data berasal dari pengetahuan dan teknologi yang telah berkembang dan diwujudkan dalam perangkat lunak khusus yang disebut sistem manajemen basis data, atau *Database Management System (DBMS)*, atau disebut juga Sistem Basis Data [23].

DBMS adalah alat yang ampuh untuk membuat dan mengelola jumlah data yang besar secara efisien dan memungkinkannya bertahan dalam jangka waktu yang lama dengan aman. DBMS adalah kumpulan data yang saling terkait dan seperangkat program untuk mengakses data tersebut. Pengumpulan data, biasanya disebut sebagai basis data, berisi informasi yang relevan dengan suatu perusahaan. Tujuan utama DBMS adalah menyediakan cara untuk menyimpan dan mengambil informasi basis data yang nyaman dan efisien. Sistem basis data dirancang untuk mengelola banyak informasi. Manajemen data melibatkan struktur pendefinisian untuk penyimpanan informasi dan menyediakan mekanisme untuk manipulasi informasi. Selain itu,

sistem basis data harus memastikan keamanan informasi yang disimpan, meskipun sistem macet atau upaya akses yang tidak sah. Jika data akan dibagikan di antara beberapa pengguna, sistem harus menghindari kemungkinan hasil yang tidak normal. DBMS diharapkan untuk [23].

Tujuan efektivitas *database* meliputi hal-hal berikut [15]:

1. Memastikan bahwa data dapat dipakai di antara pemakai untuk berbagai aplikasi.
2. Memelihara data baik keakuratan maupun kekonsistennannya.
3. Memastikan bahwa semua data yang diperlukan untuk aplikasi sekarang dan yang akan datang akan disediakan dengan cepat.
4. Memperoleh basis data untuk berkembang dan kebutuhan pemakai untuk berkembang.

Tujuan yang telah disebut diatas meningkatkan akan keuntungan dan kerugian pendekatan basis data. Pertama, pemakaian data berarti data perlu disimpan hanya sekali. Membantu mencapai integritas data, karena mengubah data diselesaikan lebih mudah dan dapat dipercaya jika data muncul hanya sekali dalam banyak *file* berbeda [15].

Ketika pemakai memerlukan data khusus, basis data yang dirancang dengan baik (*well-designed*) memenuhi lebih dahulu kebutuhan data yang demikian (atau mungkin telah digunakan untuk aplikasi lain). Akibatnya, data memiliki kesempatan tersedia yang lebih baik dalam basis data dari pada dalam sistem *file* yang konvensional. Basis data yang dirancang dengan baik juga lebih fleksibel dari pada *file* terpisah. Oleh karena itu, basis data dapat berkembang seperti pada perubahan kebutuhan pemakai dan aplikasinya [15].

Akhirnya, pendekatan basis data memiliki keuntungan yang memperoleh pemakai untuk memiliki pandangan sendiri mengenai data. Pemakai tidak perlu memperhatikan struktur sebenarnya basis data atau penyimpanan fisiknya [15].

Kerugian pertama pendekatan basis data adalah bahwa semua data disimpan dalam satu tempat. Oleh karena itu, data lebih mudah diserang bencana dan membutuhkan *back-up* yang lengkap. Terdapat resiko bahwa administrator basis data menjadi satu dan satunya orang yang mempunyai hal istimewa atau kemampuan cukup untuk mendekati data. Prosedur birokratis perlu memodifikasi atau memperbaharui basis data secara lengkap terlihat tidak dapat diatasi [15].

2.1.4 Pendidikan Sekolah

Pendidikan sekolah adalah suatu proses pembelajaran yang sistematis dan terstruktur yang dilaksanakan di lembaga pendidikan formal seperti sekolah. Pendidikan sekolah melibatkan

pengajaran dan pembelajaran di bawah bimbingan guru atau pendidik yang memiliki tujuan tertentu untuk mengembangkan pengetahuan, keterampilan, sikap, dan nilai-nilai pada peserta didik. Pendidikan sekolah merupakan lembaga pendidikan formal. Sedangkan, pendidikan luar sekolah mengambil bentuk dalam pendidikan informal dan pendidikan non formal [24].

1. Pendidikan formal yaitu pendidikan yang dasar, isi, metode, dan alat-alatnya disusun secara eksplisit, sistematis, dan distandarisasikan. Pendidikan formal ini diselenggarakan di sekolah-sekolah atau lembaga pendidikan yang bersifat akademis. Adapun usia peserta didik di suatu jenjang relatif homogen, khususnya pada jenjang-jenjang permulaan. Pendidikan formal diselenggarakan mulai dari taman kanak-kanak hingga tingkat perguruan tinggi.
2. Pendidikan informal yaitu pendidikan yang diperoleh seseorang dari pengalaman sehari-hari dengan sadar atau tidak sadar, sejak seseorang lahir sampai meninggal dunia baik dalam keluarga, dalam pekerjaan atau pengalaman sehari-hari.

2.1.5 Administrasi Keuangan Sekolah

Administrasi keuangan adalah upaya pengelolaan mencakup semua aktivitas yang berhubungan erat dengan semua sistem keuangan untuk mencapai tujuan tiap perusahaan atau organisasi. Jika dikaitkan ke dalam administrasi keuangan sekolah dapat diartikan sebagai suatu proses pencatatan dan pengendalian keuangan milik sekolah yang dilaksanakan secara bertanggungjawab, jujur, terbuka, tertib, cermat, efektif, efisien sehingga terarah pada pencapaian tujuan sekolah secara optimal. Administrasi keuangan sekolah juga mencakup seluruh kegiatan yang berhubungan dengan proses penerimaan, penyimpanan, pengeluaran, dan pertanggungjawaban keuangan sekolah [25].

Terdapat beberapa jenis dari administrasi keuangan sekolah khususnya pada bagian pemasukan keuangan sekolah yaitu [25]:

1. Sumbangan Pembinaan Pendidikan (SPP) atau sering disebut uang sekolah digunakan untuk membantu pembinaan pendidikan yakni membantu penyelenggaraan sekolah, kesehatan personil, perbaikan sarana dan prasarana, dan kegiatan supervisi.
2. Pembayaran kebutuhan sekolah yang merujuk pada proses pembayaran berbagai biaya terkait pendidikan dan administrasi sekolah. Hal ini dapat mencakup berbagai jenis pembayaran yang harus diselesaikan oleh siswa atau orang tua/wali siswa sebagai kontribusi untuk pemeliharaan dan pengelolaan fasilitas sekolah, penyediaan layanan pendidikan, dan berbagai kegiatan

sekolah. Beberapa contoh pembayaran kebutuhan sekolah meliputi pembayaran buku dan alat tulis, seragam sekolah, biaya kegiatan ekstrakurikuler, biaya kegiatan dan acara sekolah, biaya transportasi atau catering.

3. Pembayaran biaya pendaftaran siswa baru adalah pembayaran yang biasanya harus dilakukan oleh orang tua atau wali siswa yang baru mendaftar ke sekolah. Biaya pendaftaran ini dirancang untuk menutupi sejumlah pengeluaran administratif dan proses penerimaan siswa baru.



UNIVERSITAS MIKROSKIL