

BAB II TINJAUAN PUSTAKA

2.1 Sistem Informasi

2.1.1 Definisi Sistem Informasi

Sistem informasi dikembangkan sesuai dengan kebutuhan pengguna. Dalam penerapannya, sebuah sistem informasi dapat berupa sebuah mainframe, sebuah *server* dari komputer biasa, maupun hosting di internet pada sebuah komputer *server*. Namun tetap saja ada kesamaan di antara ketiga penerapan berbeda ini. Kesamaan itu yaitu sama-sama menggunakan sarana jaringan komputer (*intranet* dan internet) untuk melakukan pemrosesan data secara bersama (terdistribusi), baik oleh beberapa pengguna maupun beberapa grup pengguna, menggunakan layanan/fitur/aplikasi yang disertakan.

Sebuah sistem informasi memiliki sejumlah komponen yang mempunyai fungsi dan tugas masing-masing yang saling berkaitan satu sama lain. Keterkaitan antar komponen ini membentuk suatu kesatuan kerja, yang menjadikan sistem informasi dapat mencapai tujuan dan fungsi yang ingin di capai oleh pengguna dan pengembang sistem informasi bersangkutan [1].

2.1.2 Komponen Sistem Informasi

Komponen-komponen yang terdapat di dalam semua jenis sistem informasi mencakup tujuh poin. Berikut ketujuh poin tersebut [1]:

1. *Input* (Masukan)

Komponen *input* ini berfungsi untuk menerima semua *input* (masukan) dari pengguna. *Input* yang diterima dalam bentuk data. Data ini berasal dari satu maupun beberapa buah sumber.

2. *Output* (Keluaran)

Komponen *output* berfungsi untuk menyajikan hasil akhir ke pengguna sistem informasi. Informasi yang disajikan ini merupakan hasil dari pengolahan data yang telah di-*input* sebelumnya. Pada komponen *output*, informasi yang disajikan disesuaikan dengan data yang diinput dan fungsionalitas dari sistem informasi bersangkutan.

3. *Software* (Perangkat Lunak)

Komponen perangkat lunak ini membantu sistem informasi di dalam menjalankan tugasnya dan untuk dapat dijalankan sebagaimana mestinya. Komponen perangkat lunak ini melakukan proses pengolahan data, penyajian informasi, penghitungan data, dan lain-lain.

4. *Hardware* (Perangkat Keras)

Komponen *hardware* mencakup semua perangkat keras yang digunakan secara fisik di dalam sistem informasi, baik di komputer *server* maupun komputer *client*.

5. *Database* (Basis Data)

Komponen basis data berfungsi untuk menyimpan semua data dan informasi ke dalam satu atau beberapa tabel. Setiap tabel memiliki *field* masing-masing, serta antar tabel dapat juga terjadi relasi (hubungan)

6. Kontrol dan Prosedur

Kontrol dan prosedur adalah dua komponen yang menjadi satu. Komponen kontrol berfungsi untuk mencegah terjadinya beragam gangguan dan ancaman terhadap data dan informasi yang ada di dalam sistem, termasuk juga sistem informasi itu sendiri beserta fisiknya. Komponen prosedur mencakup semua prosedur dan aturan yang harus dilakukan dan wajib ditaati bersama, guna mencapai tujuan yang diinginkan. Keterkaitan antara komponen kontrol dan prosedur dalam hal pencegahan terhadap kemungkinan ancaman dan gangguan yang terjadi pada sistem informasi, yang berpengaruh terhadap layanan yang diberikan, informasi yang disajikan, dan tingkat kepuasan pengguna.

7. Teknologi dan Jaringan Komputer

Komponen teknologi mengatur *software*, *hardware*, *database*, kontrol dan prosedur, *input*, dan *output*, sehingga sistem dapat berjalan dan terkendali dengan baik. Komponen jaringan komputer berperan di dalam menghubungkan sistem informasi dengan sebanyak mungkin pengguna, baik melalui kabel jaringan maupun tanpa kabel.

2.2 Administrasi

2.2.1 Pengertian Administrasi

Secara teoritis, pengertian administrasi adalah melayani secara intensif, sedangkan secara etimologis administrasi dalam bahasa Inggris “administer” yaitu kombinasi dari kata Latin yang terdiri dari *AD* dan *MINISTRARE* yang berarti “to serve” melayani, membantu dan memenuhi. Jadi, secara etimologis administrasi adalah melayani secara intensif. Kata “*administratio*” yang kemudian masuk ke dalam bahasa Inggris menjadi “*administration*” dalam bahasa Indonesia menjadi administrasi [2].

Administrasi akan berhasil baik apabila didasarkan atas dasar-dasar yang tepat. Dasar diartikan sebagai suatu kebenaran yang fundamental yang dapat dipergunakan sebagai landasan dan pedoman bertindak dalam kehidupan bermasyarakat. Berikut adalah dasar-dasar administrasi, antara lain [3]:

a. Prinsip efisiensi

Seorang administrator akan berhasil dalam tugasnya bilamana dia efisien dalam menggunakan semua sumber tenaga, dana dan fasilitas yang ada.

b. Prinsip pengelolaan

Administrator akan memperoleh hasil yang paling efektif dan efisien melalui orang lain dengan jalan melakukan pekerjaan manajemen, yakni merencanakan, mengorganisasikan, mengarahkan dan mengontrol.

c. Prinsip pengutamaan tugas pengelolaan

Jika disertai pekerjaan manajemen dan operatif dalam waktu yang sama, seorang administrator cenderung memberikan prioritas pertama pada pekerjaan operatif. Administrator harus mampu menghindari kecenderungan negatif ini, karena dia terlalu sibuk dengan tugas operatif, maka pekerjaan pokoknya yaitu pengelolaan akan terbengkalai.

d. Prinsip kepemimpinan yang efektif

Seorang administrator yang berhasil dalam tugasnya apabila ia menggunakan gaya kepemimpinan yang efektif, yakni yang memperhatikan dimensi-dimensi hubungan antar manusia (*human relationship*), dimensi pelaksanaan tugas dan dimensi situasi dan kondisi yang ada.

e. Prinsip kerjasama

Seorang administrator akan berhasil baik dalam tugasnya apabila dia mampu mengembangkan kerja sama di antara orang-orang yang terlibat, baik secara horizontal maupun secara vertikal.

2.2.2 Administrasi Pendidikan

Administrasi sendiri seringkali dikaitkan dengan aktivitas administrasi perkantoran, aktivasi ini adalah salah satu bidang yang diperoleh dari aktivasi administrasi sebenarnya. Administrasi jika dilihat dari katanya memiliki arti sempit dan juga luas. Administrasi dalam arti yang sempit memiliki arti kegiatan pencatatan data dan informasi secara tertulis serta proses penyimpanan dokumen untuk dipergunakan kembali di lain waktu. Tata usaha adalah salah satu bidang pekerjaan yang terdapat dalam kegiatan administrasi. Jika ditinjau dalam arti luas, maka administrasi adalah suatu bidang yang di dalamnya terdapat kegiatan manajemen atau pengelolaan pada keseluruhan komponen organisasi yang bertujuan untuk mewujudkan program atau tujuan organisasi. Oleh sebab itu, pekerjaan administrasi dapat dikatakan sebagai pekerjaan manajemen dan operatif [4].

Pengertian administrasi pendidikan dapat ditinjau dari perpaduan dua kata yaitu “administrasi dan pendidikan”. Hakikatnya administrasi pendidikan dapat diartikan sebagai penerapan ilmu administrasi pada dunia pendidikan, diantaranya pembinaan, pengembangan serta pengendalian dalam praktik-praktik pendidikan. Administrasi pendidikan meliputi administrasi sekolah, yaitu administrasi pendidikan yang pelaksanaannya di sekolah. Tata usaha merupakan salah satu alat administrasi pendidikan [4].

Administrasi pendidikan ialah segenap proses pengarahan dan pengintegrasian segala sesuatu baik personal, spiritual, dan material yang terkait dengan tercapainya tujuan pendidikan [4].

Adapun prinsip – prinsip administrasi Pendidikan yaitu [3]:

a. Prinsip fleksibilitas

Dalam penyelenggaraan pendidikan hendaknya dilakukan dengan mengingat faktor-faktor dan kemampuan untuk menyediakan fasilitas bagi berlangsungnya program pendidikan.

b. Prinsip efisien dan efektivitas

Pada hakikatnya efisiensi tidak hanya menyangkut penggunaan waktu secara tepat, melainkan juga menyangkut masalah pendayagunaan tenaga secara optimal. Prinsip ini juga harus digunakan sebagai landasan operasional bagi kegiatan administrasi pendidikan.

c. Prinsip berorientasi pada tujuan

Sesuai dengan pendekatan sistem maka semua kegiatan pendidikan harus berorientasi pada tujuan. Karena administrasi pendidikan merupakan komponen *input* instrumental dalam sistem pendidikan maka untuk menjamin tercapainya tujuan tersebut, tujuan operasional yang sudah dirumuskan itu juga menjadi gantungan orientasi bagi pelaksanaan kegiatan administrasi pendidikan.

d. Prinsip kontinuitas

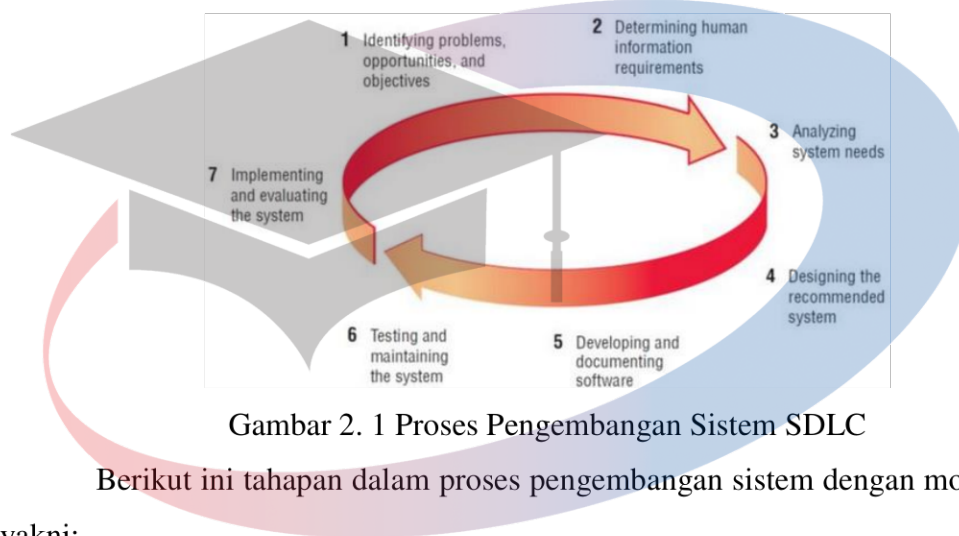
Prinsip kontinuitas ini hendaknya juga dipergunakan sebagai landasan operasional dalam melaksanakan kegiatan administrasi pendidikan.

e. Prinsip pendidikan seumur hidup

Prinsip ini berarti bahwa setiap manusia Indonesia diharapkan untuk selalu berkembang di dalam hidupnya, di lain pihak masyarakat dan pemerintah diharapkan untuk dapat menciptakan situasi yang menantang untuk belajar. Dalam melaksanakan administrasi pendidikan kiranya prinsip tersebut perlu digunakan sebagai landasan operasional

2.3 Siklus Hidup Pengembangan Sistem

System Development Life Cycle atau Siklus Hidup Pengembangan Sistem adalah model pengembangan yang melalui beberapa tahap untuk menganalisis dan merancang sistem dimana sistem tersebut telah dikembangkan dengan sangat baik melalui penggunaan siklus hidup kegiatan penganalisis dan pemakai secara spesifik [5].



Gambar 2. 1 Proses Pengembangan Sistem SDLC

Berikut ini tahapan dalam proses pengembangan sistem dengan model SDLC yakni:

1. Mengidentifikasi masalah, peluang, dan tujuan

Pada tahap pertama dari SDLC, penganalisis mengidentifikasi masalah, peluang serta tujuan dari yang akan dicapai. Tahap ini sangat penting bagi tahap selanjutnya dan juga keberhasilan dari suatu proyek. Analisis melihat dengan jujur apa yang terjadi dalam bisnis. Kemudian bersama-sama dengan anggota organisasional lainnya menentukan dengan tepat masalah-masalah tersebut.

Peluang adalah situasi dimana analisis yakin bahwa peningkatan dapat dilakukan melalui penggunaan sistem informasi terkomputerisasi. Mengukur peluang memungkinkan bisnis untuk mencapai sisi kompetitif atau menyusun standar-standar industri. Mengidentifikasi tujuan menjadi komponen terpenting dalam tahap ini, dimana analis harus mengetahui apa yang sedang dijalankan dalam perusahaan, barulah analis akan bisa melihat beberapa aspek dalam aplikasi-aplikasi sistem informasi untuk membantu bisnis supaya mencapai tujuan dengan menyebut *problem* atau peluang-peluang tertentu.

2. Menentukan syarat-syarat informasi

Dalam tahap ini, analis menentukan apa saja yang menjadi syarat-syarat informasi untuk para pemakai yang terlihat. Analis berusaha keras memahami informasi apa yang dibutuhkan pemakai agar bisa ditampilkan dalam pekerjaan mereka. Analis perlu tahu detail-detail fungsi sistem yang ada seperti 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. Beberapa cara analisis informasi yang dipakai adalah menentukan sampel dan memeriksa data mentah, wawancara, mengamati cara pengambilan keputusan dan lingkungan kantor, serta *prototyping*.

3. Menganalisis kebutuhan sistem

Pada tahap ini, dilakukan analisis terhadap kebutuhan-kebutuhan sistem. Pada ini digunakan *tools* yang dapat membantu analis menentukan kebutuhan. Perangkat yang dimaksud adalah diagram aliran data atau *Data Flow Diagram* (DFD) untuk menyusun daftar *input*, proses, dan *output* fungsi bisnis dalam bentuk grafik terstruktur. Dari DFD, dikembangkan suatu kamus data yang berisikan daftar seluruh item data yang digunakan dalam sistem berikut spesifikasinya, apakah berupa *alphanumeric* atau teks serta berapa banyak spasi yang dibutuhkan saat mencetak.

4. Merancang sistem yang direkomendasikan

Dalam tahap merancang sistem, analisis menggunakan informasi-informasi yang terkumpul sebelumnya untuk mencapai rancangan sistem informasi yang logis. Analisis rancangan prosedur *data entry* sedemikian rupa sehingga data yang dimasukkan ke dalam sistem informasi benar-benar akurat. Selain itu, analis menggunakan teknik-teknik membentuk perancangan tertentu untuk menjamin keefektifan *input* sistem informasi. Bagian dengan *users*, rancangan prosedur *data-entry*, rancangan *user interface*, rancangan *file* atau *database*, serta rancangan *control* dan prosedur *backup* untuk proteksi sistem informasi.

5. Mengembangkan dan mendokumentasikan perangkat lunak

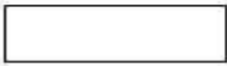
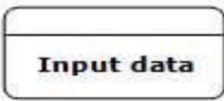
Pada tahap pengembangan sistem, analis bekerja sama dengan *programmer* untuk membuat perangkat lunak yang dibutuhkan serta bekerja sama dengan pemakai untuk mengembangkan dokumentasi perangkat lunak yang efektif, mencakup melakukan prosedur secara manual, bantuan online dan *website* yang memuat fitur *Frequently Asked Question* (FAQ). Kegiatan dokumentasi menunjukkan kepada pemakai apa yang harus dilakukan bila perangkat lunak mengalami masalah. *Programmer* adalah pelaku utama dalam tahap ini karena mereka merancang, membuat kode, dan mengatasi kesalahan-kesalahan dari komputer. Untuk memastikan kualitasnya, *programmer* bisa membuat perancangan dan kode program yang akan dijalankan, menjelaskan bagian-bagian kompleks dari program kepada tim pemrogram lainnya [5].

2.4 Teknik Pengembangan Sistem

2.4.1 Diagram Aliran Data/*Data Flow Diagram* (DFD)

Data Flow Diagram (DFD) merupakan suatu gambaran sistem secara logika. DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau suatu sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut disimpan. DFD dapat digunakan untuk mempresentasikan suatu sistem yang otomatis atau manual dengan melalui gambar yang berbentuk jaringan grafik.

Keempat simbol dasar yang digunakan untuk menetapkan gerakan diagram aliran data adalah sebagai berikut [5]:

SIMBOL	ARTI	CONTOH
	Entitas	
	Aliran data	
	Proses	
	Penyimpanan data	

Gambar 2. 2 Simbol-Simbol DFD

Kegunaan dari masing-masing simbol adalah sebagai berikut:

1. Entitas, untuk menggambarkan bagian lain, sebuah perusahaan, seorang atau sebuah mesin yang dapat mengirim data atau menerima data dari sistem. Entitas diberi nama dengan sebuah kata benda. Entitas yang sama biasanya digunakan lebih dari satu kali atas suatu diagram aliran data tertentu untuk mengetahui jalur aliran data.
2. Aliran data, untuk menunjukkan perpindahan data dari satu titik ke titik yang lain, dengan kepala tanda panah mengarah ke tujuan data. Aliran data yang muncul secara simultan biasanya digambarkan hanya dengan menggunakan tanda panah paralel. Karena sebuah tanda panah menunjukkan seorang, tempat, atau sesuatu, maka harus digambarkan dalam kata benda.
3. Proses, untuk menunjukkan proses transformasi atau perubahan data sehingga aliran data yang meninggalkan suatu proses selalu diberi label yang berbeda dari aliran data yang masuk. Sebuah proses juga harus ditetapkan dengan sebuah nama yang unik yang menunjukkan tingkatannya di dalam diagram.
4. Penyimpanan data, untuk menunjukkan tempat penyimpanan untuk data-data yang memungkinkan penambahan dan perolehan data.

Langkah-langkah dalam membuat diagram DFD adalah sebagai berikut:

1. Mengembangkan *Data Flow Diagram*

Data flow diagram dapat diagram secara sistematis. Untuk memulai suatu diagram aliran data, rangkumlah narasi sistem organisasi menjadi sebuah daftar dengan empat kategori yang terdiri dari entitas eksternal aliran data, proses dan penyimpanan data. Daftar ini untuk membantu menentukan batas-batas sistem yang akan di gambarkan. Setelah daftar unsur-unsur data dasar ini tersusun, mulailah menggambar dengan aliran data.

2. Menciptakan konteks

Menciptakan 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 dengan pengguna dan sebagai hasil analisis dokumen.

3. Menggambar Diagram Level 0 (Level Berikutnya)

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

4. Menciptakan Diagram Anak (tingkat yang lebih detail)

Setiap proses dalam diagram 0 bisa dikembangkan untuk menciptakan diagram anak yang lebih mendetail. Proses pada diagram 0 yang dikembangkan 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 dimana proses induknya juga tidak menghasilkan dan menerima semua aliran data yang menuju atau keluar dari proses induk harus ditunjukkan mengalir kedalam atau keluar dari diagram anak.

2.4.2. Use Case

Use case modeling atau pemodelan *use case* adalah sebuah pendekatan yang memfasilitasi pengembangan berpusatkan kegunaan. *Use case modeling* adalah proses pemodelan fungsi-fungsi sistem dalam konteks-konteks, peristiwa-peristiwa bisnis, siapa yang mengawalinya dan bagaimana sistem itu merespon suatu hal tersebut [6].

Manfaat pemodelan *use case* adalah:

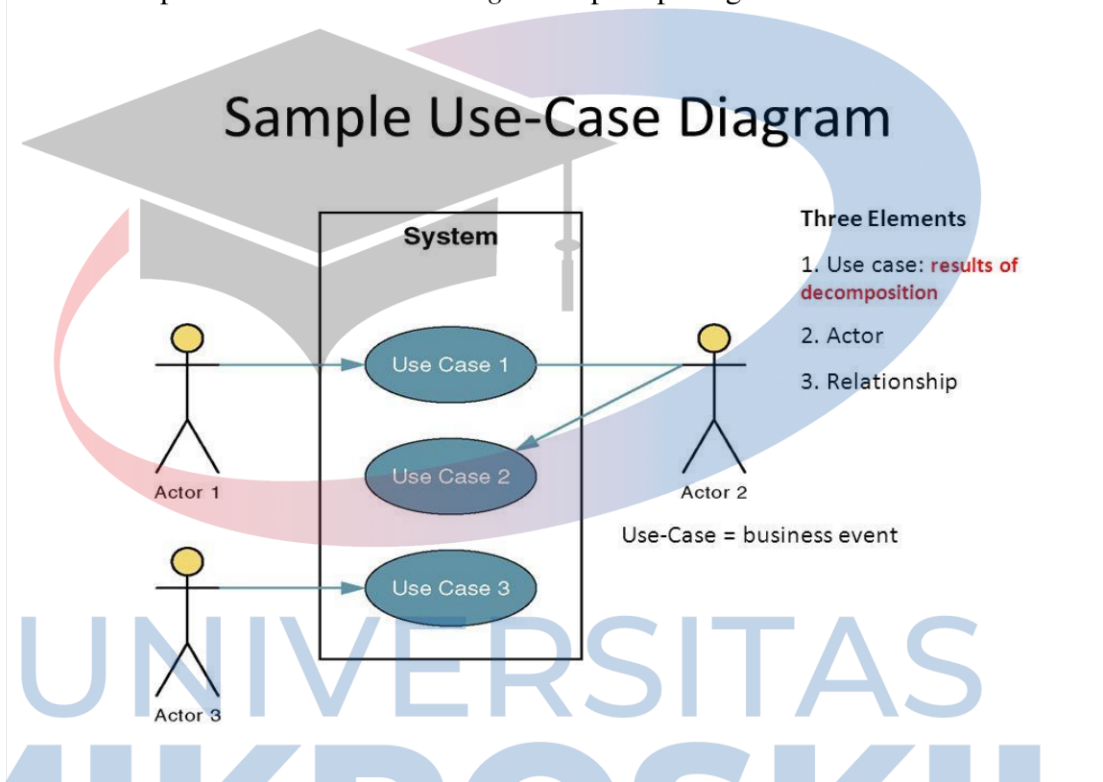
1. Menyediakan *tool* untuk meng-*capture* persyaratan fungsional
2. Membantu menyusun ulang lingkup sistem menjadi bagian-bagian yang lebih cepat dikelola
3. Menyediakan alat komunikasi dengan para pengguna dan *stakeholder* yang berhubungan dengan fungsionalitas sistem. *Use case* menyajikan bahasa umum yang dapat dipahami oleh berbagai macam *stakeholder*
4. Memberikan cara bagaimana mengidentifikasi, menetapkan, melacak, mengontrol, dan mengelola kegiatan pengembangan sistem inkremental dan interaktif
5. Menyajikan panduan untuk mengsystemasi lingkup, usaha dan jadwal proyek
6. Menyajikan garis pokok pengujian, khususnya menentukan rencana tes dan *use case*
7. Menyajikan *tool* untuk melacak persyaratan
8. Menyajikan titik mulai/awal untuk identifikasi objek data atau entitas
9. Menyajikan spesifikasi fungsional untuk mendesain antar muka pengguna dan sistem
10. Menyajikan alat untuk menentukan persyaratan akses *database* dalam menambah, mengubah, menghapus dan membaca

Use case diagram adalah diagram yang menggambarkan interaksi antara sistem internal dengan sistem eksternal pengguna. Dengan kata lain, secara garis besar menggambarkan siapa yang akan menggunakan sistem dan dengan cara apa pengguna mengharapkan untuk berinteraksi dengan sistem.

Use case adalah urutan langkah-langkah yang secara tindakan saling terkait baik terotomatisasi maupun secara manual, untuk tujuan melengkapi satu tugas bisnis tunggal. *Actor* adalah segala sesuatu yang perlu berinteraksi dengan sistem untuk

pertukaran informasi. *Association* adalah hubungan antar pelaku/actor dengan *use case* dimana terjadi interaksi diantara mereka. *Extension use case* adalah *use case* yang lebih kompleks untuk menyederhanakan masalah orisinal dan karena itu memperluas fungsinya. *Abstract use case* adalah *use case* yang mengurangi redundansi antara dua atau lebih *use case* dengan menggabungkan langkah-langkah yang biasa ditemukan pada *use case* tersebut [6].

Adapun contoh *Use Case Diagram* seperti pada gambar berikut:



Gambar 2. 3 *Use Case Diagram*

2.4.3 PIECES

Proses dan teknik yang digunakan oleh analisis sistem untuk mengidentifikasi, menganalisis, dan memahami persyaratan sistem disebut *requirement discovery/* penemuan persyaratan. Persyaratan sistem menentukan apa yang seharusnya dikerjakan oleh sistem informasi atau property serta kualitas apa yang harus dimiliki oleh sistem. Persyaratan sistem yang menetapkan apa yang harusnya dilakukan oleh sistem informasi sering disebut persyaratan fungsional. Persyaratan sistem yang menetapkan properti atau kualitas yang harus dimiliki oleh sistem sering disebut persyaratan non fungsional.

Kerangka PIECES memberikan alat unggul untuk menggolongkan persyaratan sistem. Keuntungan menggolongkan berbagai tipe persyaratan adalah kemampuan untuk menggolongkan persyaratan tersebut untuk tujuan pelaporan, pelacakan, dan validasi. Hal tersebut membantu identifikasi persyaratan sistem secara cermat [6]:

1. *Performance* (kinerja)

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

- a. *Throughput*, yaitu jumlah pekerjaan/*output/deliverabk*ursus yang dapat dilakukan/ dihasilkan pada saat tertentu.
- b. *Response time*, yaitu waktu yang dibutuhkan untuk menyekursusaikan serangkaian kegiatan untuk menghasilkan *output / deliverabk*ursus tertentu.

2. *Information* (informasi)

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

- a. Keluaran (*output*): Suatu sistem dalam memproduksi keluaran.
- b. Masukan (*input*): Dalam memasukkan suatu data sehingga kemudian diolah untuk menjadi informasi yang berguna.

3. *Economic* (ekonomi)

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

4. *Control* (pengendalian)

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

5. *Efficiency* (efisiensi)

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

6. *Service* (layanan)

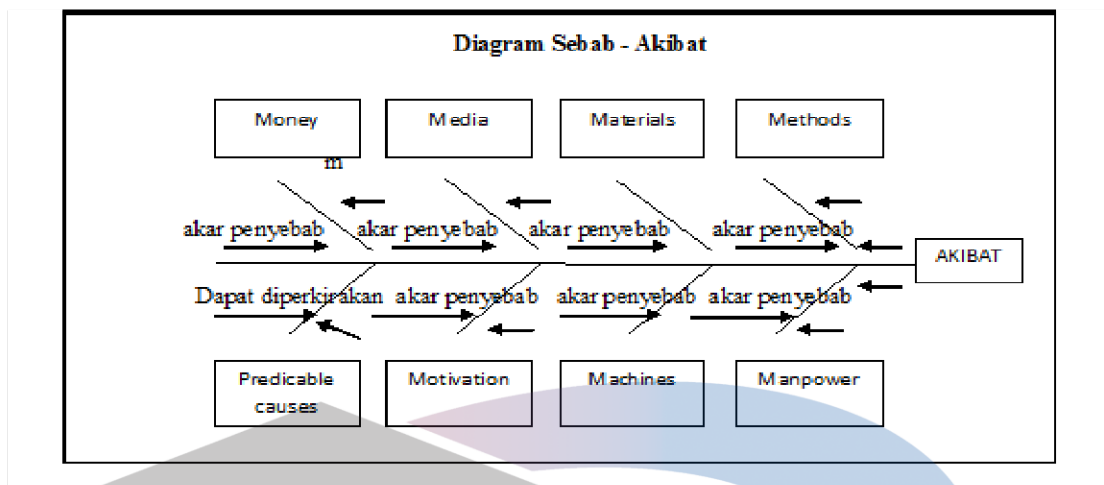
Menilai apakah prosedur yang ada saat ini masih dapat diperbaiki kemampuannya untuk mencapai peningkatan kualitas layanan. Buatlah kualitas layanan yang sangat *user friendly* untuk *end-user* (pengguna) sehingga pengguna mendapatkan kualitas layanan yang baik.

2.4.4. Ishikawa Diagram (*Diagram Fishbone*)

Ishikawa diagram adalah sebuah alat grafis yang digunakan untuk mengidentifikasi, mengeksplorasi, menggambarkan suatu masalah, sebab dan akibat dari masalah. Ishikawa diagram sering juga disebut diagram tulang ikan (*diagram fishbone*) karena menyerupai tulang ikan.

Konsep dasar dari *diagram fishbone* adalah nama masalah yang mendapat perhatian dicantumkan disebelah kanan diagram (pada kepala ikan) dan penyebab masalah yang mungkin digambarkan sebagai tulang-tulang dari tulang utama. Secara khusus “tulang-tulang” ini mendeskripsikan empat kategori dasar yaitu: *material*, mesin, kekuatan manusia, dan metode (empat M yaitu: *material*, *machine*, *manpower*, *method*). Kategori alternatif atau tambahan meliputi tempat, prosedur, kebijakan, dan orang (empat P : *place*, *procedure*, *policy*, *people*) atau lingkungan sekeliling, pemasok, sistem, dan keterampilan (empat S yaitu: *surrounding*, *supplier system*, *skill*) [6].

Adapun contoh *Diagram Fishbone* atau Ishikawa diagram seperti pada gambar berikut:



Gambar 2. 4 Diagram Fishbone

2.4.4. Kamus Data

Kamus data adalah suatu aplikasi khusus dari jenis-jenis kamus yang digunakan sebagai referensi kehidupan setiap hari. Kamus data merupakan hasil referensi data mengenai data, suatu data yang disusun oleh analisis sistem untuk membimbing mereka selama melakukan analisis dan desain. Sebagai suatu dokumen, kamus data mengumpulkan dan mengkoordinasi istilah-istilah data tertentu dan menjelaskan apa arti setiap istilah yang ada [5].

Sebagai tambahan untuk dokumentasi serta mengurangi terjadinya redundansi kamus data bisa digunakan untuk :

1. Memvalidasi 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 dalam *file-file*.
4. Mengembangkan logika untuk proses-proses diagram aliran data.
5. Membuat XML.

Struktur data dari kamus data biasanya digambarkan menggunakan notasi aljabar. Metode ini memungkinkan analisis membuat suatu gambaran mengenai elemen-elemen yang membentuk struktur data bersama-sama dengan informasi mengenai elemen tersebut. Notasi aljabar pada struktur data kamus data menggunakan simbol-simbol seperti :

1. Tanda sama dengan (=) artinya “terdiri dari”.

2. Tanda *plus* (+) artinya “dan”.
3. Tanda kurung kurawal { }, menunjukkan elemen *repetitive* atau kelompok yang berulang-ulang. Kemungkinan bisa ada satu atau beberapa elemen yang berulang di dalam kelompok tersebut. Kelompok berulang bisa mengandung keadaan-keadaan tertentu seperti jumlah pengulangan.
4. Tanda kurung siku [], menunjukkan salah satu dari dua situasi tertentu. Satu elemen bisa ada sedangkan elemen lainnya juga ada, tetapi tidak bisa keduanya ada secara bersama-sama. Elemen-elemen yang ada di dalam tanda kurung ini saling terpisah satu sama lain.
5. Tanda kurung (), menunjukkan elemen yang bersifat pilihan. Elemen-elemen yang bersifat pilihan ini bisa dikosongkan pada layar masukan atau bisa juga dengan memuat spasi dan nol untuk *field-field numeric* pada struktur *file*.

2.4.6. Normalisasi

Normalisasi adalah transformasi tinjauan pemakai yang kompleks dan data tersimpan pada sekumpulan bagian-bagian struktur data yang kecil dan stabil sehingga struktur data yang dinormalisasikan lebih mudah diatur daripada struktur data lainnya [6].

Ada 3 tahapan dalam pembuatan normalisasi yaitu :

1. Tahap pertama
Tahap pertama dari proses meliputi menghilangkan semua kelompok terulang dan mengidentifikasi kunci utama. Untuk mengerjakannya, hubungan perlu di pecah ke dalam dua atau lebih hubungan. Pada saat ini, hubungan mungkin sudah menjadi bentuk normalisasi ketiga, bahkan banyak tahap akan diperlukan untuk mentransformasikan hubungan ke bentuk normalisasi ketiga.
2. Tahap kedua
Tahap kedua menjamin bahwa semua atribut bukan kunci sepenuhnya tergantung pada kunci utama. Semua ketergantungan parsial diubah dan diletakkan dalam hubungan lain.
3. Tahap ketiga
Mengubah ketergantungan transitif manapun. Suatu ketergantungan transitif adalah keadaan dimana atribut bukan kunci tergantung pada atribut kunci lain.

Dalam normalisasi terdapat jenis-jenis normalisasi yaitu :

Hubungan tidak normalisasi

↓ Tahap 1 Menghilangkan kelompok terulang

Hubungan Normalisasi (1NF)

↓ Tahap 2 Mengubah ketergantungan parsial

Hubungan bentuk normalisasi kedua (2NF)

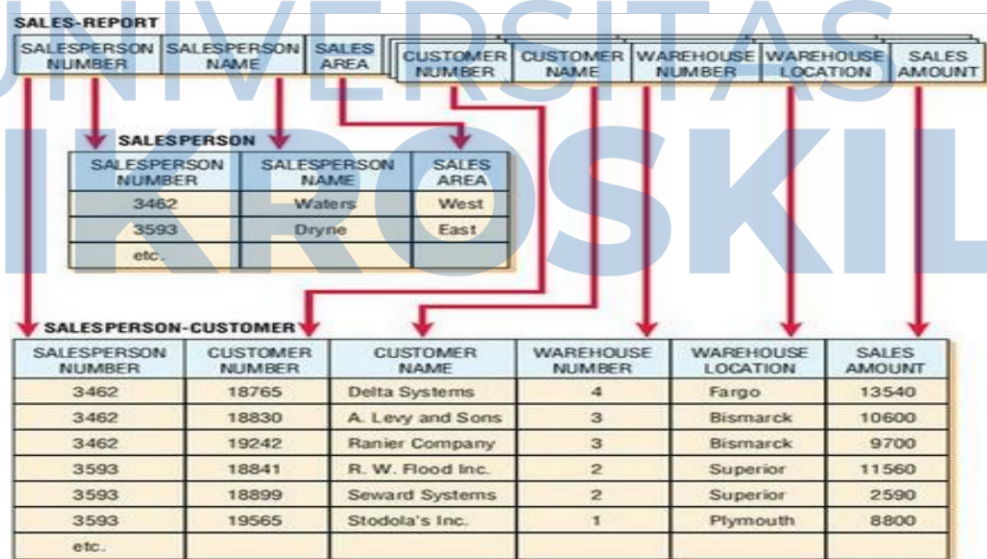
↓ Tahap 3 Mengubah ketergantungan transitif

Hubungan bentuk normalisasi ketiga (3NF)

1. Bentuk normalisasi pertama (1NF / *First Normal Form*)

Dalam bentuk normalisasi pertama proses meliputi menghilangkan semua kelompok terulang dan mengidentifikasi kunci utama. Untuk mengerjakannya, hubungan perlu dipecah kedalam dua atau lebih hubungan. Pada titik ini, hubungan mungkin sudah menjadi bentuk normalisasi ketiga, bahkan lebih banyak tahap akan diperlukan untuk mentransformasi hubungan ke bentuk normalisasi ketiga.

Berikut adalah contoh gambar bentuk normalisasi pertama (1NF/*First Normal Form*):

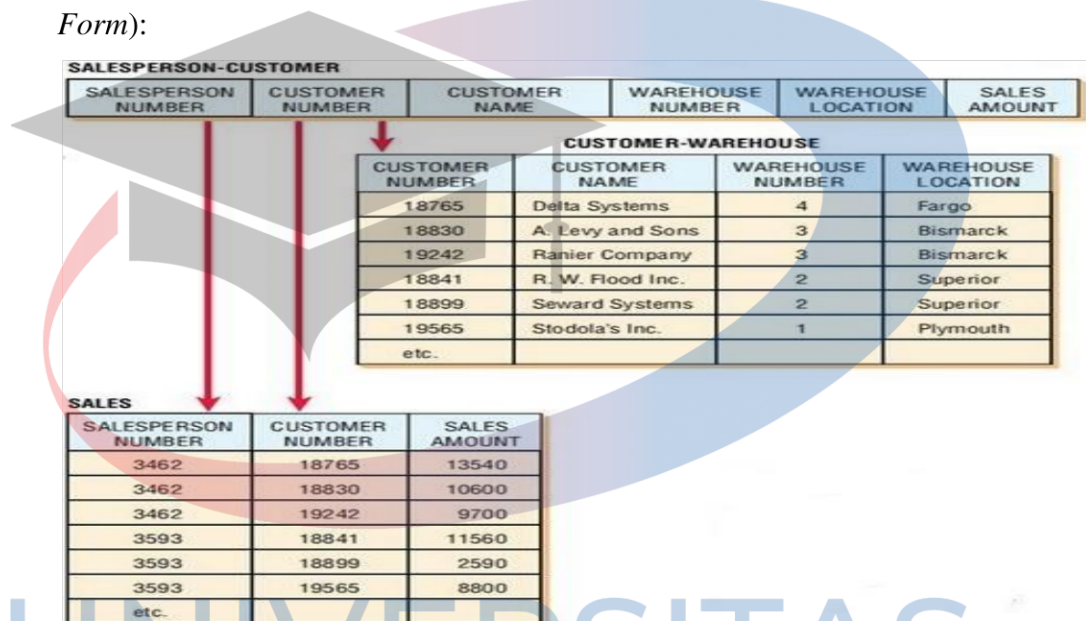


Gambar 2. 5 Bentuk Normalisasi 1NF

2. Bentuk normal kedua (2NF/Second Normal Form)

Dalam bentuk normalisasi kedua, semua atribut akan tergantung secara fungsional pada kunci utama. Oleh karena itu, langkah selanjutnya adalah menghilangkan semua atribut yang bergantung sebagian dan meletakkannya dalam hubungan lain.

Berikut adalah contoh gambar bentuk normalisasi kedua (2NF/Second Normal Form):

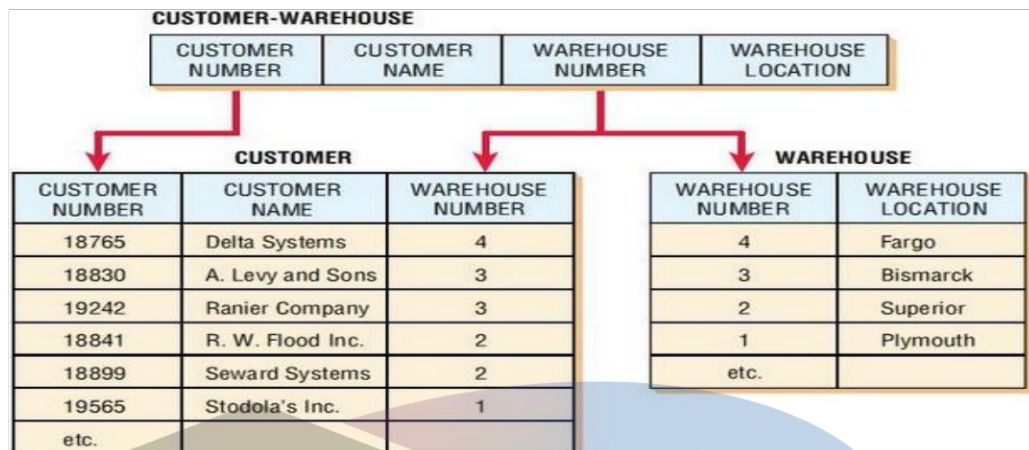


Gambar 2. 6 Bentuk Normalisasi 2NF

3. Bentuk normal ketiga (3NF/Third Normal Form)

Suatu hubungan normalisasi adalah bentuk normalisasi ketiga jika semua atribut bukan kunci sepenuhnya tergantung secara fungsional pada kunci utama dan tidak terdapat ketergantungan transitif (bukan kunci).

Berikut adalah contoh gambar bentuk normalisasi ketiga (3NF/Third Normal Form):



Gambar 2. 7 Bentuk Normalisasi 3NF

2.4.5 Basis Data

Basis data adalah pusat sumber data yang dapat dipakai oleh banyak pemakai untuk berbagai aplikasi. Inti dari basis data adalah *Data Base Managemen System* yang memperoleh pembuatan, modifikasi, pembaruan basis data, mendapatkan kembali data, dan membangkitkan laporan [5].

Tujuan dari basis data yaitu:

- Memastikan bahwa data dapat dipakai diantara pemakai untuk berbagai aplikasi.
- Memelihara baik keakuratan maupun kekonsistenan.
- Memastikan bahwa semua data yang diperlukan untuk aplikasi sekarang dan yang akan datang disediakan dengan cepat.
- Membolehkan basis data untuk berkembang.

Tujuan yang telah disebutkan diatas memberikan keuntungan dan kerugian pendekatan basis data. Pemakaian data berarti bahwa data perlu disimpan hanya sekali. Membantu mencapai integritas data, karena mengubah data yang dise Kursus akan lebih mudah dan dapat dipercaya jika data muncul hanya sekali dalam banyak *file* berbeda.

Ketika pemakai memerlukan data khusus, basis data yang dirancang dengan baik (*well-designed*) memenuhi lebih dahulu kebutuhan data yang demikian. Akibatnya, data memiliki kesempatan tersedia yang lebih baik dalam basis data daripada dalam sistem *file* yang konvensional. Basis data yang dirancang dengan baik

juga lebih fleksibel dari pada *file* terpisah, karena itu, basis data dapat berkembang seperti pada perubahan kebutuhan pemakai dan aplikasinya.

Akhirnya, pendekatan hasil basis data memiliki keuntungan yang membolehkan pemakai untuk memiliki pandangan sendiri mengenai data. Pemakai tidak perlu memperhatikan struktur sebenarnya basis data atau penyimpanan fisiknya.

Kerugian pertama pendekatan basis data adalah bahwa semua data disimpan dalam satu tempat. Oleh karena itu, data lebih mudah diserang bencana dan membutuhkan *backup* yang lengkap. Terdapat resiko bahwa administrator basis data menjadi satu-satunya orang yang mempunyai hal istimewa atau kemampuan cukup untuk mendekati data. Prosedur birokratis perlu untuk memodifikasi atau memperbaharui basis data secara lengkap yang terlihat tidak dapat diatasi [5].

Kerugian lain terjadi ketika usaha untuk mencapai 2 tujuan efektif untuk mengatur sumber data, seperti:

1. Menjaga waktu yang diperlukan untuk *insert*, *update*, *delete*, dan memperoleh kembali data untuk suatu jumlah yang dapat dipertahankan.
2. Menjaga harga penyimpanan data untuk jumlah yang dapat diterima.

UNIVERSITAS
MIKROSKIL