

BAB II

TINJAUAN PUSTAKA

2.1 Sosial Media

Sosial media merupakan aplikasi platform *online* berbasis jaringan internet yang digunakan seseorang untuk berinteraksi dengan orang lain diseluruh dunia [1]. Sosial media adalah fitur berbasis *website* yang dapat membentuk jaringan serta memungkinkan seseorang untuk berinteraksi dalam sebuah komunitas [2]. Sosial media juga merupakan sebuah kelompok jaringan yang berbasiskan aplikasi dalam internet yang di bangun berdasar teknologi dan konsep web 2.0, sehingga dapat membuat pengguna (*users*) menciptakan dan mengganti konten yang di sebarakan [3].

Sosial Media merupakan istilah yang lazim digunakan untuk *website*, alat-alat *online*, dan teknologi komunikasi interaktif lainnya yang memungkinkan pengguna untuk berinteraksi satu sama lain dengan, yaitu untuk berbagi informasi, opini, pengetahuan, atau kepentingan [4].

Dari beberapa penjelasan diatas, penulis mendefinisikan, sosial media merupakan sebuah fitur di dalam web yang digunakan seseorang untuk membuat komunitas dalam sistem, serta menjalin hubungan dengan komunitas lain untuk saling bertukar informasi.

2.2 Tracer Study

Tracer Study adalah sebuah pendekatan yang memungkinkan lembaga pendidikan untuk memperoleh informasi tentang kemungkinan kekurangan-kekurangan dalam proses pendidikan dan proses pembelajaran dan dapat membentuk dasar perencanaan kegiatan untuk peningkatan masa depan [5]. *Tracer Study* adalah survey yang terstandarisasi terhadap lulusan institusi pendidikan tinggi yang dilakukan beberapa waktu setelah lulusan tersebut meninggalkan institusi pendidikan tingginya [6].

Tracer Study adalah studi pelacakan jejak lulusan/alumni yang dilakukan antara 1-3 tahun setelah lulus dan bertujuan untuk mengetahui *outcome* pendidikan dalam bentuk transisi dari dunia pendidikan tinggi ke dunia kerja [7]. *Tracer study* adalah diperolehnya informasi tentang relevansi program pendidikan yang

diselenggarakan dengan kebutuhan lapangan. Menjamin adanya relevansi antara program pendidikan dengan kebutuhan lapangan merupakan keharusan bagi setiap lembaga pendidikan [8]. *Tracer Study* adalah suatu cara untuk memperbaiki relevansi kurikulum dan memberikan manfaat yang ditargetkan untuk lulusan dalam meningkatkan program pendidikan [9].

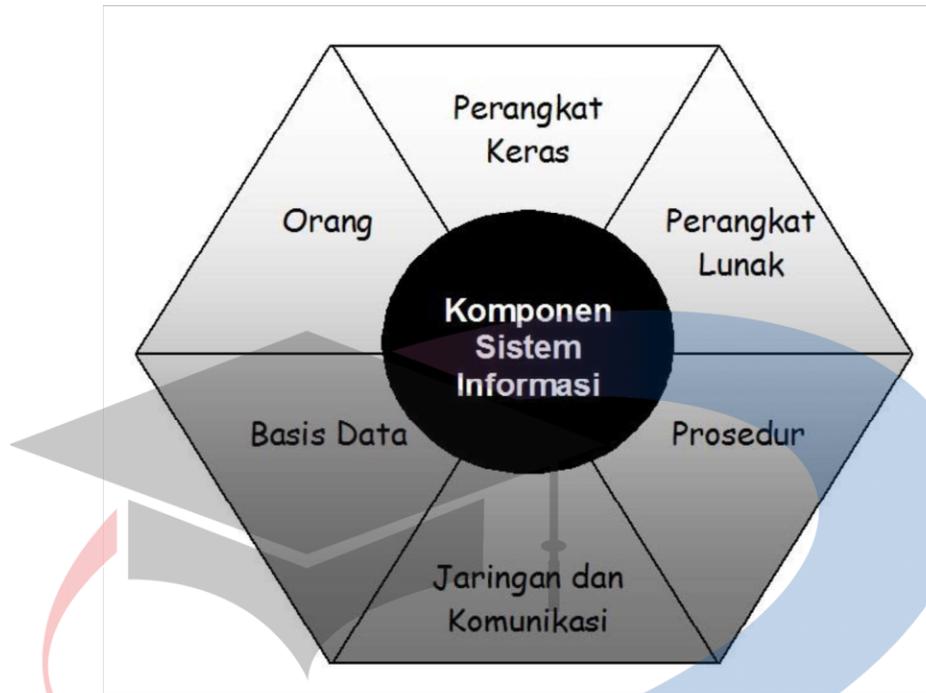
Studi pelacak (*Tracer Study*) atau survei pascasarjana adalah survei yang berdiri sendiri (dalam bentuk tertulis atau lisan) dari lulusan dari lembaga pendidikan, yang berlangsung beberapa waktu setelah kelulusan atau akhir pelatihan. Subjek penelitian pelacak dapat bermacam macam, tetapi topik umum mencakup pertanyaan tentang kemajuan studi, transisi ke pekerjaan, masuk kerja, karir pekerjaan, penggunaan kompetensi yang dipelajari, pekerjaan saat ini dan obligasi ke lembaga pendidikan (sekolah, pusat, universitas) [10].

2.3 Sistem Informasi

Sistem Informasi merupakan gabungan dari empat bagian utama yang mencakup perangkat lunak (*software*), perangkat keras (*hardware*), infrastruktur dan sumber daya manusia (SDM) yang terlatih. Keempat bagian utama ini saling berkaitan untuk menciptakan sebuah sistem yang dapat mengolah data menjadi informasi yang bermamfaat [13].

Sistem Informasi adalah memberikan nilai tambah terhadap proses, produksi, kualitas, manajemen, pengambilan keputusan dan pemecahan masalah, serta keunggulan kompetitif yang tentu saja sangat berguna bagi kegiatan bisnis [11].

Komponen – komponen Sistem Informasi dapat dilihat pada gambar di bawah ini [13]:



Gambar 2. 1 Komponen-Komponen Sistem Informasi

1. Perangkat Keras (*hardware*), yang mencakup peranti-peranti fisik seperti komputer dan printer
2. Perangkat Lunak (*software*), yaitu sekumpulan instruksi yang memungkinkan perangkat keras memproses data.
3. Prosedur, yaitu sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki
4. Orang, yakni semua pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan, dan penggunaan keluaran sistem informasi.
5. Basis data (*database*), yaitu kumpulan tabel, hubungan, dan lain-lain yang berkaitan dengan penyimpanan data.
6. Jaringan komputer dan komunikasi data, yaitu sistem penghubung yang memungkinkan sumber (*resource*) dipakai secara bersama atau diakses oleh sejumlah pemakai.

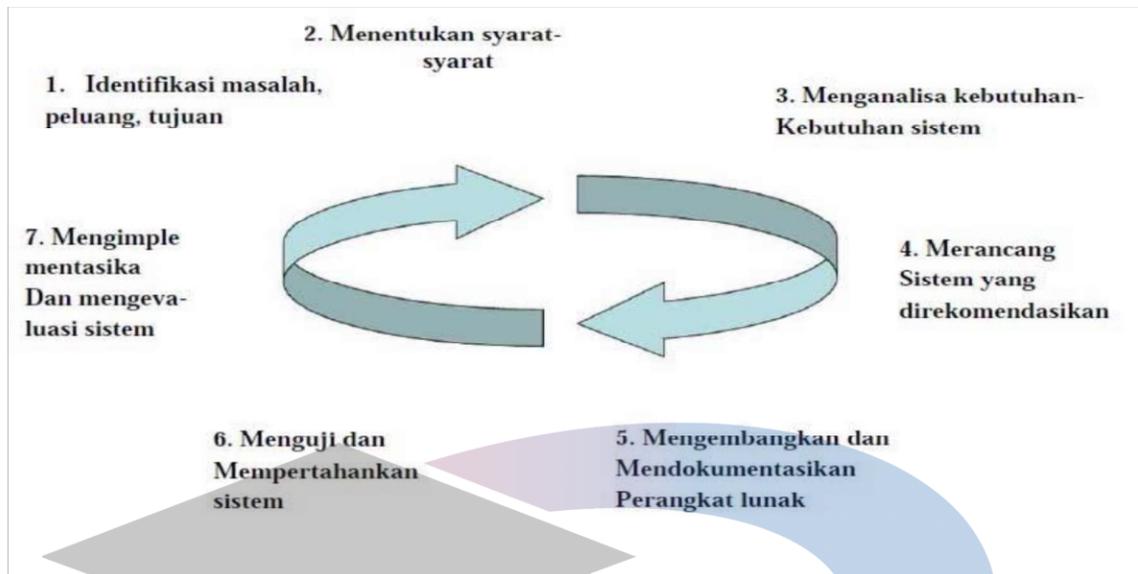
Sistem Informasi juga mempunyai karakteristik yaitu [13]:

1. Sistem Informasi memiliki komponen yang berupa subsistem yang merupakan elemen-elemen yang lebih kecil yang membentuk sistem informasi tersebut misalnya bagian *input*, proses, *output*.
2. Ruang Lingkup Sistem Informasi yaitu ruang lingkup yang ditentukan dari awal pembuatan yang meupakan garis batas lingkup kerja sistem tersebut sehingga sistem informasi tersebut tidak bersinggungan dengan sistem informasi lainnya.
3. Tujuan Sistem Informasi adalah hal pokok yang harus ditentukan dan dicapai dengan menggunakan sistem informasi tersebut, sebuah informasi dianggap berhasil apabila dapat mencapai tujuan tersebut.
4. Lingkungan Sistem Informasi yaitu sesuatu yang berada diluar ruang lingkup sistem informasi yang dapat mempengaruhi sistem informasi, hal ini turut dipertimbangkan pada saat perencanaan sistem informasi. Sistem informasi dikembangkan untuk tujuan yang berbeda-beda, tergantung pada kebutuhan bisnis.

2.4 Siklus Hidup Pengembangan Sistem

Siklus hidup pengembangan sistem adalah pendekatan melalui beberapa tahap untuk menganalisis dan merancang sistem yang di mana sistem tersebut telah di kembangkan dengan sangat baik melalui penggunaan siklus kegiatan penganalisis dan pemakai secara spesifik [14].

Penganalisis tidak sepakat dengan berapa banyaknya tahap yang ada di dalam siklus hidup. Pengembangan sistem, namun mereka umumnya memuji pendekatan terorganisir mereka. Disini kita telah membagi siklus ke dalam tujuh tahap meskipun masing-masing tahap di tampilkan secara terpisah, namun tidak pernah tercapai sebagai satu langkah terpisah. Melainkan beberapa aktivitas muncul secara simultan, dan aktivitas tersebut di lakukan berulang-ulang. Lebih berguna lagi memikirkan bahwa SHPS bisa di capai dalam tahap-tahap (dengan aktivitas berulang yang saling tumpang tindih satu sama lainnya dan menuju ke tujuan terakhir) dan tidak dalam langkah-langkah terpisah [14].



Gambar 2. 2 Siklus Hidup Pengembangan Sistem

Berikut Tahapan Siklus Hidup Pengembang Sistem [14]:

1. Mengidentifikasi Masalah, Peluang dan Tujuan

Ditahap pertama di siklus hidup pengembangan sistem ini, penganalisis mengidentifikasi masalah peluang dan tujuan – tujuan yang hendak di capai. Tahap pertama ini berarti bahwa penganalisis melihat dengan jujur pada apa yang terjadi di dalam bisnis. Kemudian bersama-sama dengan anggota organisasional lain, penganalisis menentukan dengan tepat masalah-masalah tersebut. Peluang adalah situasi dimana penganalisis yakin bahwa peningkatan bisa dilakukan melalui penggunaan sistem informasi terkomputerisasi

Mengidentifikasi tujuan yang juga menjadi komponen terpenting di tahap pertama ini. Pertama penganalisis harus menemukan apa yang sedang dilakukan dalam bisnis. Barulah kemudian penganalisis akan bisa melihat beberapa aspek dalam aplikasi-aplikasi sistem informasi untuk membantu bisnis mencapai tujuan-tujuannya dengan menyebut problem atau peluang-peluang tertentu.

Orang- orang yang terlibat dalam tahap pertama ini di antaranya ialah pemakai, penganalisis dan manajer sistem yang bertugas untuk mengkoordinasi proyek. Aktivitas dalam tahap ini meliputi wawancara terhadap manajemen pemakai, menyimpulkan pengetahuan yang di peroleh, mengestimasi cakupan proyek dan

mendokumentasikan hasil-hasilnya. *Output* tahap ini ialah laporan yang layak berisikan definisi problem dan ringkasan tujuan.

2. Menentukan Syarat-Syarat Informasi

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

Dalam tahap ini, penganalisis berusaha keras untuk memahami informasi apa yang di butuhkan pemakai agar bisa ditampilkan dalam pekerjaan mereka. 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 operasi dan pegawai operasional dan penganalisis sistem harus perlu tahu detail-detail fungsi-fungsi sistem yang ada. Dan pada akhir tahap ini, penganalisis akan bisa memahami bagaimana fungsi-fungsi bisnis dan melengkapi informasi tentang masyarakat, tujuan, data dan prosedur yang terlibat.

3. Menganalisis Kebutuhan Sistem

Tahap berikutnya ialah menganalisis kebutuhan-kebutuhan sistem. Sekali lagi, perangkat dan teknik-teknik tertentu akan membantu penganalisis menentukan kebutuhan. Perangkat yang dimaksud ialah penggunaan diagram aliran data untuk menyusun daftar *input*, proses dan *output* fungsi bisnis dalam grafik terstruktur. Selama tahap ini, penganalisis sistem juga menganalisis keputusan terstruktur yang dibuat. Keputusan terstruktur adalah keputusan-keputusan dimana kondisi, kondisi alternatif, tindakan serta aturan tindakan ditetapkan. Ada tiga metode utama untuk menganalisis keputusan terstruktur, yakni: bahasa Inggris terstruktur, rancangan keputusan, dan pohon keputusan.

Pada tahap ini, penganalisis sistem menyiapkan suatu proposal sistem yang berisikan ringkasan apa saja yang di temukan, analisis biaya/keuntungan alternatif yang tersedia, serta rekomendasi atas apa saja yang ditemukan (bila ada) yang harus di lakukan. Bila salah satu rekomendasi tersebut bisa diterima oleh manajemen, lalu penganalisis akan memprosesnya lebih lanjut.

4. Merancang Sistem Yang Direkomendasi

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

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

Terakhir penganalisis harus merancang prosedur-prosedur *back-up* dan kontrol untuk melindungi sistem dan data serta untuk membuka 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 subprogram yang sudah tertulis.

5. Mengembangkan dan Merekomendasi Perangkat Lunak

Dalam tahap ini penganalisis bekerja bersama-sama dengan pemrogram untuk mengembangkan suatu perangkat lunak awal yang di perlukan. Selama tahap ini, penganalisis bekerja sama dengan pemakai untuk mengembangkan dokumentasi perangkat lunak yang efektif, mencakup melakukan prosedur secara manual, bantuan online, dan website yang membuat fitur *Frequently Asked Questions* (FAQ) di file "*Read Me*" yang dikirimkan bersama-sama dengan perangkat lunak baru. Kegiatan dokumentasi menunjukkan kepada pemakai tentang cara penggunaan perangkat lunak dan apa yang harus di lakukan bila perangkat lunak mengalami masalah.

Pemrogram adalah pelaku utama dalam tahap ini karena mereka merancang, membuat kode dan mengatasi kesalahan-kesalahan dari program komputer. 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 dahulu. Sebagian pengujian dilakukan oleh pemrogram sendiri, dan lainnya dilakukan oleh penganalisis sistem. Mempertahankan sistem dan dokumentasinya dimulai di tahap ini dan dilakukan secara rutin selama sistem informasi dijalankan. Untuk 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 dari pengembangan sistem, penganalisis membantu untuk mengimplementasikan sistem informasi. Tahap ini melibatkan pelatihan bagi pemakai untuk mengendalikan sistem. Selain itu penganalisis perlu merencanakan konversiperlahan dari sistem lama ke sistem baru. Proses ini mencakup pengubahan file-file dari format lama ke format baru atau membangun suatu basis data, menginstal peralatan dan membawa sistem baru untuk di produksi.

Evaluasi yang ditunjukkan sebagai bagian dari tahap terakhir dari siklus hidup pengembangan sistem biasanya dimaksudkan untuk penambahan. Sebenarnya, evaluasi dilakukan di setiap tahap. Kriteria utama yang harus dipenuhi ialah apakah pemakai yang dituju benar-benar menggunakan sistem.

2.5. Alat Bantu Pengembangan Sistem

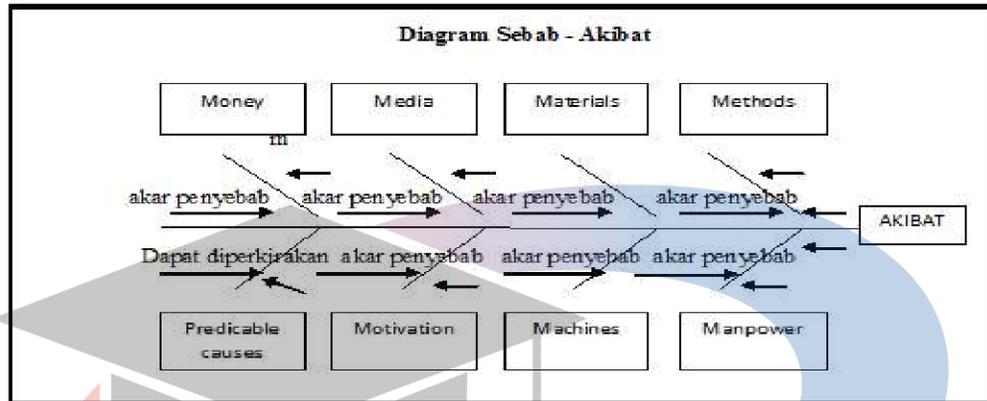
2.5.1 Diagram *Fishbone*

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. [15]

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*) [15].

Adapun Penyusunan diagram *fishbone* ditunjukkan pada gambar dibawah ini [9]:



Gambar 2. 3 Penyusunan Diagram Fishbone

2.5.2 Pemodelan *Use-Case Diagram*

Use-Case diagram merupakan diagram yang menggambarkan interaksi antara sistem dengan sistem eksternal dan pengguna. Dengan kata lain, secara grafis menggambarkan siapa yang menggunakan sistem dan dengan cara apa pengguna mengharapkan untuk berinteraksi dengan sistem [15].

Use-Case Modeling / pemodelan *use-case* adalah sebuah pendekatan yang memfasilitasi pengembangan yang berpusatkan kegunaan. Pemodelan *use-case* terbukti menjadi sebuah alat bantu yang sangat berharga dalam menghadapi tantangan untuk menentukan apa yang harus dilakukan oleh sistem menurut perspektif pengguna dan stakeholder. Pemodelan *use-case* secara luas dikenal sebagai aplikasi terbaik dalam menentukan, mendokumentasikan, dan memahami persyaratan fungsional sistem informasi [15].

Pemodelan *use-case* mengidentifikasi dan menggambarkan fungsi-fungsi sistem dengan menggunakan alat yang disebut *use-case*. *Use-case* menggambarkan fungsi-fungsi sistem dari sudut pandang pengguna eksternal dan dalam sebuah cara dan *terminology* yang mereka pahami. Agar permintaan tersebut dipenuhi secara akurat dan menyeluruh, diperlukan tingkat keterlibatan pengguna yang sangat tinggi,

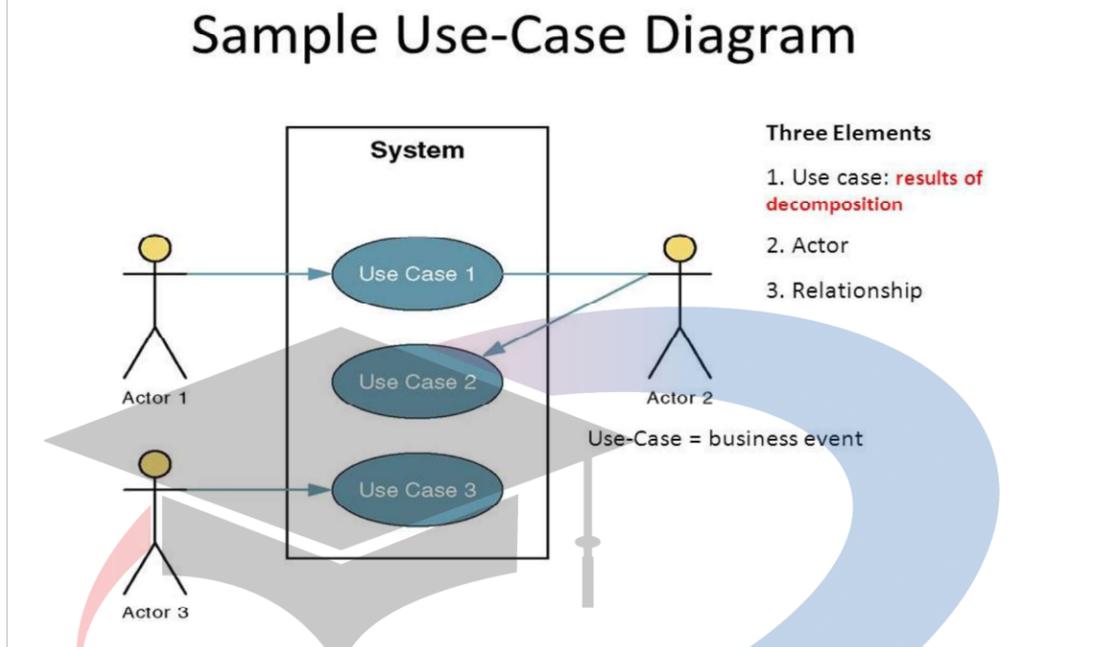
juga pakar yang mempunyai pengetahuan mengenai proses bisnis atau kejadian bisnis [15].

Use-case merupakan hasil penyusunan kembali lingkup fungsionalitas sistem menjadi banyak pernyataan fungsionalitas sistem yang lebih kecil. *Use-case* disajikan secara grafis dengan elips horizontal dengan nama *use-case* muncul diatas, bawah atau di dalam elips tersebut. Sebuah *use-case* mempresentasikan satu tujuan tunggal dari sistem dan menggambarkan satu rangkaian kegiatan dan interaksi pengguna untuk mencapai tujuan. Pembuatan *use-case* terbukti menjadi teknik yang sangat bagus untuk memahami dan mendokumentasikan persyaratan fungsional, tapi cerita baik. *Use-case* sendiri bukan merupakan persyaratan fungsional, tapi cerita (skenario) yang disampaikan oleh *use-case* terdiri dari satu atau lebih persyaratan [15].

Use-case awalnya ditentukan pada tahap persyaratan dari siklus hidup dan akan diperbaiki secara bertahap di sepanjang siklus hidup. Selama penemuan persyaratan, *use-case* digunakan untuk menangkap esensi masalah-masalah bisnis dan untuk memodelkan (pada tingkat tinggi) fungsionalitas dari sistem yang diajukan. Selain itu, *use-case* merupakan titik awal untuk mengidentifikasi entity data atau objek sistem [15].

Use case adalah urutan langkah-langkah yang secara tindakan saling terkait baik terotomatisasi maupun secara manual, untuk tujuan melengkapinya 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 arena 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. [15]

Adapun contoh *Use Case Diagram* seperti pada gambar berikut: [15]



Gambar 2. 4 Contoh Diagram Model Use-Case

2.5.3 Data Flow Diagram (DFD)

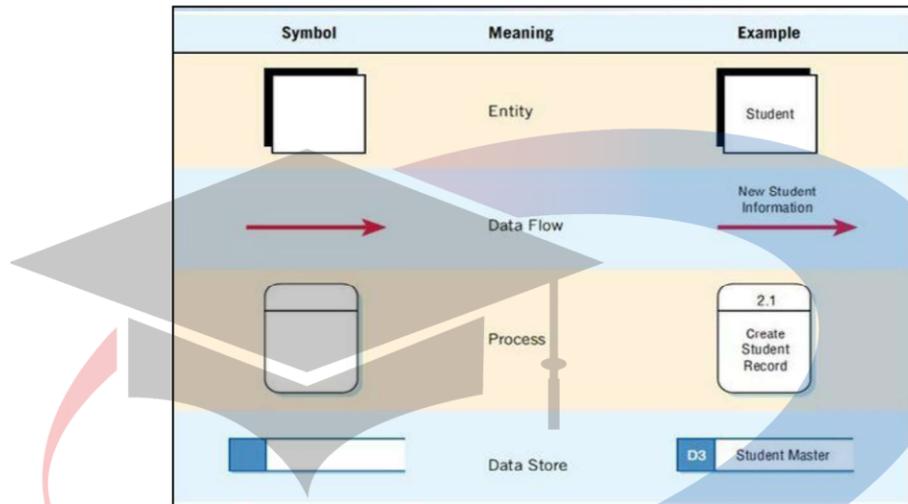
Diagram Aliran Data (DAD) / *Data Flow Diagram (DFD)* merupakan suatu teknik analisa data struktur yang dapat mempresentasikan proses-proses data. Pendekatan aliran data menekankan logika yang mendasari sistem. Pendekatan aliran data memiliki empat aliran utama melalui penjelasan naratif mengenai cara data-data berpindah di sepanjang sistem, yaitu [14]:

1. Kebebasan dari menjalankan implementasi teknis sistem yang terlalu dini.
2. Pemahaman lebih jauh mengenai keterkaitan satu sama lain dalam sistem dan subsistem.
3. Mengkomunikasikan pengetahuan sistem yang ada dengan pengguna melalui aliran data.
4. Menganalisis sistem yang diajukan untuk menentukan apakah data-data dan proses yang diperlukan sudah ditetapkan.

Dengan menggunakan kombinasi dari empat simbol, penganalisis sistem dapat menciptakan suatu gambaran proses-proses yang bisa menampilkan dokumentasi sistem yang solid. Empat simbol dasar yang digunakan untuk memetakan gerakan

diagram aliran data adalah kotak rangkap dua, tanda panah, bujur sangkar dengan sudut membulat, dan bujur sangkar dengan sudut membulat, dan bujur sangkar dengan ujung terbuka (tertutup pada sisi sebelah kiri dan terbuka pada sisi sebelah kanan) [14]

Keempat simbol dasar yang digunakan untuk menetapkan gerakan di data adalah sebagai berikut: [14]



Gambar 2. 5 Empat simbol dasar yang digunakan dalam Data Flow Diagram (DFD)

Empat simbol yang digunakan untuk memetakan gerakan Diagram aliran data, yaitu [14]:

1. Kontak Rangkap dua
Digunakan untuk menggambarkan suatu entitas eksternal (bagian lain, sebuah perusahaan, seseorang, atau sebuah mesin) yang dapat mengirim data atau menerima data dari sistem.
2. Bujur sangkar dengan sudut membulat
Digunakan untuk menunjukkan adanya proses transformasi proses-proses tersebut selalu menunjukkan suatu perubahan data, jadi aliran data yang meninggalkan suatu proses selalu diberi label yang berbeda dari aliran data yang masuk.
3. Bujur sangkar dengan ujung terbuka
Menunjukkan tempat penyimpanan untuk data-data yang memungkinkan penambahan dan perolehan data. Bujur sangkar yang digambarkan dengan dua

garis parallel yang tertutup oleh sebuah garis pendek disisi kiri dan ujungnya terbuka disisi sebelah kanan.

4. Tanda Panah

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

2.5.4 PIECES (Performance, Information, Economics, Control, Efficiency, Service)

Proses dan teknik yang digunakan oleh analis 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 [15].

Persyaratan sistem yang menetapkan apa yang harus dilakukan oleh sistem informasi sering disebut persyaratan fungsional. Persyaratan sistem yang menetapkan property atau kualitas yang harus dimiliki oleh sistem sering disebut persyaratan non fungsional [15].

Kerangka kerja 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 mengidentifikasi persyaratan sistem secara cermat [15].

Kategori dalam kerangka PIECES yaitu [15]:

- P : Kebutuhan untuk mengoreksi atau memperbaiki *performance* / performa.
- I : Kebutuhan untuk mengoreksi atau memperbaiki *information* / informasi.
- E : Kebutuhan untuk mengoreksi atau memperbaiki *economics* / ekonomi, mengendalikan biaya, atau meningkatkan keuntungan.
- C : Kebutuhan untuk mengoreksi atau memperbaiki *control* / kendali atau keamanan.

E : Kebutuhan untuk mengoreksi atau memperbaiki *efficiency* / efisiensi orang dan proses.

S : Kebutuhan untuk mengoreksi atau memperbaiki *service* / layanan ke pelanggan, pemasok, rekan kerja, karyawan, dan lain-lain.

Analisis PIECES ini sangat penting untuk dilakukan sebelum mengembangkan sebuah sistem informasi karena dalam analisis ini biasanya akan ditemukan beberapa masalah utama maupun masalah yang bersifat gejala dari masalah utama. Metode ini menggunakan enam variabel evaluasi yaitu:

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 handalkah suatu sistem informasi dalam berproses untuk menghasilkan tujuan yang diinginkan. Dalam hal ini kinerja diukur dari:

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

2.5.5 Kamus Data

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 (maksudnya metadata), suatu data yang disusun oleh penganalisis 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 [14].

Penganalisis sistem harus menunjuk pada item data yang sama. Kehati-hatian ini membantu mereka menghindari duplikasi, memungkinkan adanya komunikasi yang baik antara bagian-bagian organisasi yang saling berbagi basis data dan membuat upaya pemeliharaan lebih bermanfaat lagi. Kamus data juga bertindak sebagai standar tetap untuk elemen-elemen data [14].

Memahami proses penyusunan suatu kamus bisa membantu penganalisis sistem mengkonseptualisasikan sistem serta bagaimana cara kerjanya. Sebagai tambahan untuk dokumentasi serta mengurangi redundansi, kamus data bisa digunakan untuk [14]:

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.

Struktur data dari kamus data biasanya digambarkan menggunakan notasi aljabar. Metode ini memungkinkan penganalisis membuat suatu gambaran mengenai elemen-elemen yang membentuk struktur data bersama-sama dengan informasi-informasi mengenai elemen-elemen tersebut. Notasi aljabar menggunakan simbol-simbok sebagai berikut [14]:

1. Tanda sama dengan ($=$), artinya “terdiri dari”.
2. Tanda plus ($+$), artinya “dan”.
3. Tanda kurung $\{\}$, menunjukkan elemen-elemen repetitive, juga disebut kelompok berulang atau table-tabel. Kemungkinan bisa ada satu atau beberapa elemen berulang di dalam kelompok tersebut. Kelompok berulang bisa mengandung keadaan-keadaan tertentu, seperti misalnya, jumlah pengulangan.
4. Tanda kurung $[\]$, menunjukkan salah satu dari dua situasi tertentu. Satu elemen bisa ada sedangkan elemen lainnya juga ada, tetapi tidak bisa kedua-duanya ada secara bersamaan. Elemen-elemen yang ada di dalam tanda kurung ini saling terpisah satu sama lain.
5. Tanda kurung $()$, menunjukkan 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* numeric pada struktur file.

2.5.6 Normalisasi

Normalisasi adalah transformasi tujuan pemakai yang kompleks dan data tersimpan ke sekumpulan bagian-bagian struktur data yang kecil dan stabil. Di samping menjadi lebih sederhana dan lebih stabil, struktur data yang dinormalisasikan lebih mudah diatur daripada struktur data lainnya [14].

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 transisif manapun. Suatu ketergantungan transisif adalah keadaan dimana atribut bukan kunci tergantung pada atribut kunci lain [14].

Dalam normalisasi terdapat jenis-jenis normalisasi yaitu: [14]

Hubungan tidak normalisasi

↓ Tahap 1 Menghilangkan kelompok terulang

Hubungan Normalisasi (1NF)

↓ Tahap 2 Mengubah ketergantungan parsial

Hubungan bentuk normalisasi kedua (2NF)

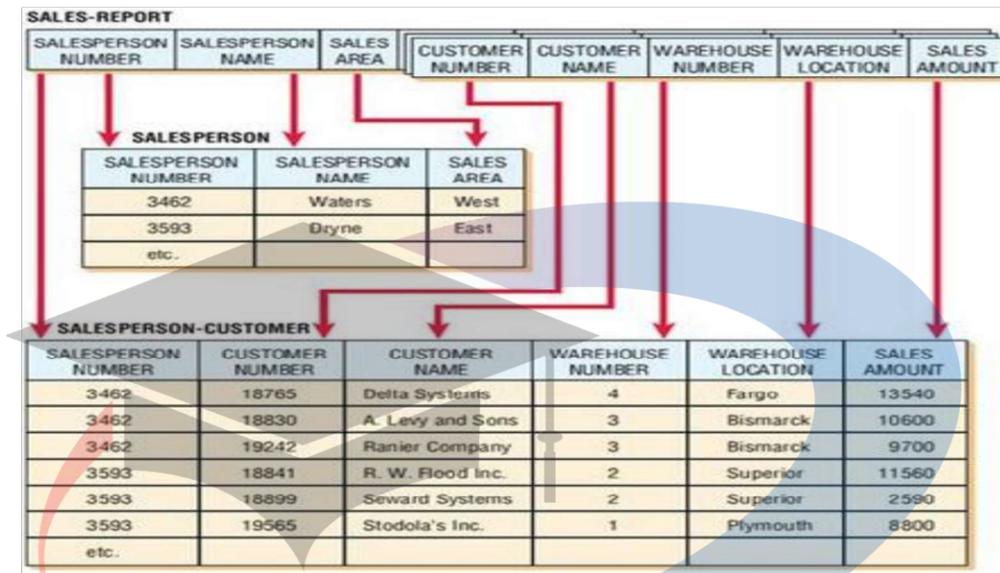
↓ Tahap 3 Mengubah ketergantungan transitif

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 mentransformasikan hubungan ke bentuk normalisasi ketiga.

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



Gambar 2. 6 Bentuk Normalisasi 1NF

2. Bentuk Normalisasi 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*):

