

BAB II TINJAUAN PUSTAKA

2.1. Sistem Informasi

Konsep sistem mendasari semua proses bisnis bidang sistem informasi. Sistem kebanyakan dapat didefinisikan secara sederhana sebagai kelompok elemen yang saling berhubungan atau berinteraksi sehingga membentuk satu kesatuan. [1]

Sistem adalah sekelompok komponen yang saling berhubungan, bekerja sama untuk mencapai tujuan bersama dengan menerima input serta menghasilkan *output* dalam proses transformasi yang teratur. Sistem seperti ini memiliki 3(tiga) komponen atau fungsi dasar yang berinteraksi:

1. **Input** melibatkan dan perakitan berbagai elemen yang memasuki sistem untuk di proses.
2. Pemrosesan melibatkan proses transformasi yang mengubah input menjadi *output*.
3. **Output** melibatkan perpindahan elemen yang telah diproduksi oleh proses transformasi ke tujuan akhirnya. [1]

Sumber dari informasi adalah data. Data adalah kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian – kejadian adalah sesuatu yang terjadi pada saat tertentu. Di dalam dunia bisnis, kejadian – kejadian yang terjadi adalah transaksi perubahan dari suatu nilai yang disebut transaksi. Kesatuan nyata adalah berupa suatu objek nyata seperti tempat, benda dan orang yang betul – betul ada atau terjadi. [1]

Data merupakan bentuk yang masih mentah, yang perlu diambil lebih lanjut melalui suatu metode untuk menghasilkan informasi dimana belum dapat bercerita banyak sehingga perlu diolah lebih lanjut. Data diolah melalui satu metode untuk menghasilkan informasi. Data dapat berbentuk simbol – simbol semacam huruf, angka bentuk suara, sinyal, gambar dan sebagainya. Data yang diolah itu diterapkan dalam sistem menjadi *input* yang berguna dalam suatu sistem. [1]

Informasi adalah sebagai data yang telah diubah menjadi konteks yang berarti dan berguna bagi para pemakai tertentu.

Sistem Informasi merupakan kombinasi teratur apapun dari orang-orang, hardware, software, jaringan komunikasi dan sumber daya data yang mengumpulkan, mengubah dan menyebarkan informasi dalam sebuah organisasi. Orang bergantung pada sistem informasi untuk berkomunikasi antara satu sama lain dengan menggunakan berbagai jenis alat fisik, perintah dan prosedur pemrosesan informasi (*Software*), saluran komunikasi (jaringan) dan data yang disimpan. [1]

Sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi. Sistem informasi adalah kumpulan perangkat keras dan perangkat lunak yang dirancang untuk mentransformasikan data ke dalam bentuk informasi yang berguna. Sistem informasi yang melayani tingkatan organisasi yang berbeda dibagi menjadi 4 (empat) tingkatan yaitu:

1. Sistem informasi tingkat operasional (*operational level – system*)

Sistem informasi yang memantau aktivitas dan kegiatan organisasi pada tingkat dasar.

2. Sistem informasi tingkat pengetahuan (*knowledge level- system*)

Sistem informasi yang mendukung pekerja berbasis pengetahuan (*knowledge level – system*) dan pekerja berbasis data (*data worker*) dalam suatu organisasi

3. Sistem informasi tingkat manajemen (*management level – system*)

Sistem informasi yang melayani kegiatan pengawasan, pengontrolan, pengambilan keputusan, dan administrasi bagi manajemen tingkat menengah.

4. Sistem informasi tingkat strategis (*strategic level – system*)

Sistem informasi yang mendukung aktivitas jangka panjang yang disusun oleh manajemen tingkat atas. [1]

2.2. Prototyping

Prototyping merupakan kegiatan membangun berskala kecil, representatif, atau fungsional dari persyaratan pemakai dalam rangka menemukan atau memeriksa

persyaratan tersebut. Dengan metode *prototyping*, pengembang dan umat dapat saling berinteraksi selama proses pembuatan sistem sehingga cepat dalam mendapatkan informasi yang dibutuhkan oleh umat. Metode ini secara keseluruhan akan mengacu kepada kepuasan umat [2].

Pembuatan *Prototype* dapat digunakan untuk aplikasi besar dan aplikasi kecil. *Prototype* aplikasi bisnis yang diperlukan oleh pemakai akhir dikembangkan secara cepat dengan menggunakan berbagai alat software pengembangan aplikasi. Kemudian sistem *Prototype* tersebut diperbaiki berkali-kali hingga dapat diterima oleh pengguna akhir [2].

Berikut ini adalah tahapan-tahapan dalam metodologi *prototyping* yang digunakan:

a. Analisis/Penyelidikan

Kegiatan yang dilakukan pada tahapan ini adalah mengidentifikasi kebutuhan, menyelidiki, menyatakan masalah, dan menilai kelayakan beberapa alternative solusi.

b. Analisis/Desain

Kegiatan yang dilakukan pada tahapan ini adalah mendesain dan menguji *Prototype* dari berbagai komponen sistem yang memenuhi kebutuhan pengguna akhir.

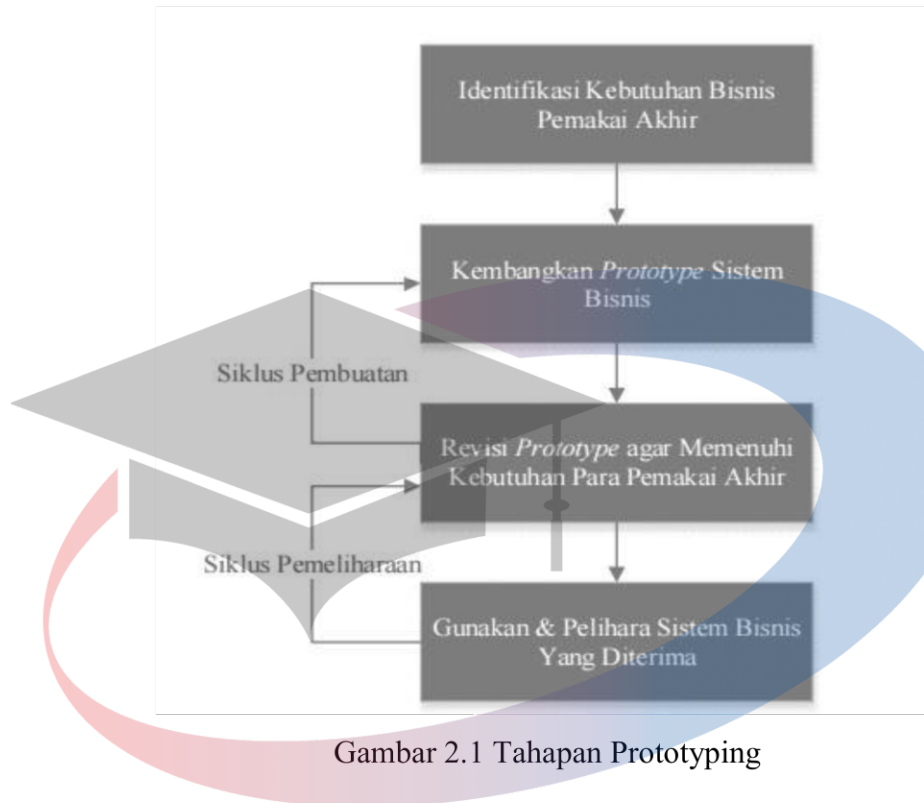
c. Desain/Implementasi

Pada tahapan ini prototipe sistem diuji, dievaluasi, dan dimodifikasi berulang-ulang hingga hasil akhir dari segi output dan umat interface dapat diterima.

d. Implementasi/Pemeliharaan

Pada tahapan ini, sistem yang diterima dapat dimodifikasi dengan mudah karena adanya dokumentasi [2].

Langkah-langkah prototyping diilustrasikan pada Gambar 2.1 di bawah berikut ini [2].



Gambar 2.1 Tahapan Prototyping

2.3. Teknik Pengembangan Sistem

Berikut adalah teknik pengembangan sistem yang akan dilakukan:

2.3.1. Data Flow Diagram (DFD)

Untuk membaca suatu *Data Flow Diagram* (DFD) terlebih dahulu harus memahami elemen-elemen yang menyusun suatu DFD. Ada 4 (empat) elemen yang menyusun sebuah DFD, yaitu:

1. Proses

Aktivitas atau fungsi yang dilakukan untuk alasan bisnis yang spesifik, biasa berupa manual maupun terkomputerisasi.

2. Aliran data

Satu data tunggal atau kumpulan logis suatu data, selalu diawali atau berakhir pada suatu proses.

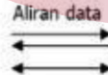
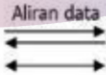
3. Simpanan Data

Kumpulan data yang disimpan dengan cara tertentu. Data yang mengalir disimpan dalam Simpanan data. Aliran data di-*update* atau ditambahkan ke simpanan data.

4. Entitas Eksternal

Entitas Eksternal dapat berupa orang, organisasi, atau sistem yang berada di luar sistem tetapi berinteraksi dengan system [3].

Masing-masing elemen akan diberi lambang tertentu untuk membedakan satu dengan yang lain. Ada beberapa metode untuk menggambarkan elemen-elemen tersebut. Untuk lebih jelasnya dapat dilihat pada gambar berikut [3]:

Gane/Sarson	Yourdon/De Marco	Keterangan
		Entitas eksternal, dapat berupa orang/unit terkait yang berinteraksi dengan sistem tetapi diluar sistem
		Orang, unit yang mempergunakan atau melakukan transformasi data. Komponen fisik tidak diidentifikasi.
		Aliran data dengan arah khusus dari sumber ke tujuan
		Penyimpanan data atau tempat data direfer oleh proses.

Gambar 2.2 Elemen DFD dan simbolnya

Dalam penggambaran DFD, terdapat beberapa tingkatan (level) penggambaran yang harus diperhatikan, yakni:

- 1 DFD Context Diagram: DFD pertama dalam proses bisnis Menunjukkan konteks di mana proses bisnis berada. Menunjukkan semua proses bisnis dalam satu proses tunggal (proses 0). *Context diagram* juga menunjukkan semua entitas luar yang menerima informasi dari atau memberikan informasi ke sistem.
- 2 DFD Level 0 Diagram: Menunjukkan semua proses utama yang menyusun keseluruhan sistem. Level ini juga menunjukkan komponen internal dari proses 0 dan menunjukkan bagaimana proses-proses utama direlasikan dengan menggunakan aliran data. Pada level ini juga ditunjukkan bagaimana proses-proses

utama terhubung dengan entitas eksternal. Pada level ini juga dilakukan penambahan simpanan data dalam penggambarannya.

- 3 DFD Level 1 Diagram: umumnya diagram level 1 diciptakan dari sistem proses utama dari level 0. Level ini menunjukkan proses-proses internal yang menyusun setiap proses-proses utama dalam level 0, sekaligus menunjukkan bagaimana informasi berpindah dari satu proses ke proses lainnya. Jika semisalnya proses untuk induk dipecah, katakanlah menjadi tiga proses anak, maka tiga proses anak ini harus secara utuh dapat menyusun proses induk.
- 4 DFD Level 2 Diagram: menunjukkan semua proses yang menyusun sebuah proses pada level 1. Bisa penyusunan DFD tidak mencapai level 2 ini. Atau mungkin harus dilanjutkan ke level berikutnya (level 3, level 4, dan seterusnya) [3].

2.3.2. Use Case Diagram

Use Case Diagram merupakan suatu bentuk diagram yang menggambarkan fungsionalitas yang diharapkan dari sistem dilihat dari perspektif pengguna di luar sistem, dimana merepresentasikan interaksi yang terjadi antara aktor dengan proses atau sistem yang dibuat. Interaksi antara *use case* dengan asosiasi berupa garis sederhana. Sebuah *use case* dapat berinteraksi dengan lebih dari satu aktor dan dapat berinteraksi dengan lebih dari satu *use case* [4].

Tujuan dari penggunaan *Use Case Diagram* adalah:

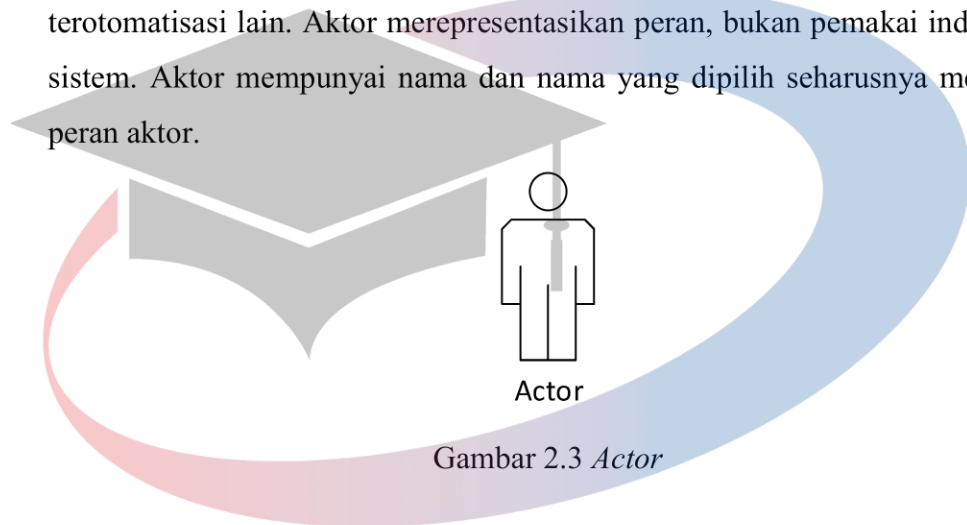
- 1 Mendapatkan pemahaman tentang sistem atau perangkat lunak yang akan dikembangkan.
- 2 Memperlihatkan hubungan – hubungan yang terjadi antara aktor (seseorang/sesuatu yang berinteraksi dengan sistem) dengan *use case* (proses yang terjadi dalam sistem).
- 3 Membantu dalam menyusun *requirement* sebuah sistem, mengkomunikasikan rancangan dengan klien dan merancang *test case* untuk semua fitur yang ada pada sistem.
- 4 Dengan melihat aktor – aktor, pengguna akan mengetahui siapa atau apa saja yang akan berinteraksi dengan sistem.

- 5 Dengan melihat kombinasi sejumlah aktor dan *use case*, pengguna akan mengetahui secara jelas ruang lingkup dari sistem atau perangkat lunak yang akan dikembangkan [4].

Komponen dalam use case terdiri dari:

1. *Actor*

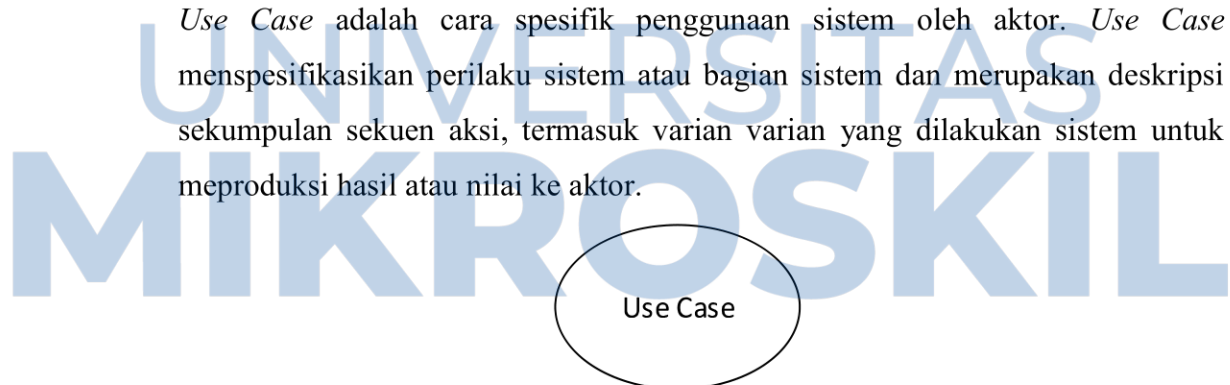
Aktor (*actor*) adalah pemakai sistem, dapat berupa manusia atau sistem terotomatisasi lain. Aktor merepresentasikan peran, bukan pemakai individu dari sistem. Aktor mempunyai nama dan nama yang dipilih seharusnya menyatakan peran aktor.



Gambar 2.3 *Actor*

2. *Use Case*

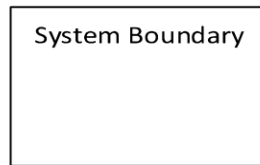
Use Case adalah cara spesifik penggunaan sistem oleh aktor. *Use Case* menspesifikasikan perilaku sistem atau bagian sistem dan merupakan deskripsi sekumpulan sekuen aksi, termasuk varian varian yang dilakukan sistem untuk memproduksi hasil atau nilai ke aktor.



Gambar 2.4 *Use Case*

3. *System Boundary*

System Boundary merupakan ruang lingkup sistem yang berisi *use-case* yang merupakan fungsional dari sistem tersebut dan antara sistem yang satu dengan sistem yang lain dimungkinkan saling berinteraksi. *System Boundary* dilambangkan dengan persegi panjang.

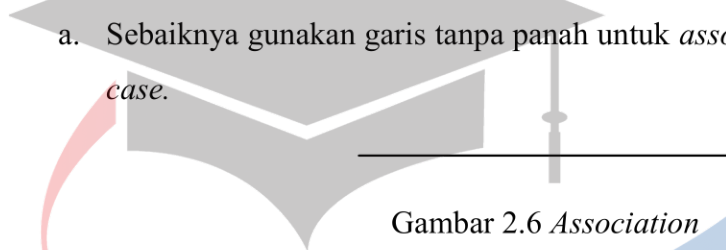


Gambar 2.5 *System Boundary*

4. *Association*

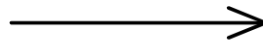
Association merupakan arah panah yang menunjukkan komunikasi antara aktor dengan *use case* dan siapa yang mengawali komunikasi *Association* dapat digambarkan dengan sebuah garis sebagai relasi asosiasi [4].

- a. Sebaiknya gunakan garis tanpa panah untuk *association* antara *actor* dan *use case*.



Gambar 2.6 *Association*

- b. *Association* antara *actor* dan *use case* yang menggunakan panah terbuka untuk mengindikasikan bila actor berinteraksi secara pasif dengan sistem.

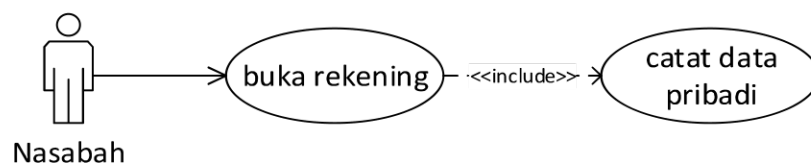


Gambar 2.7 *Association dengan Panah*

Pada *association* terdapat 4(empat) jenis yang bisa timbul pada *Use Case Diagram* yaitu [4]:

1. *Association antara actor dengan use case*

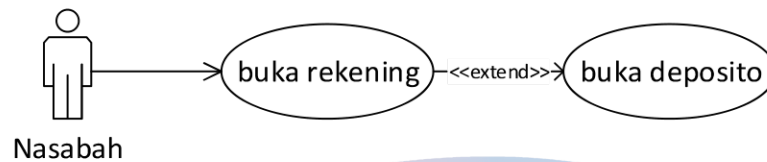
`<<include>>` termasuk di dalam *use case* lain required(diharuskan). Pemanggilan *use case* oleh *use case* lain, contohnya adalah pemanggilan sebuah fungsi program. Tanda panah terbuka harus terarah ke *sub use case*.



Gambar 2.8 *Association Include antara Actor dan Use Case*

2. *Association antara use case<<extend>>*

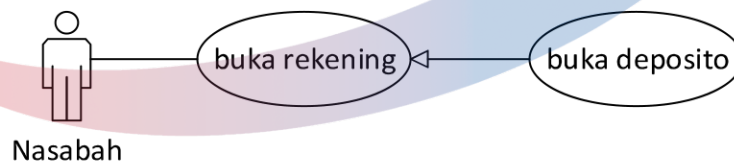
Perluasan dari *use case* lain jika kondisi atau syarat terpenuhi. Pemakaian *association* ini membuat diagram sulit dipahami dan tanda panah terbuka harus ke arah ke *parent/base use case*.



Gambar 2.9 *Association Extend antara Use Case*

3. *Generalization/inheritance antara use case*

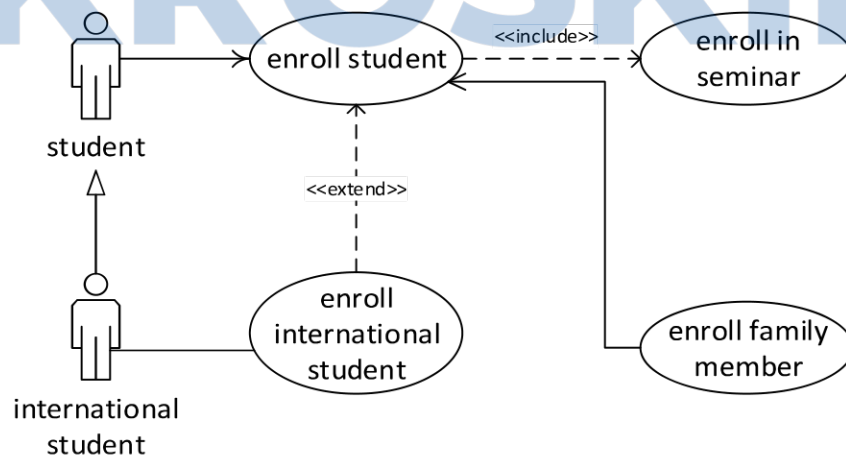
Generalization/inheritance digambarkan dengan sebuah garis panah tertutup pada salah satu ujungnya yang menunjukkan lebih umum. *Generalization/inheritance* dipakai ketika ada sebuah keadaan lain yang lain sendiri/perlakuan khusus (*single condition*).



Gambar 2.10 *Generalization/Inheritance antara Use Case*

4. *Generalization/inheritance antara actors*

Generalization/inheritance antara *actors* secara vertikal dengan *inheriting actor* dibawah base/parent *use case* [4].



Gambar 2.11 *Generalization/inheritance antara actors*

2.3.3. PIECES

PIECES menghasilkan suatu pelayanan publik yang berkualitas untuk suatu organisasi dalam hal ini harus mampu sejalan dengan perkembangan teknologi modern. Oleh karena itu dengan masuknya teknologi modernisasi yang berbasis komputerisasi atau internet maka kinerja organisasi dapat berjalan dengan lebih optimal sehingga pelayanan untuk publik atau *umat* pun dapat terpenuhi dengan baik [3].

Analisis PIECES yaitu *Performance, Information, Economy, Control, Efficiency*, dan *Service* merupakan teknik untuk mengidentifikasi dan memecahkan permasalahan yang terjadi pada sistem informasi [3].

1. Analisis Kinerja (*Performance*)

Masalah kinerja terjadi ketika tugas-tugas bisnis yang dijalankan tidak mencapai sasaran. Kinerja diukur dengan jumlah produksi dan waktu tanggap. Jumlah produksi adalah jumlah pekerjaan yang bisa diselesaikan selama jangka waktu tertentu. Pada bagian pemasaran, kinerja diukur berdasarkan *volume* pekerjaan, pangsa pasar yang diraih atau citra perusahaan. Waktu tanggap adalah keterlambatan rata-rata antara suatu transaksi dengan tanggapan yang diberikan kepada transaksi tersebut.

2. Analisis Informasi (*Information*)

Evaluasi kemampuan sistem informasi dalam menghasilkan informasi atau data yang bermanfaat untuk menyikapi peluang dalam menangani masalah yang muncul. Situasi dalam analisis informasi ini meliputi:

- a. Kurangnya informasi mengenai keputusan atau situasi yang sekarang.
- b. Kurangnya informasi yang relevan mengenai keputusan ataupun situasi sekarang.
- c. Kurangnya informasi yang tepat waktu.
- d. Terlalu banyak informasi.
- e. Informasi tidak akurat.

Informasi juga dapat merupakan fokus dari suatu batasan atau kebijakan. Sementara analisis informasi memeriksa *output* sistem, analisis data meneliti data yang tersimpan dalam sebuah sistem. Permasalahan yang dihadapi meliputi:

- a. Data yang berlebihan. Data yang sama ditangkap atau disimpan di banyak tempat.
- b. Kelakuan data. Data ditangkap dan disimpan, tetapi diorganisasikan sedemikian rupa sehingga laporan dan pengujian tidak dapat atau sulit dilakukan.

3. Analisis Ekonomi (*Economy*)

Ekonomi barangkali merupakan motivasi paling umum bagi suatu proyek. Pijakan dasar bagi kebanyakan manajer adalah biaya atau rupiah. Persoalan ekonomis dan peluang berkaitan dengan masalah biaya. Adapun hal-hal yang perlu diperhatikan dapat disimak berikut:

- a. Biaya
 - Biaya tidak diketahui.
 - Biaya tidak dapat dilacak ke sumber.
 - Biaya terlalu tinggi.
- b. Keuntungan
 - Pasar-pasar baru dapat dieksplorasi.
 - Pemasaran saat ini dapat diperbaiki.
 - Pesanan-pesanan dapat ditingkatkan.

4. Analisis Keamanan (*Control*)

Sistem keamanan yang digunakan harus dapat mengamankan data dari kerusakan. Misalnya dengan memback-up data. Selain itu sistem keamanan juga harus dapat mengamankan data dari akses yang tidak diizinkan. Adapun hal-hal yang perlu diperhatikan adalah:

1. Keamanan atau kontrol yang lemah
 - a. Input data tidak diedit dengan cukup.
 - b. Kejahatan (misalnya: penggelapan atau pencurian) terhadap data.
 - c. Pelanggaran etika pada data atau informasi.
2. Kontrol atau keamanan berlebihan
 - a. Prosedur birokratis memperlamban sistem.
 - b. Pengendalian yang berlebihan mengganggu para pelanggan atau karyawan.
 - c. Pengendalian berlebihan menyebabkan penundaan pemrosesan.

5. Analisis Efisiensi (*Efficiency*)

Sumber daya yang ada guna meminimalkan pemborosan. Efisiensi dari sistem yang dikembangkan adalah pemakaian secara maksimal terhadap sumberdaya infrastruktur dan sumberdaya manusia. Serta efisiensi juga menganalisis keterlambatan pengolahan data yang terjadi.

Berikut adalah indikasi bahwa suatu sistem dapat dikatakan tidak efisien:

- a. Banyak waktu yang tebuang pada aktivitas sumber daya manusia, mesin atau komputer.
 - b. Data diinput atau disalin secara berlebihan.
 - c. Data diproses secara berlebihan.
 - d. Informasi dihasilkan secara berlebihan.
 - e. Usaha yang dibutuhkan untuk tugas-tugas terlalu berlebihan.
 - f. Material yang dibutuhkan untuk tugas-tugas terlalu berlebihan.
6. Layanan (*Service*)

Mengkoordinasikan aktifitas dalam pelayanan yang ingin dicapai sehingga tujuan dan sasaran pelayanan dapat tercapai. Berikut adalah beberapa kriteria penilaian dimana kualitas suatu sistem bisa dikatakan buruk:

- a. Sistem menghasilkan produk yang tidak akurat.
- b. Sistem menghasilkan produk yang tidak konsisten.
- c. Sistem menghasilkan produk yang tidak dipercaya.
- d. Sistem tidak mudah dipelajari.
- e. Sistem tidak mudah digunakan.
- f. Sistem canggung untuk digunakan.
- g. Sistem tidak fleksibel. [3].

2.4. Basis Data

Basis data merupakan kumpulan *file*. Selain itu basis data adalah sumber data yang caranya dipakai oleh banyak pemakai untuk berbagai aplikasi. Inti dari sumber basis data adalah *Database Management System* (DBMS), yang memperbolehkan pembuatan modifikasi dan pembaharuan basis data pembuatan laporan. Tujuan basis data yang efektif adalah sebagai berikut:

- 1) Memastikan bahwa data dapat dipakai di antara pemakai untuk berbagai aplikasi.
- 2) Memelihara data baik keakuratannya maupun tingkat konsistensinya.
- 3) Memastikan semua data yang diperlukan untuk aplikasi sekarang dan yang akan datang dapat disediakan dengan cepat.
- 4) Memperbolehkan basis data untuk membangun pandangan personalnya tentang data tanpa memperhatikan cara penyimpanan data secara fisik [5].

Terdapat 3(tiga) jenis basis data yang terstruktur antara lain:

1) Struktur Data Hierarki

Struktur data hierarki menyatakan bahwa sebuah entitas data tidak memiliki lebih dari satu entitas pribadi. Oleh karena itu, struktur data merupakan struktur hubungan banyak ke satu, satu ke banyak, atau satu ke satu. Struktur data hierarki kadang-kadang disebut dengan pohon karena sub koordinatnya dihubungkan ke pemilik entitas yang mempunyai cabang pohon.

2) Struktur Data Jaringan

Suatu struktur jaringan membolehkan entitas apapun untuk memiliki sejumlah sub koordinat atau superior. Entitas dihubungkan dengan menggunakan *link* jaringan yang merupakan item data biasa untuk kedua entitas terhubung. Beberapa kelemahan dari struktur hierarki dapat dikurangi dengan menggunakan struktur jaringan yang lebih rumit.

3) Struktur Data Relasional.

Suatu struktur relasional terdiri atas satu atau lebih tabel dua dimensi, yang dipandang sebagai hubungan (*relation*). Basis data pada *tabel* mewakili *record* dan kolom memuat atribut [5].

2.5. Administrasi

Pengertian administrasi oleh Van Der Schroeff adalah merupakan seluruh himpunan catatan-catatan mengenai perusahaan dan peristiwa-peristiwa perusahaan untuk keperluan pimpinan dan penyelenggaraan perusahaan [6].

Administrasi adalah keseluruhan proses kerjasama antara dua orang manusia atau lebih yang didasarkan pada rasionalitas tertentu untuk mencapai tujuan yang telah ditentukan sebelumnya [7].

Administrasi secara sempit berasal dari kata *Administratie* (Bahasa Belanda) yaitu meliputi kegiatan catat-mencatat, surat-menyurat, pembukuan ringan, ketik-mengetik, agenda dan sebagainya yang bersifat teknis ketatausahaan. Administrasi dapat diartikan sebagai upaya memanfaatkan sumberdaya (3M: *Man*, *Money*, dan *Material*) dimiliki secara efektif dan efisien untuk mencapai tujuan bersama yang telah ditetapkan. Ungkapan yang sederhana ini tidak dapat menjelaskan membutuhkan pengetahuan yang menyeluruh [8].

Administrasi yaitu *We consider administrasion to be slightly more encompassing than management* Ketika kita berbicara administrasi, berarti di dalamnya berbicara mengenai manajemen [8].

Newman mengatakan pengertian administrasi sebagai bimbingan, kepemimpinan dan pengawasan usaha kelompok individu guna mencapai tujuan bersama [9].

Berdasarkan hal tersebut diatas, administrasi ialah proses penyelenggara kerja yang dilakukan bersama-sama untuk mencapai tujuan yang telah ditetapkan. Administrasi baik dalam pengertian luas maupun sempit di dalam penyelenggara diwujudkan melalui fungsi-fungsi manajemen, yang terdiri dari perencanaan, pengorganisasian pelaksanaan dan pengawasan.