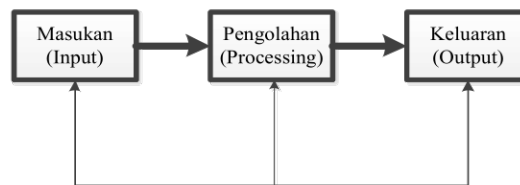


BAB II TINJAUAN PUSTAKA

2.1 Konsep Sistem Informasi

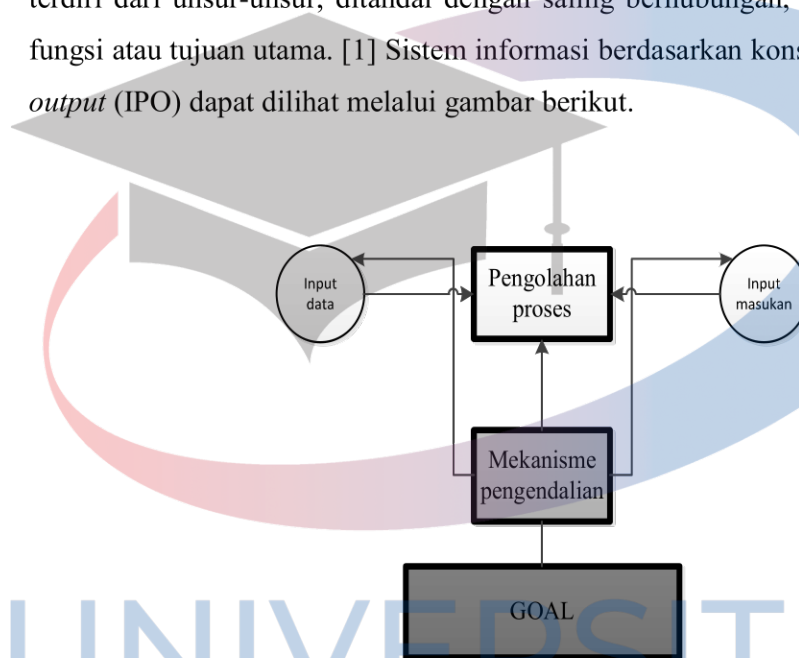
Dalam memahami pengertian sistem informasi, dapat dilihat dengan melalui keterkaitan antara data dan informasi sebagai entitas dari pembentuk sistem informasi. Data tersebut merupakan nilai, keadaan, atau sifat yang dapat berdiri sendiri dan lepas dari konteks apapun sehingga informasi adalah data yang telah diproses, dan memiliki arti. [1] Data adalah deskripsi tentang benda, kejadian, aktivitas, dan transaksi yang tidak mempunyai makna atau bisa dikatakan tidak berpengaruh secara langsung kepada pemakai. Data tersebut dapat berupa nilai yang terformat, teks, citra, audio, dan video. [1]

Melalui proses transformasi data dapat dibuat bermakna menjadi bahan mentah informasi. Dengan demikian informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau mendatang. Jadi, untuk membedakan informasi dan data terletak pada makna, biasanya informasi itu bermakna sedangkan untuk data tidak memiliki makna. [2] Sedangkan sistem adalah sekumpulan elemen yang saling terkait atau terpadu untuk mencapai suatu tujuan. Dengan demikian, secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung sama lain. [1] Di dalam sistem terdapat beberapa elemen berupa tujuan, masukan (*input*), keluaran (*output*), proses (perubahan atau *transformasi*), mekanisme pengendalian (*control mechanism*), umpan balik (*feed back*) dan juga sistem berinteraksi dengan lingkungan dan juga memiliki batas (*boundary*). [1]



Gambar 2.1 Model Sistem [1]

Sistem informasi sebetulnya tidak harus melibatkan komputer, karena sistem informasi yang menggunakan komputer biasanya disebut sistem informasi berbasis komputer (*Computer-Based Information Systems* atau CBIS). [2][3] Dengan demikian sistem informasi merupakan kumpulan perangkat keras, dan perangkat lunak serta manusia yang akan mengolah dan menggunakan perangkat tersebut. [4] Sedangkan sistem menurut Gaspert merupakan operasi sistem dalam lingkungan, terdiri dari unsur-unsur, ditandai dengan saling berhubungan, dan mempunyai satu fungsi atau tujuan utama. [1] Sistem informasi berdasarkan konsep *input, processing, output* (IPO) dapat dilihat melalui gambar berikut.



Gambar 2.2 Konsep Sistem Informasi [1]

2.2 Analisis dan Perancangan Sistem

Analisis sistem merupakan defenisi tentang bagaimana cara untuk memahami dan memberikan spesifikasi dengan secara detail tentang apa yang harus dilakukan oleh sistem, sedangkan sistem desain diartikan sebagai penjelasan dengan detail tentang bagaimana bagian-bagian dari sistem informasi itu dapat diimplementasikan. Dengan demikian, analisis dan desain sistem informasi (ANSI) bisa didefinisikan sebagai proses organisasional yang kompleks dimana sistem informasi berbasis komputer dapat diimplementasikan atau diringkas sebagai analisis sebagai defenisi masalah, dan *design* sebagai memecahkan masalah. [1]

Urutan tahapan ANSI sangat penting dengan beberapa alasan yang sangat spesifik. Pertama, kesuksesan suatu sistem informasi dapat tergantung pada analisis dan perancangan yang baik. Tahapan analisis ini akan menentukan masalah apa yang harus diselesaikan pada sebuah organisasi atau perusahaan. Bila terjadi kesalahan pada tahap ini akan mengakibatkan masalah tetap ada walaupun sistem informasi tersebut telah diimplementasikan. Sementara tahapan desain akan sangat menentukan seperti apa sistem untuk dapat berfungsi. Walaupun demikian, pada tahapan analisis masalah utama sudah dapat dikatakan dengan benar, namun pada kesalahan desain akan mengakibatkan kegagalan penyelesaian oleh sistem komputer. [1]

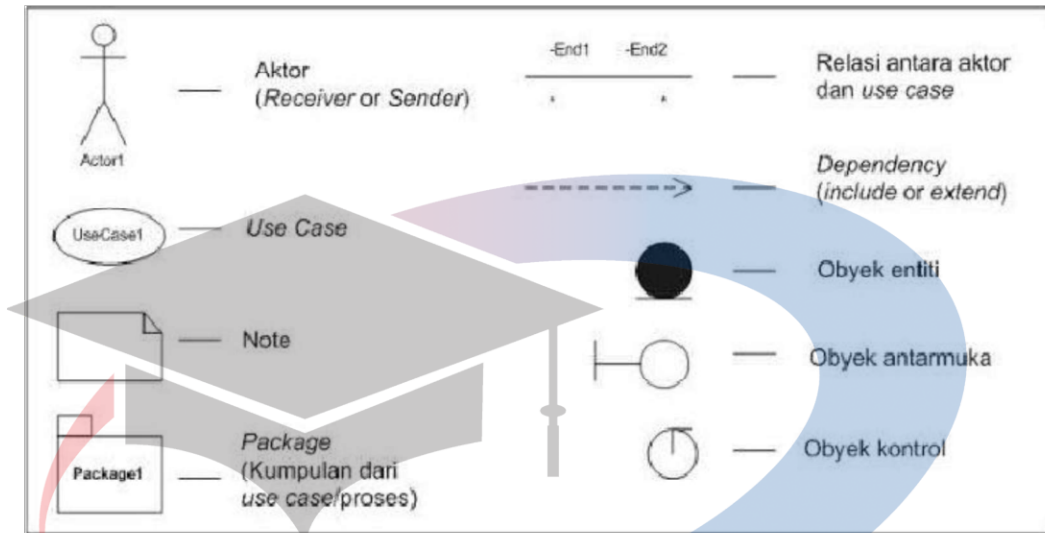
Tahapan analisis merupakan tahapan dimana sistem yang sedang berjalan dapat dipelajari dan pergantian sistem dapat diusulkan. Pada tahapan ini dapat dideskripsikan dengan sistem yang sedang berjalan, masalah, dan kesempatan didefinisikan, serta rekomendasi umum untuk perbaikan. Enam hal utama dalam pengetahuan tahapan analisis yaitu pengumpulan, pendefinisian sistem *requirement*, prioritaskan kebutuhan, penyusunan, evaluasi alternatif, dan mengetahui ulasan tentang kebutuhan dengan pihak pada manajemen. [1]

2.3 Use Case Diagram

Use case diagram merupakan gambaran graphical dari beberapa atau semua actor sehingga *use case* dan interaksi diantara komponen-komponennya dapat memperkenalkan suatu sistem yang akan dibangun. [5] *Use case* memiliki dua istilah umum yaitu *system use case* yang berinteraksi dengan sistem dan *business use case* yang berinteraksi dengan bisnis dan konsumen atau kejadian nyata. [6]

Use case bekerja untuk mendeskripsikan tipikal interaksi antara user (pengguna) dengan sistem melalui cerita untuk sistem aplikasi, sehingga menjelaskan urutan langkah-langkah yang menerangkan antara pengguna dan sistem disebut scenario. [6] *Use case diagram* merupakan penetapan yang seharusnya dikerjakan oleh sistem, yaitu kebutuhan fungsional sistem dan tidak untuk menentukan kebutuhan non fungsional seperti sasaran kinerja, bahasa pemrograman dan lain sebagainya. [5]

Karakteristik *use case* memiliki interaksi antara sistem dan actor, sehingga actor tersebut melibatkan peran actor lainnya.[14] Berikut model penggambaran sistem *use case* :



Gambar 2.3 Pemodelan Use Case [14]

Proses pemodelan *Use-Case* persyaratan digunakan untuk mendapatkan dan menganalisis informasi persyaratan yang cukup dalam menentukan persiapan model yang mengkomunikasikan apa yang diperlukan oleh pengguna. [7] Langkah-langkah yang diperlukan untuk menghasilkan model ini [7]:

1. Mengidentifikasi pelaku bisnis
2. Mengidentifikasi *use-case* persyaratan bisnis
3. Membuat diagram model *use-case*
4. Mendokumentasikan naratif *use-case* persyaratan bisnis

Hubungan digambarkan sebagai garis antara dua symbol pada diagram *use-case*, terdiri atas [7]:

1. *Asosiasi* adalah hubungan satu pelaku dan satu *use-case*.
2. *Extend* adalah hubungan antara extension *use-case* dan *use-case* yang diperluas.
3. *Uses relationship* adalah hubungan antara abstract *use-case* dan *use-case* yang menggunakannya.
4. *Inheritance relationship* terjadi pada saat seorang pelaku mewarisi kemampuan dari *use case* lain untuk menginisiasi suatu *use-case*.
5. *Depend-on relationship* mengindikasikan ketergantungan antara *use-case*.

Salah satu tantangan bagi semua tim pengembangan sistem informasi, dan khususnya analisis sistem adalah kemampuan untuk memperoleh persyaratan sistem yang benar dan yang diperlukan dari para *stakeholder* dan menetakannya dalam sebuah cara yang dapat dipahami para *stakeholder* agar persyaratan-persyaratan itu dapat diverifikasi dan divalidasi. Dalam kenyataannya, hal ini terus menjadi masalah selama bertahun-tahun, seperti dijelaskan oleh *Fred Brooks* dalam tulisannya yang terkenal pada tahun 1987. [8]

Bagian terberat dalam pembangunan sistem perangkat lunak adalah memutuskan dengan tepat apa yang akan dibangun. Tidak ada bagian lain dari kerja konseptual yang sesulit menentukan persyaratan teknis yang terinci, termasuk semua antarmuka untuk orang, mesin, dan sistem perangkat lunak yang lain. Tidak ada pekerjaan lain yang demikian berpengaruh terhadap sistem yang dihasilkan, seandainya dilakukan dengan salah. Tidak ada bagian lain yang lebih sulit dari bagian ini untuk diralat kemudian. [8]

Komunitas teknologi informasi selalu bermasalah dalam menentukan persyaratan, khususnya persyaratan fungsional untuk pengguna. Dalam bahasan terdahulu telah menggunakan berbagai alat seperti model data, model proses, *prototype*, dan spesifikasi persyaratan yang dipahami dan cukup nyaman untuk digunakan, tapi tampaknya alat-alat tersebut sulit dipahami oleh semua pengguna yang tidak mendapatkan pendidikan dibidang praktik pengembangan perangkat lunak. Oleh sebab itu, banyak proyek pengembangan masih terganggu dengan adanya lingkup yang tidak jelas, pembengkakan, dan masalah keterlambatan jadwal. Sangat sering sistem yang dikembangkan dan disebarluaskan sungguh-sungguh tidak memuaskan kebutuhan para pengguna. Beberapa ditanggihkan dan tidak digunakan sama sekali, dan dalam persentase yang lebih besar dibatalkan bahkan sebelum usaha pengembangan itu selesai. Sebuah perusahaan riset yang sangat terkenal, *The Standish Group*, mempelajari 23.000 aplikasi TI pada tahun 1994, 1996, dan 1998. Penelitian tahun 1998 menemukan lebih dari seperempat proyek pada tahun 1998 mengalami sukses (sesuai anggaran, tepat waktu, dan memasukkan semua fitur), dan lebih dari seperempatnya mengalami kegagalan (dibatalkan sebelum selesai). Penelitian itu juga menemukan bahwa hampir setengah dari jumlah proyek tahun itu, oleh *Standish* dinilai sebagai proyek yang menantang – selesai dan beroperasi, tapi

selesai dengan biaya melebihi anggaran, melewati waktu yang ditentukan, atau tanpa semua *fitur* yang ditentukan oleh pengguna. Nilai positif yang digambarkan dalam penelitian tersebut dan juga penelitian lainnya adalah cara dan alat yang digunakan untuk mengembangkan sistem informasi mengalami peningkatan. Industri pengembangan perangkat lunak telah mempelajari bahwa untuk meraih sukses perencanaan, analisis, desain, konstruksi, dan penyebaran sistem informasi, analisis sistem pertama-tama harus memahami kebutuhan para *stakeholder* dan alasan mengapa sistem harus dikembangkan – konsep yang disebut *user-centered development*/pengembangan berpusatkan pengguna. Dengan fokus kepada pengguna sistem, analisis dapat berkonsentrasi untuk mengembangkan bagaimana sistem akan digunakan dan bukan pada bagaimana sistem dibangun. *Use case modeling*/pemodelan *use case* adalah sebuah pendekatan yang memfasilitasi pengembangan berpusatkan kegunaan. [8]

Pemodelan *use case* berakar pada pemodelan berorientasi-objek, dan akan mempelajari lebih banyak bagaimana mengaplikasikan pemodelan *use case* dalam bab-bab tentang desain dan analisis berorientasi-objek. Akan tetapi, pemodelan *use case* terkenal di lingkungan pengembangan non-objek. Pada bab-bab selanjutnya, akan mempelajari bagaimana pemodelan *use case* melengkapi analisis sistem tradisional dan alat-alat desain seperti pemodelan data dan pemodelan proses; juga bagaimana pemodelan tersebut memberikan satu basis bagi keputusan arsitektural dan keputusan desain antarmuka pengguna. [8]

Pemodelan *use case* awalnya disusun oleh Dr. Ivar Jacobson pada tahun 1986 dan menjadi populer setelah beliau menerbitkan buku, *Object-Oriented Software Engineering*, pada tahun 1992. Dr. Jacobson menggunakan pemodelan *use case* sebagai kerangka kerja untuk metodologi *object-oriented*-nya yang dengan sukses digunakannya untuk mengembangkan sistem informasi berorientasi-objek. 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. [8]

Penggunaan pemodelan *use case* memfasilitasi dan mendorong keterlibatan pengguna, yang merupakan faktor sukses kritis untuk memastikan sukses proyek. Sebagai tambahan, pemodelan *use case* memberikan manfaat berikut [8]:

1. Menyediakan *tool* untuk meng-*capture* persyaratan fungsional.
2. Membantu menyusun ulang lingkup sistem menjadi bagian-bagian yang lebih dapat 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, terutama pengembangan *incremental* dan iteratif.
5. Menyajikan panduan untuk mengestimasi lingkup, usaha, dan jadwal proyek.
6. Menyajikan garis pokok pengujian, khususnya menentukan rencana tes atau *test case*.
7. Mengajukan garis pokok bagi *help system* dan manual pengguna, dan juga dokumentasi pengembangan sistem.
8. Menyajikan *tool* untuk melacak persyaratan.
9. Menyajikan titik mulai/awal untuk identifikasi objek data atau entitas.
10. Menyajikan spesifikasi fungsional untuk mendesain antarmuka pengguna dan sistem.
11. Menyajikan alat untuk menentukan persyaratan akses *database* dalam hal menambah, mengubah, menghapus, dan membaca.
12. Menyajikan kerangka kerja untuk mengarahkan proyek pengembangan sistem.

2.4 Basis Data

Data merupakan fakta mengenai objek seperti konsep, benda, peristiwa yang dapat dicatat dan mempunyai arti secara implisit. Data tersebut dapat dinyatakan dalam bentuk angka, karakter dan simbol, sehingga bila dikumpulkan akan memiliki saling keterhubungan yang disebut basis data (*database*). [9] *Database* (basis data) adalah kumpulan informasi yang berhubungan dengan suatu objek atau tujuan khusus. Dengan adanya *database* maka seluruh informasi dapat diorganisasikan

dengan baik dan disimpan secara terpisah dalam ruangan penyimpanan yang disebut tabel. [10] Basis data tersebut dapat digunakan untuk mengatasi problem pada sistem yang memakai pendekatan berbasis berkas, sehingga dapat dijelaskan basis data merupakan sekumpulan data yang saling terkait dalam memudahkan memperoleh suatu informasi. [3]

DBMS memiliki bahasa untuk mengelola dan mengorganisasikan, yaitu bahasa definisi data (*Data Defenition Language* atau *DDL*) yang digunakan *administrator* dalam mendefenisikan skema dan subskema basis data, dan bahasa manipulasi data (*Data Manipulation Language* atau *DML*) yang digunakan pemakai untuk menentukan apa dan bagaimana cara mendapatkan data. Dalam mengelola basis data diperlukan perangkat lunak yang disebut *Database Management System (DBMS)*. *DBMS* adalah perangkat lunak sistem yang memungkinkan para pengguna membuat, memelihara, mengontrol dan mengakses basis data dengan cara praktis dan efisien. Umumnya *DBMS* menyediakan fitur independensi data-program, keamanan, integritas, konkurensi, pemulihan (*recovery*), katalog sistem, perangkat produktivitas. Dalam hal komponen *DBMS* terdiri atas perangkat keras berupa PC, perangkat lunak dalam program aplikasi (*Visual Basic, C++*), data untuk memperoleh informasi, prosedur dengan merancang penggunaan basis data, orang sebagai pemakai, pemrogram aplikasi dan *administrator* basis data. [3]

DBMS dapat diakses secara efisien tanpa mengharuskan pemakai tahu detail tentang cara data dapat disimpan sehingga dibutuhkan *arsitektur basis data*. Pada tahun 1975 ANSI-SPARC (*American National Standards Institute – Standards Planning and Requirements Committee*) mendefenisikan arsitektur 3 level yaitu level eksternal yang berhubungan secara langsung dengan pemakai, level konseptual yang biasa dipakai oleh administrator, level internal yang berhubungan secara langsung dengan basis data (alokasi penyimpanan, dekripsi, kompresi data). [3]

Di dalam sebuah *disk*, basis data dapat diciptakan dan dapat pula ditiadakan. Di dalam sebuah *disk*, dapat pula menempatkan beberapa (lebih dari satu) basis data. Sementara dalam sebuah basis data, dapat menempatkan satu atau lebih *file*/tabel. Pada *file*/tabel inilah sesungguhnya data disimpan/ditempatkan. Setiap basis data umumnya dibuat untuk mewakili sebuah semesta data yang spesifik. Misalnya, ada basis data kepegawaian, basis data akademik, basis data inventori (pergudangan), dan

sebagainya. Sementara dalam basis data akademik, misalnya, dapat menempatkan *file* mahasiswa, *file* mata kuliah, *file* dosen, *file* jadwal, *file* kehadiran, *file* nilai, dan seterusnya. [11] Karena itu, operasi-operasi dasar yang dapat dilakukan dengan basis data dapat meliputi [11]:

1. Pembuatan basis data baru (*create database*), yang identik dengan pembuatan lemari arsip yang baru.
2. Penghapusan basis data (*drop database*), yang identik dengan perusakan lemari arsip (sekaligus beserta isinya, jika ada).
3. Pembuatan *file*/tabel baru ke suatu basis data (*create table*), yang identik dengan penambahan *map* arsip baru ke sebuah lemari arsip yang telah ada.
4. Penghapusan *file*/tabel dari suatu basis data (*drop table*), yang identik dengan perusakan *map* arsip lama yang ada di sebuah lemari arsip.
5. Penambahan/pengisian data baru ke sebuah *file*/tabel di sebuah basis data (*insert*), yang identik dengan penambahan lembaran arsip ke sebuah *map* arsip.
6. Pengambilan data dari sebuah *file*/tabel (*retrieve/search*), yang identik dengan pencarian lembaran arsip dari sebuah *map* arsip.
7. Pengubahan data dari sebuah *file*/tabel (*update*), yang identik dengan perbaikan isi lembaran arsip yang ada di sebuah *map* arsip.
8. Penghapusan data dari sebuah *file*/tabel (*delete*), yang identik dengan penghapusan sebuah lembaran arsip yang ada di sebuah *map* arsip.

Operasi yang berkenaan dengan pembuatan objek (basis data dan tabel) merupakan operasi awal yang hanya dilakukan sekali dan berlaku seterusnya. Sedangkan operasi-operasi yang berkaitan dengan isi tabel (data) merupakan operasi rutin yang akan berlangsung berulang-ulang dan karena itu operasi-operasi inilah yang lebih tepat mewakili aktivitas pengelolaan (*management*) dan pengolahan (*processing*) data dalam basis data. [11]

2.5 Website

Internet merupakan sekumpulan jaringan yang terhubung satu dengan yang lainnya, dimana jaringan menjadikan sambungan menuju global informasi. [5] *Internet* tidak ada yang memiliki, tetapi memiliki lembaga yang memonitor dan mengurus internet. Lembaga tersebut bernama *the internet society* (www.isoc.org)

yang merupakan grup *nonprofit* yang berdiri tahun 1992 dan membuat aturan protocol yang digunakan pada internet. [12] *Web* merupakan bagian terpenting dalam *internet*. Dengan menggunakan web dapat mencapai dunia tanpa batas dengan izin mengakses dan berinteraksi dengan teks, *grafik*, animasi, foto, suara, dan video. Dengan kata lain, *web* secara fisik adalah kumpulan komputer pribadi, *web browser*, *koneksi*, *computer server*, *router*, dan *switch* yang digunakan untuk mengalirkan informasi. [12]

Website adalah kumpulan dari halaman-halaman yang berhubungan dengan *web* atau *file-file* lain yang saling terkait. Di dalam sebuah *website* terdapat satu halaman yang dikenal dengan sebutan *home-page*. *Homepage* ini adalah halaman yang menjadi halaman pertama kali dapat dilihat dalam *website* ketika seseorang mengunjungi *website*. [5] *Website* tersebut merupakan suatu koleksi dokumen *HTML* atau perusahaan yang memuat informasi dalam *web server* (suatu unit komputer yang berfungsi untuk menyimpan informasi dan untuk mengelola jaringan komputer) untuk fasilitas *world wide web* atau *web*, dan dapat diakses oleh seluruh pemakai *internet*). [13]

UNIVERSITAS MIKROSKIL