

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Konsep Sistem Informasi**

##### **2.1.1 Sistem**

Pengertian sistem secara umum yaitu:

1. Setiap sistem terdiri dari beberapa unsur.
2. Unsur-unsur tersebut merupakan bagian yang tidak terpisahkan dari sistem yang bersangkutan.
3. Unsur-unsur didalam sistem tersebut bekerja sama untuk mencapai tujuan sistem.
4. Suatu sistem merupakan bagian dari sistem lain yang lebih besar [14].

Sistem adalah sekumpulan komponen yang saling berhubungan, bekerja sama untuk mencapai tujuan dengan menerima masukan serta menghasilkan keluaran dalam proses transformasi yang teratur [15]. Dari definisi sistem diatas, dapat diketahui bahwa sistem adalah kumpulan komponen-komponen yang saling berhubungan satu dengan yang lain, menerima masukan, dan menghasilkan keluaran dalam suatu proses transformasi yang teratur.

##### **2.1.2 Informasi**

Informasi (*Information*) adalah data yang telah diproses atau diorganisasi ulang menjadi bentuk yang berarti, yang dibentuk dari kombinasi data yang diharapkan memiliki arti bagi penerima [1].

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerima dan bermanfaat dalam pengambilan keputusan saat ini dan masa mendatang [2]. Dari definisi diatas, dapat diartikan bahwa informasi adalah hasil pemrosesan data kebentuk yang penting bagi penerima dan dapat digunakan sebagai dasar pengambilam keputusan.

Untuk memperoleh informasi, diperlukan adanya data yang akan diolah dan unit pengolah.

### 2.1.3 Sistem Informasi

Sistem informasi merupakan suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan [3]. Sistem informasi merupakan pengaturan orang, data, proses dan informasi teknologi yang berinteraksi untuk mengumpulkan data dan menyediakan *output* informasi yang diperlukan untuk mendukung sebuah organisasi [1].

Dari definisi diatas, dapat diartikan bahwa sistem informasi adalah sistem yang mentransformasikan data menjadi informasi untuk mendukung sebuah organisasi mencapai sasaran.

Komponen sistem informasi merupakan semua sistem informasi yang menggunakan sumber daya manusia, perangkat keras, perangkat lunak, data dan jaringan untuk melakukan aktifitas masukan, pemrosesan, keluaran, penyimpanan dan pengendalian yang mengubah sumber daya data menjadi produk informasi.

Aktifitas sistem informasi meliputi :

1. Masukan sumber daya data.

Data mengenai transaksi bisnis dan kegiatan lainnya harus ditangkap dan disiapkan untuk pemrosesan melalui aktifitas masukan. Masukan biasanya berbentuk aktifitas *entry data* seperti pencatatan dan pengeditan.

2. Pemrosesan data menjadi informasi.

Data biasanya tergantung pada aktifitas pemrosesan seperti penghitungan, perbandingan, pemindahan, pengklasifikasian dan pengikhtisaran. Aktifitas ini mengatur, menganalisis, dan memanipulasi data hingga mengubahnya kedalam informasi bagi para pemakai akhir.

3. Keluaran produk informasi.

Informasi dalam berbagai bentuk dikirim ke pemakai akhir dan disediakan untuk mereka dalam aktifitas keluaran. Tujuan dari sistem informasi adalah untuk menghasilkan produk informasi yang tepat bagi para pemakai akhir. Produk informasi umum meliputi pesan, laporan, formulir dan lainnya. [5].

Dengan demikian, aktifitas sistem informasi meliputi masukan sumber daya data, pemrosesan data menjadi bentuk informasi, dan *output*/keluaran dari informasi itu sendiri.

## 2.2 Analisis dan Perancangan Sistem

Analisis sistem merupakan pembelajaran sebuah sistem dan komponen-komponennya sebagai prasyarat desain sistem, spesifikasi sebuah sistem yang baru dan diperbaiki. Analisis sistem ditujukan untuk menyediakan tim proyek dengan pemahaman yang lebih menyeluruh terhadap masalah-masalah dan kebutuhan-kebutuhan yang memicu proyek. [1].

Dengan demikian, analisis sistem merupakan suatu pemecahan dari suatu sistem informasi untuk dievaluasi permasalahan atau kekurangannya sesuai kebutuhan/tujuan yang diharapkan.

Perancangan sistem merupakan spesifikasi atau konstruksi solusi yang teknis dan berbasis komputer untuk persyaratan bisnis yang diidentifikasi dalam analisis sistem [1]. Dengan demikian, desain sistem sebagai perancangan kembali sistem informasi yang diuraikan/dipecahkan menjadi suatu kesatuan yang utuh dan berfungsi.

Terdapat beberapa strategi atau teknik untuk melakukan desain sistem, yaitu:

1. Desain struktur modern (*modern structured design*).

Yaitu sebuah teknik desain sistem yang menguraikan proses-proses sistem menjadi komponen-komponen yang dapat dikelola.

2. Teknik informasi (*information engineering*).

Yaitu teknik model-driven dan pusat data, tetapi sensitif terhadap proses yang dapat digunakan untuk merencanakan menganalisis dan mendesain sistem informasi.

3. Prototyping.

Yaitu teknik yang menggunakan kertas dan pensil dalam menggambar *layout* atau struktur *output*, *input* dan *database* serta aliran prosedur.

4. Desain berorientasi objek (*object oriented design*).

Yaitu teknik yang menghilangkan pemisahan yang menyangkut data dan proses. Teknik ini dapat memperbaiki karakteristik data atau proses untuk sebuah objek yang telah diidentifikasi selama analisis sistem.

#### 5. *Rapid Application Development (RAD)*.

Yaitu sebuah pendekatan desain sistem yang menggunakan teknik terstruktur, dan *prototyping* untuk mengembangkan sistem secara cepat [1].

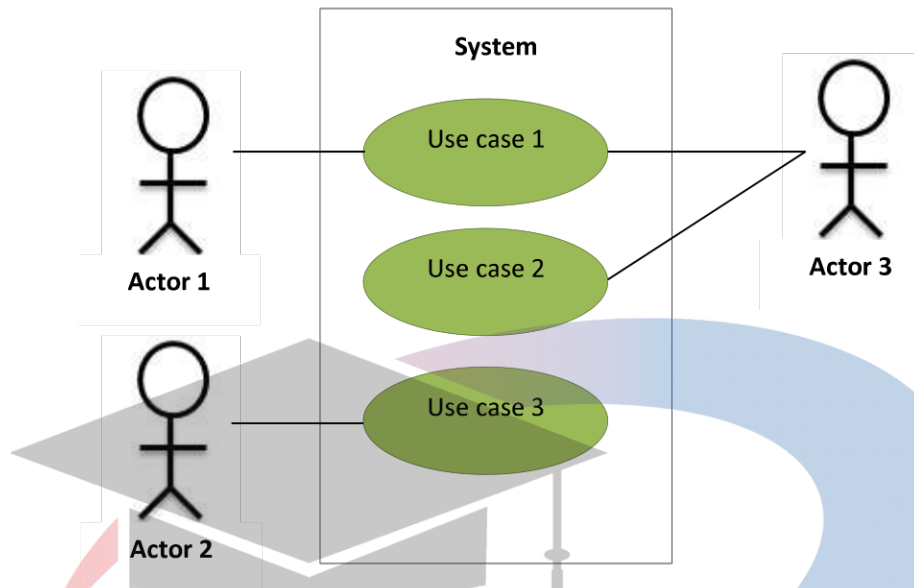
Dengan demikian, terdapat beberapa cara untuk melakukan perancangan suatu sistem informasi.

### 2.3 **Pemodelan *Use-Case Diagram***

*Use-case modelling*/pemodelan *use-case* adalah sebuah pendekatan yang memfasilitasi pengembangan berpustakaan 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. Penggunaan pemodelan *use-case* memfasilitasi dan mendorong keterlibatan pengguna, yang merupakan faktor sukses kritis yang memastikan sukses proyek.

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 terminologi 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. *Use-case* merupakan hasil penyusunan kembali lingkup fungsionalitas sistem menjadi banyak pernyataan fungsionalitas sistem yang lebih kecil.

*Use-case* disajikan secara gratis dengan elips horizontal dengan nama *use-case* muncul diatas, dibawah, atau didalam elips tersebut. Sebuah *use-case* merepresentasikan satu tujuan tunggal dari sistem dan menggambarkan satu rangkaian kegiatan dan interaksi pengguna untuk mencapai tujuan. [1].



Gambar 2.1 Pemodelan *Use-Case*

*Use-case* diawali atau dipicu oleh pengguna eksternal yang dinamakan *actor*/pelaku. Pelaku menginisiasi kegiatan sistem, yakni sebuah *use-case*, dengan maksud melengkapi beberapa tugas bisnis yang menghasilkan sesuatu yang dapat diukur. Dalam kenyataannya, seorang pelaku tidak harus manusia, dapat saja berupa perusahaan, sistem informasi lain, alat eksternal seperti sensor panas, atau bahkan konsep waktu.

Pada diagram *use-case*, hubungan digambarkan sebagai sebuah garis antara dua simbol. Pemaknaan hubungan berbeda-beda tergantung bagaimana garis tersebut digambarkan dan tipe simbol apa yang digunakan untuk menghubungkan garis tersebut. Adapun hubungan-hubungan yang ada pada sebuah diagram *use-case* :

- a. *Association* (Gabungan). Hubungan antara seorang pelaku dan satu *use-case* terbentuk kapanpun *use-case* menggambarkan interaksi antar keduanya.
- b. *Extend*. Untuk membuat *use-case* menjadi sederhana dan mudah dipahami, maka dapat mengekstrak langkah-langkah yang lebih kompleks kedalam *use-case* mereka sendiri dengan tujuan untuk memperluas fungsionalitas *use-case* sebelumnya.
- c. *Uses (Include)*. Merepresentasikan satu bentuk “*reuse*” merupakan alat yang sangat baik untuk mengurangi redundansi diantara *use-case*. [10].



## 2.4 Konsep Database

Basis data merupakan kumpulan data yang saling berhubungan, yang disimpan secara bersama sedemikian rupa dan tanpa pengulangan (redundansi) yang tidak perlu, untuk memenuhi berbagai kebutuhan [11].

Tujuan basis data yang efektif, yaitu :

1. Memastikan bahwa data dapat dipakai diantara pemakai untuk berbagai aplikasi.
2. Memelihara data baik keakuratan maupun kekonsistennannya.
3. Memastikan bahwa semua data yang diperlukan untuk aplikasi sekarang dan yang akan datang akan disediakan dengan cepat.
4. Membolehkan pemakai untuk membangun pandangan personalnya tentang data tanpa memperhatikan cara data disimpan secara fisik [12].

Adapun konsep *database* untuk analisis sistem yaitu :

### 1. *Field*

Merupakan implementasi fisik pada sebuah atribut data. *Field* adalah unit terkecil dari data *meaningful* yang telah disimpan pada sebuah *file* dan *database*. *Field* mempunyai empat tipe yaitu:

- a. *Primary key*, sebuah *field* yang nilainya mengidentifikasi satu dan hanya satu *record* pada sebuah *file*.
- b. *Secondary key*, sebuah pengidentifikasi alternatif pada sebuah *database*. Nilai *secondary key*, mungkin mengidentifikasi sebuah *record* tunggal atau sebuah subset dari semua *record*.
- c. *Foreign key*, yaitu *pointer* ke *record-record* dari sebuah *file* lain pada sebuah *database*.
- d. *Descriptive key*, yaitu semua *field* lainnya (*nonkey*) yang menyimpan data bisnis.

### 2. *Record*

Merupakan sebuah kumpulan *field* yang disusun pada format yang telah ditentukan.

### 3. *File* dan tabel

*File* merupakan kumpulan dari semua kejadian dari sebuah struktur *record* yang ditentukan. Tabel merupakan ekuivalen *database* relasional dari sebuah *file* [1].

## 2.5 Persyaratan Sistem

Selama menentukan persyaratan sistem, para analis mengumpulkan informasi tentang sistem apa yang harus dilakukan dari berbagai sumber sebanyak mungkin. Sumber tersebut meliputi pengguna dari sistem saat ini, laporan, formulir dan prosedur. Semua persyaratan sistem secara hati-hati didokumentasikan dan siap untuk distruktur. Menstruktur berarti mengambik persyaratan sistem yang ditemukan selama penentuan persyaratan dan memerintahkan mereka kedalam tabel, diagram dan format lain yang membuat mereka lebih mudah untuk menerjemahkan kedalam spesifikasi sistem teknis.

Sistem analis yang baik memerlukan karakteristik sebagai berikut :

1. Rasa ingin tahu. Sistem analis harus mempertanyakan segala sesuatu.
2. Ketidakberpihakan. Peran sistem analis adalah untuk menemukan solusi terbaik untuk setiap masalah maupun peluang. Sistem analis harus mempertimbangkan isu yang diangkat oleh semua orang dan memikirkan cara terbaik untuk mengatasinya.
3. Perhatikan setiap detail. Setiap fakta harus sesuai dengan fakta lainnya jika ketidakcocokan pada satu elemen maka sistem tersebut akan gagal kedepannya.
4. Kreatif. Analisis adalah melihat sebagian maupun keseluruhan sistem secara kreatif jadi sistem analis harus memiliki pemikiran yang dinamis tentang sebuah sistem.

Hasil dari penentuan persyaratan adalah berbagai jenis informasi yang dikumpulkan selama proses penentuan. Informasi dapat mengambil banyak bentuk : transkrip wawancara, catatan observasi dan analisis dokumen, formulir, laporan, deskripsi pekerjaan dan dokumen lainnya, dan *output* yang dihasilkan komputer seperti *prototype* sistem. Singkatnya, apapun yang analis kumpulkan, merupakan bagian dari penentuan persyaratan sistem termasuk dalam penyampaian.

### 2.5.1 Prototyping

*Prototyping* adalah sebuah proses berulang dimana analis dan pengguna membangun sebuah versi yang belum sempurna dari sebuah sistem informasi berdasarkan *feedback* dari pengguna. *Prototyping* bisa menggantikan siklus hidup pengembangan sistem atau menambah ketentuan persyaratan proses.

Mewawancarai pengguna dan mengumpulkan dokumentasi dilakukan untuk menetapkan persyaratan *prototyping*. *Prototyping* memungkinkan analisis mengkonversi persyaratan dasar menjadi versi sistem informasi yang diinginkan dengan cepat meski terbatas.

Pengguna melihat dan mencoba *prototype* tersebut, kemudian analisis mendesain kembali *prototype* dengan menggabungkan perubahan yang diusulkan. Analisis akan kembali menggabungkan perubahan yang diusulkan pengguna. Melalui proses yang berulang, analisis berkesempatan untuk dapat menangkap persyaratan sistem lebih baik. *Prototyping* bertujuan untuk mendukung penentuan persyaratan dalam mengembangkan spesifikasi yang nyata untuk sistem utama, bukan untuk membangun sistem utama.

*Prototyping* berguna untuk menentukan persyaratan ketika :

1. Persyaratan pengguna tidak jelas atau kurang dipahami, dimana dalam banyak kasus untuk sistem yang benar-benar baru atau sistem yang mendukung pembuatan keputusan.
2. Satu atau beberapa pengguna dan *stakeholder* lainnya ikut terlibat dengan sistem.
3. Kemungkinan desain yang kompleks dan memerlukan bentuk nyata untuk sepenuhnya dievaluasi.
4. Masalah komunikasi antara pengguna dan analisis yang telah ada dimasa lalu, dan kedua belah pihak ingin memastikan bahwa persyaratan sistem spesifik mungkin.
5. Alat (seperti form dan laporan generator) dan data sudah tersedia untuk membangun sistem kerja dengan cepat.

*Prototyping* juga mempunyai beberapa kelemahan sebagai alat untuk menentukan persyaratan :

1. Sebuah kecenderungan untuk menghindari menciptakan dokumentasi formal dari persyaratan sistem, yang kemudian bisa membuat sistem lebih susah dikembangkan menjadi kerja sepenuhnya.
2. *Prototype* bisa menjadi istimewa bagi pengguna awal dan sulit untuk berbaaur atau beradaptasi terhadap potensi pengguna lain.



3. *Prototype* sering dibangun sebagai sistem yang berdiri sendiri, sehingga mengabaikan isu-isu berbagai data dan interaksi dengan sistem lain yang sudah ada.
4. Melewati pemeriksaan SDLC, persyaratan sistem mungkin terlupakan.

