

## BAB II

### TINJAUAN PUSTAKA DAN LANDASAN TEORI

#### 2.1 Landasan Teori

##### 2.1.1 Sistem

Sistem mengandung arti kumpulan-kumpulan dari komponen-komponen yang dimiliki unsur keterkaitan antara satu dengan yang lainnya [7]. Sistem berkembang sesuai dengan konteks di mana pengertian itu digunakan[8]. Sistem memiliki beberapa karakteristik atau sifat yang terdiri dari komponen sistem, batasan sistem, lingkungan luar sistem, penghubung sistem, masukan sistem, keluaran sistem, pengolahan sistem dan sasaran sistem [9]. 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[10].

##### 2.1.2 Informasi

“Informasi adalah data yang telah diklasifikasikan, diolah atau diintegrasikan untuk digunakan dalam proses pengambilan keputusan” [11]. Informasi adalah data yang diolah menjadi lebih berguna dan berarti bagi penerimanya, serta untuk mengurangi ketidakpastian dalam proses pengambilan keputusan mengenai suatu keadaan [12]. Informasi dapat dikelola seperti halnya sumber daya lain, dan perhatian pada topik ini bersumber dari dua pengaruh[10]. Sumber dari informasi berupa data yang menggambarkan kejadian secara nyata yang telah terjadi pada saat tertentu[13].

##### 2.1.3 Sistem Informasi

Suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran yang tertentu [8]. Sistem Informasi merupakan suatu kombinasi teratur dari orang-orang, *hardware*, *software*, jaringan komunikasi dan sumber daya data yang mengumpulkan, mengubah, dan menyebarkan informasi dalam sebuah organisasi [12]. Sistem informasi dalam suatu organisasi dapat dikatakan sebagai suatu sistem yang menyediakan informasi bagi semua tingkatan dalam organisasi tersebut kapan saja diperlukan[14].

##### 2.1.4 Metode SDLC Waterfall

SDLC adalah sebuah metode yang digunakan untuk mengembangkan sebuah sistem. Sistem yang berjalan atau sedang digunakan oleh organisasi atau perusahaan akan terus dikembangkan untuk memperbaiki kekurangan-kekurangan pada sistem tersebut[6]. “Model *waterfall* adalah model SDLC (*Software Development Life Cycle*) yang paling sederhana. Model ini hanya cocok untuk pengembangan perangkat lunak yang spesifikasi yang tidak berubah[15].

##### 2.1.5 Metode Black Box

Pengujian ini menggunakan *Black Box Testing* (Pengujian Kotak Hitam). Metode *Black Box* adalah sebuah metode yang dipakai untuk menguji sebuah *software* tanpa harus memperhatikan detail *software*. Pengujian ini hanya memeriksa nilai keluaran berdasarkan nilai masukan masing-masing. Tidak ada upaya untuk mengetahui kode program apa yang *output* pakai[16].

### 2.1.6 Rekayasa Web

Rekayasa Web adalah suatu proses yang digunakan untuk menciptakan suatu sistem aplikasi berbasis web dengan menggunakan ilmu rekayasa, prinsip-prinsip manajemen dan pendekatan sistematis sehingga dapat diperoleh sistem dan aplikasi web dengan kualitas tinggi. Tujuannya untuk mengendalikan pengembangan, meminimalisasi resiko dan meningkatkan sistem berbasis web[17].

### 2.1.7 Pemrograman Berorientasi Objek

Pemrograman Berorientasi Objek (PBO) merupakan sebuah konsep pemrograman yang menggambarkan suatu proses penyelesaian masalah pada program dianalogikan sebagai objek yang saling berinteraksi satu sama lain. PBO secara harfiah sendiri sudah bermakna berfokus atau berorientasi objek, maka untuk memahaminya kita harus mendahului apa itu objek yang dimaksud[18].

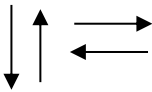
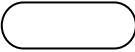

### 2.1.8 Basis Data



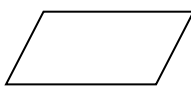

Basis data merupakan himpunan kelompok data yang saling terhubung dan diorganisasi sedemikian rupa supaya kelak dapat dimanfaatkan kembali secara cepat dan mudah. Kumpulan data dalam bentuk file, tabel, dan arsip yang saling berhubungan dan tersimpan dalam media penyimpanan elektronis, untuk kemudahan dalam pengaturan, pemilah, pengelompokan dan pengorganisasian data sesuai tujuan[19].

### 2.1.9 Flowchart

Bagan alir merupakan bagan-bagan (*chart*) yang menunjukkan alir (*flow*) di dalam program atau prosedur sistem secara logika[6]. Tabel 2. 1 adalah tabel flowchart.

**Tabel 2. 1** Tabel *Flowchart*

No	Nama	Simbol	Keterangan
1.	Flow Direction Symbol		Yaitu simbol yang digunakan untuk menghubungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga <i>connecting line</i> .
2.	Terminator Symbol		Yaitu simbol untuk permulaan ( <i>start</i> ) atau akhir ( <i>end</i> ) dari suatu kegiatan.
3.	Processing Symbol		Yaitu simbol yang menunjukkan pengolahan yang

			dilakukan oleh komputer.
4.	Manual Operation Symbol		Yaitu simbol yang menunjukkan pengolahan data yang tidak dilakukan oleh komputer.
5.	Decision Symbol		Yaitu simbol untuk pemilihan proses berdasarkan kondisi yang ada.
6.	Input-Output Symbol		Yaitu simbol yang menyatakan proses input dan output tanpa tergantung dari jenis peralatannya.
7.	Document Symbol		Yaitu simbol yang menyatakan input berasal dari dokumen dalam bentuk kertas atau <i>output</i> dicetak ke kertas.




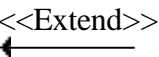

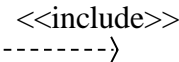
### 2.1.10 Unified Modelling Language (UML)

Basis data merupakan himpunan kelompok data yang saling terhubung dan diorganisasi sedemikian rupa supaya kelak dapat dimanfaatkan kembali secara cepat dan mudah. Kumpulan data dalam bentuk file, tabel, dan arsip yang saling berhubungan dan tersimpan dalam media penyimpanan elektronik, untuk kemudahan dalam pengaturan, pemilah, pengelompokan dan pengorganisasian data sesuai tujuan[6].

#### 1. Use Case Diagram

*Use case diagram* merupakan diagram yang digunakan untuk menggambarkan penggunaan dari sistem atau produk. Diagram ini bisa dikatakan sebagai diagram yang mudah karena pembuat umumnya mengetahui kegunaan dari produknya. Kegunaan dari produk akan terlihat secara keseluruhan dan memberikan gambaran sederhana dari bisnis[20]. Tabel 2. 2 adalah tabel usecase diagram.


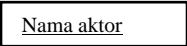

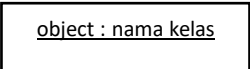

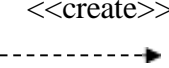
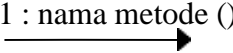
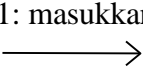
**Tabel 2. 2** Tabel *Use case Diagram*

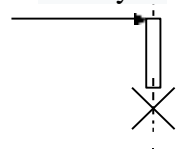
No	Nama	Simbol	Fungsi
1.	<i>Use Case</i>		Digunakan untuk saling bertukar pesan antar unit atau <i>actor</i> , biasanya dinyatakan dengan menggunakan kata kerja di awal frase nama <i>use case</i> .
2.	<i>Actor</i>		Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan <i>use case</i> .
3.	<i>Asosiasi/ association</i>		Komunikasi antara <i>actor</i> dan <i>use case</i> yang saling berhubungan atau <i>use case</i> yang memiliki interaksi dengan aktor.
4.	<i>Ekstensi/ extend</i>		Menunjukkan bahwa suatu bagian dari elemen dapat disisipkan ke dalam elemen yang lainnya
5.	<i>Generalisasi/ generalization</i>		Hubungan dimana objek anak ( <i>descendent</i> ) berbagi perilaku dan struktur data dari objek yang ada di atasnya objek induk ( <i>ancestor</i> ).
6.	<i>Include</i>		Menunjukkan bahwa suatu bagian dari elemen dapat memicu eksekusi bagian dari elemen lain (yang ada pada tanda panah).

## 2. Sequence Diagram

*Sequence diagram* adalah diagram yang menggambarkan interaksi antar objek. *Sequence diagram* secara khusus menjabarkan behavior sebuah skenario tunggal. Diagram tersebut menunjukkan sejumlah objek contoh dan pesan-pesan yang melewati objek ini dalam sebuah *use case*[20]. Tabel 2. 3 adalah tabel *sequence diagram*.

Tabel 2. 3 Tabel *Sequence Diagram*



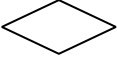

No	Nama	Simbol	Fungsi
1.	<i>Actor</i>	 atau  Tanpa waktu aktif	Aktor menyatakan orang, proses, atau sistem lain yang berkaitan dengan sistem informasi yang akan dibuat di luar sistem informasi tersebut.
2.	Garis hidup / <i>lifeline</i>		<i>Lifeline</i> menyatakan keberadaan dari suatu objek dalam basis waktu. Notasi yang digunakan untuk <i>lifeline</i> adalah garis lurus vertikal putus-putus yang ditarik dari sebuah objek.
3.	Objek		Menyatakan berinteraksi.
4.	Waktu aktif		Menyatakan objek dalam keadaan aktif dan saling terhubung yang merupakan sebuah tahapan yang dilakukan di dalamnya.
5.	Pesan tipe <i>create</i>		Pesan tipe <i>create</i> , menyatakan suatu objek membuat objek yang lain dengan arah panah yang mengarah pada objek yang dibuat.
6.	Pesan tipe <i>call</i>		Pesan tipe <i>call</i> , menyatakan suatu objek yang memanggil operasi/metode yang ada pada objek lain atau dirinya sendiri.
7.	Pesan tipe <i>send</i>		Pesan tipe <i>send</i> , menyatakan suatu objek mengirimkan data/masukan/informasi ke objek lainnya dengan arah panah yang mengarah pada objek yang dikirim..
8.	Pesan tipe <i>return</i>		Pesan tipe <i>return</i> , menyatakan suatu objek yang telah menjalankan

		1 : keluaran ----->	suatu operasi atau metode akan menghasilkan suatu kembalian ke objek tertentu dengan arah panah yang mengarah pada objek yang menerima kembalian.
9.	10. Pesan tipe <i>destroy</i>	<<destroy>> 	Pesan tipe <i>destroy</i> , menyatakan suatu objek yang mengakhiri hidup objek lain dengan arah panah yang mengarah pada objek yang diakhiri, <i>destroy</i> sebaiknya digunakan jika ada <i>create</i>

### 2.1.11 Entity Relationship Diagram (ERD)

Diagram hubungan entitas merupakan suatu model yang menggambarkan suatu 25 penyimpanan basis data yang menggunakan susunan data dalam sistem secara *abstrak*[6]. Tabel 2. 4 adalah tabel ERD.

Tabel 2. 4 Tabel ERD

No.	Nama	Simbol	Keterangan Fungsi
1.	Entitas		Persegi panjang menyatakan himpunan entitas adalah orang, kejadian, atau berada.
2.	Atribut		<i>Atribut</i> merupakan informasi yang diambil tentang sebuah entitas.
3.	Relasi		Belah ketupat menyatakan himpunan relasi merupakan hubungan antar entitas.
4.	Link		Garis sebagai penghubung antara himpunan, relasi, dan himpunan entitas dengan atributnya