

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Defenisi Sistem**

Sistem adalah suatu himpunan suatu benda nyata atau abstrak (a set of thing) yang terdiri dari bagian-bagian atau komponen-komponen yang saling berkaitan, berhubungan, berketergantungan, saling mendukung yang secara keseluruhan bersatu dalam satu kesatuan (Unity) untuk mencapai tujuan tertentu secara efisien dan efektif (Fat, 2014).

Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi (Jogianto, 2005).

Sistem merupakan kumpulan dari sub-sub sistem baik sistem abstrak maupun fisik yang saling terintegrasi dan berkolaborasi untuk mencapai tujuan tertentu. Sistem adalah setiap sesuatu yang terdiri dari objek-objek, atau unsur-unsur, atau komponen - komponen yang bertata kaitan dan bertata hubungan satu sama lain, sedemikian rupa sehingga unsur-unsur tersebut merupakan satu kesatuan pemrosesan atau pengolahan yang tertentu. Kumpulan elemen-elemen yang saling terkait dan bekerja sama untuk memproses masukan input yang ditujukan kepada sistem tersebut dan mengolah masukan sampai menghasilkan keluaran (output) yang diinginkan. Dengan adanya suatu sistem maka perlu diketahui juga batasan – batasan dengan terbentuknya sistem tersebut. Batasan Sistem yaitu suatu batasan / kondisi yang memisahkan antara sistem dengan sekitarnya. Sehingga terbentuk suatu wilayah yang berada di sekitar sitem itu sendiri yaitu yang dinamakan Sub-sistem dan Supra sistem.

1. Subsistem merupakan komponen atau bagian dari suatu system, subsistem ini bisa phisik ataupun abstrak. Subsistem sebenarnya hanyalah sistem di dalam suatu sistem, ini berarti bahwa sistem berada pada lebih dari satu tingkat. Pemisalan lainnya, mobil adalah suatu

system yang terdiri dari system-sistem bawahan seperti system mesin, system badan mobil dan system rangka. Masing-masing system ini terdiri dari system tingkat yang lebih rendah lagi.

2. Walaupun istilah supersistem jarang digunakan, system seperti ini ada. Jika suatu system adalah bagian dari system yang lebih besar, system yang lebih besar itu adalah supersistem.

Dari definisi dan penjelasan diatas dapat disimpulkan suatu system terdiri dari elemen yang bisa berbentuk individu atau bagian-bagian yang terpisah, kemudian berinteraksi satu sama lain untuk mencapai tujuan.

### **2.1.1 Karakteristik Sistem**

Supaya sistem dapat dikatakan sistem yang baik memiliki karakteristik yaitu:

1. Memiliki komponen

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, bekerja sama membentuk satu kesatuan. Komponen-komponen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap sistem tidak peduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses sistem secara keseluruhan. Suatu sistem dapat mempunyai suatu sistem yang lebih besar yang disebut supra sistem, misalnya suatu perusahaan dapat disebut dengan suatu sistem dan industri yang merupakan sistem yang lebih besar dapat disebut dengan supra sistem. Kalau dipandang industri sebagai suatu sistem, maka perusahaan dapat disebut sebagai subsistem. Demikian juga bila perusahaan dipandang sebagai suatu sistem, maka sistem akuntansi adalah subsistemnya.

2. Batas sistem (boundary)

Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu

kesatuan. Batas suatu sistem menunjukkan ruang lingkup (scope) dari sistem tersebut.

3. Lingkungan luar sistem (environment)  
Adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem.
4. Penghubung sistem (interface)  
Merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya.
5. Masukan sistem (input)  
Merupakan energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (maintenance input) dan masukan sinyal (signal input). Maintenance input adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Signal input adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh didalam sistem komputer, program adalah maintenance input yang digunakan untuk mengoperasikan komputernya dan data adalah signal input untuk diolah menjadi informasi.
6. Keluaran sistem (Output)  
Merupakan hasil dari energi yang diolah oleh sistem.
7. Pengolah sistem (Process)  
Merupakan bagian yang memproses masukan untuk menjadi keluaran yang diinginkan.
8. Sasaran sistem  
Kalau sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya.

### **2.1.2 Pengertian Informasi**

Informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi si penerima dan mempunyai nilai nyata atau yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau keputusan-keputusan yang akan datang.

Fungsi utama informasi yaitu menambah pengetahuan atau mengurangi ketidak pastian pemakai informasi, karena informasi berguna memberikan gambaran tentang suatu permasalahan sehingga pengambil keputusan dapat menentukan keputusan lebih cepat, informasi juga memberikan standard, aturan maupun indikator bagi pengambil keputusan.

### **2.1.3 Konsep Dasar Sistem Informasi**

Sistem informasi adalah sebuah sistem yang terdiri dari komponen-komponen tertentu sehingga tercapai sebuah informasi yang mendukung pengambilan keputusan didalam suatu organisasi untuk dapat mencapai sasaran dan tujuannya. Sistem informasi didalam suatu organisasi atau perusahaan mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan yang dibutuhkan.

### **2.1.4 Konsep Dasar Informasi**

Konsep dasar informasi tidak lepas dari pengertian informasi, yaitu hasil pengolahan data menjadi bentuk yang berguna bagi penerimanya. Jadi, sumber informasi adalah data. Data adalah kenyataan yang menggambarkan peristiwa yang terjadi pada saat tertentu. Informasi juga berupa data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya.

Konsep dasar informasi menurut Gordon B. Darwis (1985) yaitu bahwa informasi sebagai sebuah data yang telah dilakukan pengolahan menjadi suatu bentuk yang lebih berarti serta berguna bagi penggunaannya dalam pengambilan keputusan baik untuk masa kini atau yang akan datang.

## **2.2 Website**

Website merupakan kumpulan halaman digital yang berisi informasi berupa teks, animasi, gambar, suara dan video atau gabungan dari semuanya yang terkoneksi oleh internet, sehingga dapat dilihat oleh seluruh siapa pun yang terkoneksi jaringan internet (Sunarti et al., n.d. 2019). Website adalah keseluruhan

halaman-halaman web yang terdapat dari sebuah domain yang mengandung informasi (Yuhefizar, 2013). Website merupakan halaman atau kumpulan halaman pada sebuah domain di internet yang dibuat dengan tujuan tertentu. Website berisi tampilan berupa teks, gambar, animasi, audio, video atau gabungan satu dengan lainnya.

### **2.3 Alat Bantu Analisa Perancangan**

Unified Modelling Language merupakan alat perancangan sistem yang berorientasi pada objek. Secara filosofi kemunculan UML diilhami oleh konsep yang telah ada yaitu konsep permodelan Object Oriented (OO), karena konsep ini menganalogikan sistem seperti kehidupan nyata yang didominasi oleh obyek dan digambarkan atau dinotasikan dalam simbol-simbol yang cukup spesifik maka OO memiliki proses standard dan bersifat independen. UML diagram memiliki tujuan utama untuk membantu tim pengembangan proyek berkomunikasi, mengeksplorasi potensi desain, dan memvalidasi desain arsitektur perangkat lunak atau pembuat program. Komponen atau notasi UML diturunkan dari 3 (tiga) notasi yang telah ada sebelumnya yaitu Grady Booch, OOD (Object-Oriented Design), Jim Rumbaugh, OMT (Object Modelling Technique), dan Ivar Jacobson OOSE (Object-Oriented Software Engineering) (Haviluddin, 2011).

Unified Modelling Language (UML) merupakan bahasa pemodelan perangkat lunak atau sistem dengan konsep pemrograman berorientasi objek yang dapat analisa dan menjabarkan secara rinci apa yang diperlukan oleh sistem. Sedangkan database Oracle merupakan sebuah produk database yang ciptakan oleh Oracle Corporation yang mempunyai kemampuan proses dan performa yang tinggi (Suendri, 2019).

#### **2.3.1 Use Case Diagram**




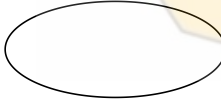

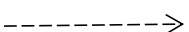
Hendini (2016) menyatakan Use case diagram merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case digunakan untuk mengetahui fungsi apa saja yang ada di dalam sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi tersebut.



Diagram yang menggambarkan actor, use case dan relasinya sebagai suatu urutan tindakan yang memberikan nilai terukur untuk aktor. Sebuah use case digambarkan sebagai elips horizontal dalam suatu diagram UML use case. Use Case memiliki dua istilah :

1. System use case (Interaksi dengan sistem).
2. Business use case (Interaksi bisnis dengan konsumen atau kejadian nyata).

**Tabel 2.3 Simbol Use Case Diagram**

Symbol	Penjelasan
	<i>Actor</i> Mendefinisikan seperangkat peranan yang user sistem dapat diperankan ketika berinteraksi dengan use case.
	<i>Association</i> Menggambarkan interaksi antara actor dan use case.
	<i>Generalization</i> Relasi antar use case, dimana salah satunya dalam bentuk yang lebih umum dari yang lain.
	<i>Use Case</i> Sebuah deskripsi dari seperangkat aksi aksi berurutan yang ditampilkan pada sebuah sistem.
	<i>System</i> Tempat seluruh aktivitas-aktivitas sistem yang sedang berjalan.
	<i>Dependency</i> Untuk menggambarkan ketergantungan sebuah use case dengan use case lainnya.
<code>&lt;&lt;Include&gt;&gt;</code>	<i>Include</i> Menggambarkan bahwa keseluruhan dari sebuah

----->	use case merupakan fungsionalitas use case lainnya.
<<Extend>> ----->	<i>Extend</i> Menggambarkan hubungan antara use case dimana sebuah use case merupakan fungsionalitas use case lainnya apabila kondisi tertentu dipenuhi.

### 2.3.2 Flowchart

Flowchart adalah bagan alir yang menunjukkan suatu program atau prosedur sistem secara logika. Flowchart digunakan untuk alat bantu komunikasi dan dokumentasi. Flowchart juga mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah dan salah satu cara penyajian algoritma.



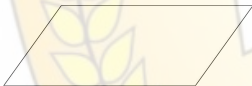

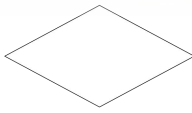

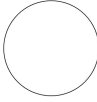
Sulindawati & Fathoni (2010) menyatakan Flowchart adalah penggambaran secara grafik dari langkah-langkah dan urutan prosedur dari suatu program. Flowchart menolong analis dan programmer untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoperasian. Flowchart biasanya mempermudah penyelesaian suatu masalah khususnya masalah yang perlu dipelajari dan dievaluasi lebih lanjut. Bila seorang analis dan programmer akan membuat flowchart, ada beberapa petunjuk yang harus diperhatikan, seperti:

1. Flowchart digambarkan dari halaman atas ke bawah dan dari kiri ke kanan.
2. Aktifitas yang digambarkan harus didefinisikan secara hati-hati dan definisi ini harus dapat dimengerti oleh pembacanya.
3. Kapan aktifitas dimulai dan berakhir harus ditentukan secara jelas.
4. Setiap langkah dari aktifitas yang harus diuraikan dengan menggunakan deskripsi kata kerja.
5. Setiap langkah dari aktifitas harus berada pada urutan yang benar.

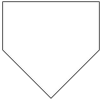



6. Lingkup dan range dari aktifitas yang sedang digambarkan harus ditelusuri dengan hati-hati. Percabangan-percabangan yang memotong aktifitas yang sedang digambarkan tidak perlu digambarkan pada flowchart yang sama. Simbol konektor harus digunakan dan percabangan diletakkan pada halaman yang terpisah atau hilangkan seluruhnya bila percabangannya tidak berkaitan dengan sistem.
7. Gunakan simbol-simbol flowchart yang standar.

Berikut simbol-simbol dari flowchart dapat dilihat di Tabel 2.4

**Tabel 2.4 Simbol Flowchart**

No	Simbol	Fungsi
1		<i>Terminal</i> yaitu awal atau akhir sebuah program
2		<i>Preparation</i> atau persiapan yaitu pemberian harga awal dan deklarasi variabel dan lain-lain
3		<i>Input / output</i> yaitu proses pemasukan atau pengeluaran data
4		<i>Process</i> yaitu proses pengolahan data yang dilakukan oleh computer
5		<i>Decision</i> yaitu proses pengambilan keputusan untuk memilih satu diantara dua alternative
6		<i>Subrutine</i> yaitu kumpulan langkah-langkah
7		<i>On Page Connector</i> yaitu tanda gabung dalam halaman yang sama



8		<i>Off Page Connector</i> yaitu tanda gabung lain halaman
9		<i>Arrow</i> yaitu penunjuk arah aliran program
10		<i>Document</i> , menyatakan simbol untuk data yang berbentuk kertas maupun untuk informasi
11		<i>Simbol</i> untuk output, yang ditunjukkan ke suatu device, seperti printer

## 2.4 Data Base

### 2.4.1 Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) adalah yang digunakan untuk menggambarkan model Entity Relationship yang berisi komponen-komponen. Himpunan Entitas dan Himpunan Relasi yang masing-masing dilengkapi dengan atribut-atribut yang mempersentasikan seluruh fakta dari dunia nyata yang kita tinjau.

Entity Relationship diagram (ERD) merupakan teknik yang digunakan untuk memodelkan kebutuhan data dari suatu organisasi, biasanya oleh System Analys dalam tahap analisis persyaratan proyek pengembangan system. Sementara seolah-olah teknik diagram atau alat peraga memberikan dasar untuk desain database relasional yang mendasari sistem informasi yang dikembangkan. ERD bersama-sama dengan detail pendukung merupakan model data yang pada gilirannya digunakan sebagai spesifikasi untuk database.

ERD terbagi atas 3 komponen, yaitu entitas (entity), atribut (attribute), dan relasi atau hubungan (relation). Secara garis besar, entitas merupakan dasar yang terlibat dalam sistem. Atribut berperan sebagai penjelas dari entitas, dan relasi atau hubungan menunjukkan hubungan yang terjadi antara dua entitas.

1. Entitas (entity) merupakan obyek yang mewakili sesuatu dalam dunia nyata, baik secara fisik maupun secara konsep

2. Relasi (relationship) akan menyatakan hubungan antar entitas terkait, termasuk terhadap entitas itu sendiri (rekursif).
3. Atribut (atributte) yaitu ciri dan karakteristik suatu tipe entitas. biasanya berupa kolom data dalam suatu tabel (field).

## **2.5 Software Pendukung**

### **2.5.1 PHP**

Erawan (2014) menyatakan PHP singkatan dari Hypertext Preprocessor. Php merupakan bahasa script yang dijalankan pada sisi server (SSS : Server Side Scripting). Database yang didukung PHP antara lain : MySQL, Informix, Oracle, Sybase, Solid, PostgreSQL, Generic ODBC. PHP adalah software Open Source, bebas untuk diunduh dan digunakan.

PHP (Hypertext Preprocessor) adalah bahasa script yang dapat ditanamkan atau disisipkan ke dalam HTML. PHP banyak dipakai untuk membuat program situs web dinamis” (Madcoms, 2016).

### **2.5.2 HTML**

HTML yang merupakan singkatan dari Hyper Text Markup Language adalah serangkaian kode program yang merupakan dasar dari representasi visual sebuah halaman Web. Didalamnya berisi kumpulan informasi yang disimpan dalam tag-tag tertentu, dimana tag-tag tersebut digunakan untuk melakukan format terhadap informasi yang dimaksud.

### **2.5.2 MySQL**

Menurut Madcoms MySQL adalah sistem manajemen Database SQL yang bersifat Open Source dan paling populer saat ini. Sistem Database MySQL mendukung beberapa fitur seperti multithreaded, multiuser dan SQL Database managemen system (DBMS)” (Madcoms, 2016).

### 2.5.3 Xampp

Menurut Madcoms (2016) Xampp adalah sebuah paket kumpulan software yang terdiri dari Apache, MySQL, PhpMyAdmin, PHP, Perl, Filezilla, dan lain.

Menurut Rima Safitri Xampp merupakan paket PHP berbasis *Open Source*. XAMPP mengkombinasi beberapa paket perangkat lunak berbeda ke dalam satu paket. Paket yang disediakan antara lain adalah Apache, MySQL, PHP, FilZilla FTP *Server*, phpMyAdmin, dan lain-lain (Safitri, 2018).

