

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Konsep Dasar Sistem**

Mempelajari suatu sistem akan lebih mengenal bila mengetahui terlebih dahulu mengenai teori konsep dasar suatu sistem itu.

##### **2.1.1 Model Pembelajaran Berbasis Web**

Pembelajaran berbasis web merupakan suatu kegiatan pembelajaran yang memanfaatkan media situs (*website*) yang bisa diakses melalui jaringan internet. (Rusman, 2013)

Secara sederhana dapat dikatakan bahwa semua pembelajaran yang memanfaatkan teknologi internet dan selama proses belajar dirasakan terjadi oleh yang mengikutinya maka kegiatan itu dapat disebut sebagai pembelajaran berbasis web.

##### **2.1.2 Definisi Sistem**

Sistem adalah sekelompok unsur yang erat hubungannya satu dengan lainnya, yang berfungsi bersama-sama untuk mencapai tujuan tertentu. (Suparlan, 2017)

Sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur atau variabel-variabel yang saling terorganisasi, saling berinteraksi, dan saling bergantung sama lain”(Suparlan, 2017)

##### **2.1.3 Sistem Berorientasi Obyek**

*Object oriented programming* atau pemrograman berorientasi objek (PBO) merupakan paradigma atau cara pandang pembuatan program dengan berorientasi atau

fokus pada objek. Semua data dan fungsi didalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek.(Amin, 2016)

Berbeda dengan pemograman terstruktur atau pemograman berorientasi prosedur atau *procedural oriented programming* (POP), dimana setiap objek menerima pesan atau data, memprosesnya dan mengirimnya ke objek lain. (Amin, 2016)

#### **2.1.4. Rancangan Website**

##### **1. Definisi Internet**

*Internet* merupakan kependekan dari *inteconnected networking* atau *international networking*, yaitu kumpulan yang sangat luas dari jaringan komputer besar dan kecil yang saling berhubungan dengan menggunakan jaringan komunikasi yang ada di seluruh dunia. (Suparlan, 2017)

##### **2. Definisi Website**

*Website* adalah keseluruhan halaman-halaman web yang terdapat dalam sebuah domain yang mengandung informasi. Sebuah *website* biasanya dibangun atas banyak halaman web yang saling berhubungan. Hubungan antara satu halaman web dengan halaman web yang lainnya disebut dengan hyperlink, sedangkan teks yang dijadikan media penghubung disebut hypertext. (Suparlan, 2017)

##### **3. Web Browser**

*Browser web* adalah *software* yang digunakan untuk menampilkan informasi dari server web. Software ini kini telah dikembangkan dengan menggunakan user interface grafis, sehingga pemakai dapat dengan melakukan point dan click untuk pindah antar dokumen.(Suparlan, 2017)

##### **4. Web Server**

*Web Server* adalah potongan perangkat lunak yang mendukung berbagai protocol Web, seperti HTTP, HTTPS, dan lain-lain untuk memproses permintaan client”. Atau pada pengertian lain dijelaskan bahwa WebServer adalah sebuah perangkat lunak server yang berfungsi menerima permintaan HTTP dan HTTPS dari klien yang dikenal dengan *web browser* yang umumnya berbentuk dokumen HTML.(Supriyanta, 2015)

#### **4. Dreamweaver**

*Dreamweaver* adalah sebuah HTML editor profesional untuk mendesain web secara visual dan mengelola situs atau halaman web.(Suparlan, 2017)

Pada *Dreamweaver CS6*, terdapat beberapa kemampuan bukan hanya sebagai *software* untuk desain *web* saja tetapi juga untuk menyunting kode serta pembuatan aplikasi *web*, antara lain: *HTML, ColdFusion, PHP, CSS, JavaScript\_ dan XML*.

#### **5. Basis Data**

Basis Data sendiri terdiri dari dua kata, yaitu Basis dan Data. Basis berarti markas atau gudang, tempat bersarang atau berkumpul. Sedangkan Data adalah representasi fakta dunia nyata yang mewakili suatu objek. (Supriyanta, 2015)

#### **6. XAMPP**

XAMPP merupakan program paket PHP dan MySQL berbasis opensource yang saat ini merupakan andalan para programmer PHP dalam melakukan programming dan melakukan testing hasil programnya. Sedangkan pengertian lainnya Xampp adalah program yang berisi paket Apache , MySQL dan phpMyAdmin. (Supriyanta, 2015)

##### **1. Apache**

Apache (Server HTTP Apache atau Server Web /WWW Apache ) adalah web server yang dapat dijalankan di banyak sistem operasi (Unix, BSD, Linux, Microsoft

Windows dan Novell Netware serta platform lainnya) yang berguna untuk melayani dan memfungsikan situs web. (Supriyanta, 2015)

## 2. MySQL

MySQL atau dibaca “My Sekuel” dengan suatu RDBMS (Relational Database Management System) merupakan aplikasi sistem yang menjalankan fungsi pengolahan data. MySQL sendiri pertama dikembangkan oleh MySQL AB yang kemudian diakuisisi oleh Sun Microsystems dan terakhir dikelola oleh Oracle Corporation. (Supriyanta, 2015)

## 3. PhpMyAdmin

PHP Myadmin adalah sebuah software berbasis pemrograman PHP yang dipergunakan sebagai administrator MySQL melalui browser (web) yang digunakan untuk management database. (Supriyanta, 2015)

### 2.1.5. Model Perkembangan Perangkat Lunak

Ada banyak model yang dikembangkan untuk membantu proses pengembangan perangkat lunak. Model-model ini pada umumnya mengacu pada model proses pengembangan sistem yang disebut *Sistem Life Cycle* (SDLC) yang terdiri dari identifikasi dan pemilihan proyek, inisiasi dan perancangan proyek, Analisis, Desain, Implementasi dan perawatan.

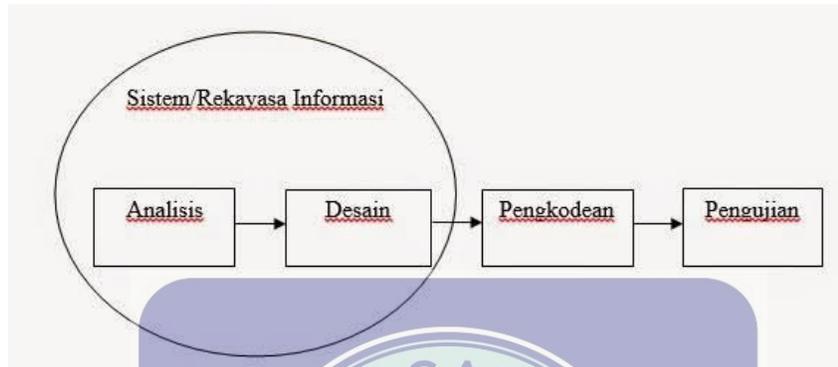
SDLC atau Software Development Life Cycle adalah proses mengembangkan atau mengubah Suatu sistem perangkat lunak dengan menggunakan model- model dan metodologi yang digunakan orang untuk mengembangkan system-sistem perangkat lunak sebelumnya”. (Larasati & Masripah, 2017)

#### 1. *Waterfall*

Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequencial linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun

menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

Tahapan-tahapan *waterfall* dapat dilihat pada gambar berikut:



Sumber : (Larasati & Masripah, 2017)

**Gambar II.1. Model SDLC Waterfall**

a. Kelebihan *waterfall*

- 1) Proses-prosesnya mudah dipahami
- 2) Mudah dalam pengolahan proyek
- 3) Dokumen dihasilkan setiap akhir *face*
- 4) Sebuah *face* dijalankan setelah *face* sebelumnya selesai
- 5) Struktur sistem jelas
- 6) Kondisi tepat SDLC
- 7) Kebutuhan *user* telah sangat dipahami
- 8) Kemungkinan terjadi perubahan kebutuhan *user* kecil

b. Kelemahan *waterfall*

- 1) Proyek dunia nyata jarang mengikuti alur proses.

Model SDLC air terjun (*waterfall*) sering juga disebut model sekuensial linier (*sequencial linear*) atau alur hidup klasik (*classic life cycle*). Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari analisis, desain, pengodean, pengujian, dan tahap pendukung (*support*).

## 2.2 Teori Pendukung

Peralatan pendukung yang digunakan penulis dalam menyusun tugas akhir adalah ERD dan UML.

### 2.2.1 Entity Relationship Diagram

Entity Relationship Diagram adalah alat pemodelan data utama dan akan membantu mengorganisasi data dalam suatu proyek ke dalam suatu proyek ke dalam entitas-entitas dan menentukan hubungan antarentitas".(Suparlan, 2017)

ERD terbagi atas tiga komponen, yaitu:

#### 1. Entitas (*Entity*)

Entity yaitu kumpulan objek-objek dengan sifat (property) yang sama, yang diidentifikasi oleh interprise mempunyai eksistensi yang independen" Entitas diberi nama dengan kata benda dan dapat dikelompokan sebagai berikut (Taufik, 2017):

- a) Nama Orang
- b) Nama benda
- c) Nama lokasi
- d) Nama Kejadian (terdapat unsur waktu didalamnya )

#### 2. Atribut (*Attribute*)

Field atau kolom data yang butuh disimpan dalam suatu entitas. Setiap entitas pasti memiliki atribut yang mendeskripsikan karakteristik (properti) dari entitas tersebut. Penempatan atribut bagi sebuah entitas umumnya memang didasarkan pada fakta yang ada, tetapi tidak selalu seperti itu. Simbol yang digunakan adalah bentuk lingkaran.

### 3. Relasi (*Relationship*)

*Relationship* adalah kumpulan keterhubungan yang mempunyai arti antara tipe entitas yang ada. Pada umumnya penghubung diberi nama dengan kata kerja dasar, sehingga memudahkan untuk melakukan pembacaan relasinya. (Taufik, 2017)

Simbol yang digunakan adalah belah ketupat. Relasi yang terjadi diantara dua himpunan entitas (misalnya A dan B) dalam satu basis data yaitu :

#### a) Satu ke Satu (*One to One*)

Yang berarti setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan B, dan begitu juga sebaliknya setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B.

#### b) Satu ke Banyak (*One to Many*)

Yang berarti setiap pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B

berhubungan dengan paling banyak dengan satu tiap entitas pada himpunan entitas A.

c) Banyak ke Satu (*Many to One*)

Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas A berhubungan dengan paling banyak satu entitas pada himpunan entitas B.

d) Banyak ke Banyak (*Many to Many*)

Yang berarti setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan demikian juga sebaliknya, dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A

### 1.2.1. LRS (*Logical Record Structure*)

*Logical Record Structured* (LRS) adalah sebuah model sistem yang digambarkan dengan dengan sebuah diagram-ER akan mengikuti pola atau aturan permodelan tertentu dalam kaitannya dengan konvensi ke LRS".(Larasati & Masripah, 2017)

### 1.2.2. *Unified Modeling Language*

UML (*Unified Modeling Language*) adalah ‘bahasa’ pemodelan untuk sistem atau perangkat lunak yang berparadigma ‘berorientasi objek”. Pemodelan (*modelling*) sesungguhnya digunakan untuk penyederhanaan permasalahan-permasalahan yang

kompleks sedemikian rupa sehingga lebih mudah dipahami dan dipelajari. (Larasati & Masripah, 2017)

Inilah beberapa tujuan atau fungsi dari penggunaan UML, yang diantaranya:

1. Dapat memberikan bahasa permodelan visual kepada pengguna dari berbagai macam pemrograman maupun proses rekayasa.
2. Dapat menyatukan praktek-praktek terbaik yang ada dalam permodelan.
3. Dapat memberikan model yang siap untuk digunakan, merupakan bahasa permodelan visual yang ekspresif untuk mengembangkan sistem dan untuk saling menukar model secara mudah.
4. Dapat berguna sebagai blue print, sebab sangat lengkap dan detail dalam perancangannya yang nantinya akan diketahui informasi yang detail mengenai koding suatu program.
5. Dapat memodelkan sistem yang berkonsep berorientasi objek, jadi tidak hanya digunakan untuk memodelkan perangkat lunak (*software*) saja.
6. Dapat menciptakan suatu bahasa permodelan yang nantinya dapat dipergunakan oleh manusia maupun oleh mesin.

#### **2.2.4. Use Case Diagram**

Diagram ini bersifat statis. Diagram ini memperlihatkan himpunan usecase dan aktor-aktor (suatu jenis khusus dari kelas). Diagram ini terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan serta diharapkan pengguna. (Susilowati, 2017)

#### **2.2.5. Activity Diagram**

Diagram ini bersifat dinamis Diagram ini adalah tipe khusus dari diagram state yang memperlihatkan aliran dari suatu aktifitas keaktifitas lainnya dari suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi–fungsi dalam suatu sistem dan memberi tekanan pada aliran kendali antar objek .(Susilowati, 2017)

#### 2.2.6. *Sequence Diagram*

Diagram ini bersifat dinamis. Diagram sequence merupakan diagram interaksi yang menekankan pada pengiriman pesan (message) dalam suatu waktu tertentu (Susilowati, 2017).

#### 2.2.7. *Class Diagram*

Tujuan utama dari *Class Diagram* adalah untuk menciptakan sebuah kosa kata yang digunakan oleh analis dan pengguna. *Class diagram* biasanya merupakan hal-hal, ide-ide atau konsep yang terkandung dalam aplikasi. Misalnya, jika sedang membangun sebuah aplikasi penggajian, diagram kelas mungkin akan berisi kelas yang mewakili hal-hal seperti karyawan, cek, dan pendaftaran gaji. *Class diagram* juga akan menggambarkan hubungan antara kelas.

*Class Diagram* bersifat statis. Diagram ini memperlihatkan himpunan kelas-kelas, antarmuka-antarmuka, kolaborasi-kolaborasi serta relasi (Susilowati, 2017).

*Class diagram* memiliki 3 area pokok:

1. Name
2. Attribute
3. Method