

## **BAB II**

### **LANDASAN TEORI**

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

Definisi sistem menurut (Muhammad Muslihudin & Oktafianto, 2016) mengemukakan bahwa “Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berkaitan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu”. Secara sederhana suatu sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur komponen atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain.

##### **A. Model Pembelajaran Berbasis Web**

Menurut (Subiantoro, 2018) dalam jurnalnya Sistem Informasi absensi merupakan faktor penting bagi suatu instansi atau untuk mencapai tujuannya, hal ini berkaitan dengan kedisiplinan dan mempengaruhi setiap pegawai. Oleh karena itu perlu adanya pengeluaran data khusus untuk melaporkan dan menemukan kegiatan yang dapat dilakukan secara real time dan baik.

Salah satunya menggunakan teknologi komputer sebagai alat bantu untuk mencapai suatu sistem informasi yang baik dalam melaporkan dan menemukan pembaharuan untuk absensi berbasis website. Sistem ini akan menghilangkan proses pencatatan kehadiran pegawai yang masih berjalan secara manual. Di dalam sebuah sistem informasi absensi yang berbasis web, semua data pegawai akan

tersimpan di dalam suatu *database* dan sistem absensi yang akan dibangun ini menggunakan rancangan sistem dengan bahasa pemrograman PHP (*Hypertext Preprocessor*) dan database *MySQL*.

Fungsi model absensi berbasis web, menurut (Andini, 2015) untuk kedisiplinan pada setiap pegawai di perusahaan. Hal ini berfungsi untuk mendapatkan rekapitulasi data absensi dan data izin, cuti, sakit atau tidak hadir.

## **B. Sistem**

Menurut (Tyoso, 2016) “Sistem Informasi adalah sistem fisik dan sosial yang ditata sedemikian rupa untuk mencapai tujuan tertentu”. Dapat disimpulkan bahwa sistem informasi adalah suatu kombinasi teratur dari orang, perangkat keras (*hardware*), Perangkat lunak (*Software*), jaringan komunikasi dan basis data (*database*) yang mengumpulkan, mengubah, dan menyebarkan informasi di dalam suatu bentuk organisasi.

Pemrograman berorientasi objek adalah suatu cara baru dalam berpikir serta berlogika untuk menghadapi masalah masalah yang akan dicoba atasi dengan bantuan komputer. Jadi pemrograman berorientasikan objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasi perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang mana membutuhkan cara berpikir atau logika yang dapat terselesaikan dengan bantuan komputer.

### C. Website / Program

#### Definisi *Internet*

Menurut (Infani, Karinawijaya, 2016) Internet sebenarnya adalah kependekan dari *interconnection-networking* dan dapat diartikan sebagai suatu jaringan komputer yang satu dengan yang lain saling terhubung untuk keperluan komunikasi dan informasi. *Internet* diartikan sebagai jaringan komputer di seluruh dunia yang berisikan informasi dan sebagai sarana komunikasi data yang berupa suara, gambar, video dan juga teks.

#### Definisi *Website*

Menurut (Kuryanti, 2015) *Website* adalah keseluruhan halaman-halaman web yang terdapat dalam sebuah domain yang mengandung informasi. Website biasanya dibangun atas banyak halaman web yang saling berhubungan antara satu halaman web dengan halaman *web* yang lainnya yang disebut *hyperlink*, sedangkan media penghubung berupateks di sebut *hyperteks*. *Web* adalah salah satu layanan yang di dapat oleh pemakai komputer yang terhubung ke internet. *Web* sebagai kumpulan dokumen yang di satukan dan mempunyai hubungan atau *link*. Aplikasi dari *website* dinamis ini sering kali kita temukan di *internet* dewasa ini: Portal berita, blog, situs social networking, dan lain – lain. *Website* ini sendiri dibagi menjadi dua macam, yaitu:

A. *Website* dinamis selalu memiliki program yang bekerja di sisi server karena dalam rentang waktu tertentu konten dari *website* tersebut berubah. Salah satu ciri dari *website* dinamis adalah adanya program yang berjalan disisi server untuk memmanage perubahan data yang ditampilkan oleh *website* dinamis tersebut.

**B.** *Website* statis adalah website yang kontennya statis / tidak berubah-ubah. *Website* statis ini persis seperti brosur. Bedanya, brosur di cetak dan disebar, sedangkan *website* statis di *host* dan diakses melalui *internet*. Sekali *website* statis di online kan di internet, umumnya jarang sekali *website* tersebut merubah kontennya. Sering kali *website* statis ini disusun dari html polos yang antar halamannya dihubungkan dengan *hyperlink* di tanpa pemrograman disisi server karena tujuannya adalah sebatas menampilkan informasi di internet. Aplikasi dari *website* statis sering kali kita temukan di *website company profile*, *personal profile*, *website* penawaran produk, dan semua *website* yang bertujuan melakukan komunikasi satu arah dari pemilik *website* ke pengunjung tanpa diperlukan interaksi.

#### **D. Basis Data**

Sistem basis data menurut (Sukamto dan Shalahuddin, 2014) Sistem basis data adalah sistem terkomputerisasi yang tujuan utamanya adalah memelihara data yang sudah diolah atau informasi dan membuat informasi tersedia saat dibutuhkan. Pada intinya basis data adalah media untuk menyimpan data agar dapat diakses dengan mudah dan cepat. Sistem absensi yang akan dibangun ini menggunakan rancangan sistem dengan bahasa pemrograman PHP (*Hypertext Preprocessor*) dan database MySQL. Menurut (Setiawan, 2017) “MySQL adalah sebuah perangkat lunak sistem manajemen basis data SQL (*Structure Query Language*) atau yang dikenal dengan DBMS (*Database Management System*), database ini multithread, multipengguna. Jadi sebenarnya MySQL merupakan turunan dari SQL yang berfungsi sebagai pengoperasian database khususnya dalam pemilihan atau seleksi

dan pemasukan data, yang memungkinkan pengoperasian atau pengerjaan tersebut dapat dengan mudah secara otomatis”.

Aplikasi basis data yang digunakan dalam *website* ini, antara lain :

### 1. XAMPP

Definisi sederhana dari Xampp adalah perangkat lunak berbasis *web server* yang bersifat *open source* (bebas), serta mendukung di berbagai sistem operasi, baik Windows, Linux, atau Mac OS. Xampp digunakan sebagai *standalone server* (berdiri sendiri) atau biasa disebut dengan *localhost*. Hal tersebut memudahkan dalam proses pengeditan, desain, dan pengembangan aplikasi.

### 2. MySQL (*My Structured Query Language*)

MySQL adalah sebuah DBMS (Database Management System) menggunakan perintah SQL (Structured Query Language) yang banyak digunakan saat ini dalam pembuatan aplikasi berbasis website. MySQL dibagi menjadi dua lisensi, pertama adalah Free Software dimana perangkat lunak dapat diakses oleh siapa saja. Dan kedua adalah Shareware dimana perangkat lunak berpemilik memiliki batasan dalam penggunaannya.

### 3. PHP

PHP adalah bahasa pemrograman khusus berbasis web untuk kebutuhan pada sisi server (*back end*). Sehingga, PHP sangat memungkinkan untuk membuat suatu halaman website menjadi lebih dinamis dengan menerapkan *server-side scripting*. PHP juga mendukung manajemen sistem pada *Oracle*, *Postgresql*, *Microsoft Access*, dan lain sebagainya.

### 4. HTML (*Hypertext Markup Language*)

HTML atau *Hypertext Markup Language* adalah bahasa markup yang

digunakan untuk membuat struktur halaman *website* agar dapat ditampilkan pada *web browser*. Jadi, HTML dapat dianalogikan sebagai pondasi awal dalam menyusun kerangka halaman *web* secara terstruktur sebelum membahas terkait tampilan desain dan sisi fungsionalitas.

### **E. Model Pengembangan Perangkat Lunak**

Penulis melakukan pengembangan sistem dengan metode SDLC (System Development Life Cycle), dengan model Waterfall. Menurut (Sukamto, R. A., dan Shalahudin, 2014)“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 sistem-sistem perangkat lunak sebelumnya”. Menurut (Sukamto, R. A., dan Shalahudin, 2014) “Air terjun (Waterfall) sering disebut juga model sekuensial linier (sequential linear) atau alur hidup klasik. Model air terjun menyediakan pendekatan alur hidup perangkat lunak secara sekuensial atau terurut dimulai dari:

- a. Analisa kebutuhan proses pengumpulan kebutuhan dilakukan secara intensif untuk mengspesifikasikan kebutuhan perangkat lunak agar dapat dipahami perangkat lunak seperti apa yang dibutuhkan oleh *user*, pada tahapan ini perlu di dokumentasikan.
- b. Desain Proses multi langkah yang fokus pada desain pembuatan program perangkat lunak termasuk struktur data, arsitektur perangkat lunak, representasi antarmuka, dan prosedur pengkodean.
- c. Pengkodean Desain harus di translatekan kedalam perangkat lunak, hasil

dari tahapan ini adalah program komputer sesuai dengan desain yang telah dibuat pada tahapan sebelumnya.

- d. Pengujian Pengujian fokus pada perangkat lunak, secara segi logik dan fungsional untuk memastikan bahwa semua bagian sudah diuji, hal ini dilakukan untuk meminimalisir kesalahan dan memastikan keluaran yang dihasilkan sesuai dengan yang diinginkan.
- e. Pendukung (*support*) atau pemeliharaan (*maintenance*) Tidak menutupi kemungkinan sebuah perangkat lunak mengalami perubahan ketika sudah dikirim ke *user*, karena adanya kesalahan yang muncul dan tidak terdeteksi saat pengujian atau perangkat lunak harus beradaptasi dengan lingkungan baru, tahapan pendukung dan pemeliharaan dapat mengulangi proses pengembangan mulai dari analisis spesifikasi untuk perubahan perangkat lunak yang sudah ada, tapi tidak untuk membuat program baru.

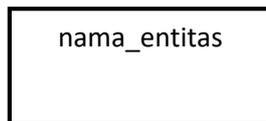
## 2.2 Teori Pendukung

### A. Entity Relationship Diagram (ERD)

Entity Relationship Diagram (ERD) merupakan sebuah cara untuk menggambarkan sebuah basis data yang menggunakan simbol-simbol beserta hubungan antara simbol-simbol tersebut. Menurut (A.S., Rosa dan Shalahuddin, 2015a) “ERD adalah bentuk paling awal dalam melakukan perancangan basis data relasional. Jika menggunakan OODMBS maka perancangan ERD tidak perlu dilakukan”. Sedangkan menurut (Analisis dan Desain Sistem Informasi, 2013) “Entity Relationship Diagram (ERD) adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara abstrak”.

### 1. Entitas / Entity

Entity menunjukkan obyek-obyek dasar yang terkait didalam sistem obyek dasar dapat berupa orang, benda atau hal lain yang keterangannya perlu disimpan dalam basis data.

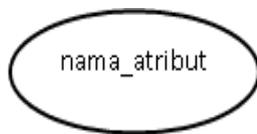


Sumber: (A.S., Rosa dan Shalahuddin, 2015)

**Gambar II.1 Entitas**

### 2. Atribut

Sering juga disebut sebagai properti, merupakan keterangan - keterangan yang terkait pada sebuah entitas yang perlu disimpan sebagai basis data.



Sumber: (A.S., Rosa dan Shalahuddin, 2015)

**Gambar II.2 Atribut**

### 3. Atribut Kunci Primer

*Field* atau kolom data yang butuh disimpan dalam suatu entitas dan digunakan sebagai kunci akses record yang diinginkan; biasanya berupa id.

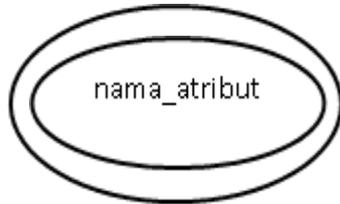


Sumber: (A.S., Rosa dan Shalahuddin, 2015)

**Gambar II.3 Atribut Kunci**

#### 4. Atribut Multinilai / *Multi value*

Atribut Multinilai data yang butuh disimpati dalam suatu entitas yang dapat memiliki nilai lebih dari satu.



Sumber: (A.S., Rosa dan Shalahuddin, 2015)

**Gambar II.4 Atribut Multinilai**

#### 5. Relasi (*Relation*)

Relasi atau hubungan adalah kejadian atau transaksi yang terjadi diantara dua entity yang keterangannya perlu disimpan dalam basis data.



Sumber: (A.S., Rosa dan Shalahuddin, 2015)

**Gambar II.5 Relasi**

#### 6. Asosiasi / *association*

Penghubung antara relasi dan entitas dimana di kedua ujungnya memiliki multiplicity.



Sumber: (A.S., Rosa dan Shalahuddin, 2015)

### **Gambar II.6 Asosiasi / Association**

Relasi yang terbentuk antar entitas dalam Entity Relationship Diagram (ERD) dinamakan derajat relationship. Menurut Ladjamudin (2013:144) mengemukakan bahwa “Derajat Relationship adalah jumlah entitas yang berpartisipasi dalam satu Relationship”. Derajat relationship yang sering dipakai didalam ERD Ladjamudin (2013:145) adalah sebagai berikut:

1. Unary Relationship Unary relationship adalah model yang terjadi diantara entity yang berasal dari entity set yang sama. Sering juga disebut sebagai recursive relationship atau *reflective relationship*.
2. Binary Relationship Adalah model relationship antara instance-instance dari suatu tipe entitas (dua entity yang berasal dari entity yang sama). Relationship ini paling umum digunakan dalam pembuatan model data.
3. Ternary Relationship Ternary Relationship merupakan relationship antara instance-instance dari tiga tipe entitas sepihak.

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

Menurut Priyadi dalam (Hidayat 2017) “Logical Record Structure (LRS) adalah resperentasi dari struktur record- record pada tabel-tabel yang terbentuk dari hasil relasi antar himpunan entitas”. Beberapa tipe record digambarkan oleh kotak empat persegi panjang dan dengan nama yang unik. Beda LRS dengan diagram

entity relationship diagram nama tipe record berada diluar kotak field tipe record ditempatkan.

Selanjutnya menurut Frieyadie (Rahmayu 2015) “LRS merupakan hasil dari pemodelan Entity Relational (ER) beserta atributnya sehingga bisa terlihat hubungan-hubungan antar entitas”. Dari penjelasan diatas dapat ditarik kesimpulan “LRS (Logical Record Structure) adalah hasil dari pemodelan yang terbentuk dari hasil relasi himpunan sehingga menjadi hubungan entitas”.

## **B. Unified Modelling Language (UML)**

Menurut (Sukamto Rosa Ariani dan M. Shalahuddin, 2015) “*UML (Unified Modelling Language)* adalah salah satu standar bahasa yang banyak digunakan didunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasikan objek. Berikut adalah macam-macam diagram:

### *1. Activity Diagram*

Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

### *2. Use Case Diagram*

*Use Case Diagram* merupakan pemodelan untuk kelakuan (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasat, *use case*

digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan apa saja yang berhak menggunakan fungsi-fungsi itu.

### 3. *Class Diagram*

Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

### 4. *Sequence Diagram*

Diagram sekuen menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Membuat diagram sekuen juga dibutuhkan untuk melihat skenario yang ada pada *use case*