

## **BAB II**

### **STUDI PUSTAKA DAN LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Penelitian tentang sistem informasi pendaftaran sebelumnya telah dilakukan oleh Schaduw [5]. Judul “Penerapan Metode *Waterfall* Dalam Sistem Informasi Registrasi Pendakian Gunung Sindoro Berbasis *Web*”. Tujuan dari penerapan sistem pendakian ini yaitu guna menerapkan sistem informasi registrasi pendakian pada gunung Sindoro dengan melibatkan unsur dunia pendidikan. Sedangkan permasalahan yang timbul dalam hal ini adalah kurangnya keefektifan dan efisien dalam melakukan proses pendaftaran. Penelitian ini menggunakan metode SDLC atau biasa disebut dengan metode *waterfall*. Hasil dari penelitian yang telah dilakukan ini yaitu merancang sistem yang mampu membantu pihak pengelola Gunung Sindoro untuk melakukan pendaftaran secara *online*.

Penelitian lain dilakukan oleh Supriyanti [6]. Judul “Paket Wisata Pendakian Gunung Berbasis *Web* Pada PT. Seven *Mountain Adventure* Dengan Menggunakan PHP dan MYSQL”. Tujuan dari penerapan sistem ini yaitu untuk memudahkan suatu instansi atau perusahaan dalam menganalisis sebuah strategi dalam menampilkan data-data agar cepat dan efisien. Sedangkan untuk permasalahan yang terjadi adalah kurangnya kecepatan dan ketepatan dalam mengambil dan menganalisis suatu keputusan dalam mengolah maupun mengambil data. Metode yang digunakan adalah menggunakan metode analisis *SWOT*. Hasil yang diperoleh dari penelitian ini adalah mempermudah manajer sebuah perusahaan dalam melakukan proses pengecekan data secara ringkas, cepat dan efisien.

Penelitian selanjutnya dilakukan oleh Rejeki [7]. Judul penelitian ini “Sistem Informasi Pendaftaran Surat Izin Masuk Kawasan Konservasi (Simaksi) Dan Pemantauan Satwa Berbasis *Web Study Kasus* Pada Gunung Slamet”. Tujuan dari melakukan pengecekan dan pemantauan kegiatan wisata dan melakukan penelitian menggunakan sistem pendaftaran agar kegiatan di kawasan konservasi ini tertib. Salah satu permasalahan yang muncul adalah bagaimana nantinya cara mendapatkan informasi maupun data mengenai wilayah konservasi

Gunung Slamet ini. Metode penelitian ini adalah menggunakan metode *waterfall* dalam perancangan sistemnya. Hasil yang didapatkan adalah memudahkan dalam melakukan pendaftaran *simaksi* atau surat izin masuk kawasan konservasi dan memudahkan dalam memantau satwa maupun pengunjung.

Penelitian dilakukan oleh Sutrisno[8]. Judul “Sistem Informasi Sewa Jasa *Porter* Dan Peralatan Untuk Pendakian Gunung Berbasis *Web*”. Tujuan penelitian untuk membangun sebuah *website* yang digunakan untuk melakukan sebuah transaksi secara *online* yang berfokus pada sewa *porter* dan peminjaman peralatan. Permasalahan awal yang muncul pada penelitian adalah kurangnya informasi dalam jasa penyewaan jasa *porter* dan peminjaman peralatan gunung sehingga pendaki yang masih pemula kesulitan dalam mencari kedua hal yaitu *porter* dan peminjaman peralatan. Metode yang diterapkan dalam penelitian ini menggunakan metode model *SDLC* atau *waterfall*. Hasil yang diharapkan adalah memudahkan pendaki pemula maupun pendaki lainnya dalam memperoleh informasi dan penyewaan secara *online*.

Penelitian terakhir dilakukan oleh Suliswaningsih [9]. Judul “Perancangan Aplikasi Pendataan Pada Pos Pendakian Jalur Gunung Slamet Berbasis *Mobile Android*” . Tujuan dari penelitian yaitu meminimalisir terjadinya suatu kehilangan seperti data, informasi bahkan laporan yang penting dari suatu objek yang terkait dengan data pos pendakian. Permasalahan yang muncul yaitu pada data *basecamp* yang tersedia masih menggunakan sistem manual seperti menulis dengan menggunakan buku besar, pihak *basecamp* khawatir jika data tersebut sampai hilang dan data yang dimasukkan terdapat kesalahan. Penelitian ini menggunakan Metode *Waterfall*. Hasil yang didapatkan adalah merancang sistem yang nantinya bisa memudahkan pihak *basecamp* dalam mengelola data para pendaki yang melakukan pendakian.

Pada penelitian yang akan dirancang dan dikembangkan untuk Sistem Informasi Pendaftaran Online Pendakian Gunung Slamet ini, dilakukan dengan tujuan untuk mempercepat kinerja dalam mengelola sistem, menampilkan sebuah laporan yang lebih jelas, cepat dan akurat. Memberikan kemudahan dalam proses pendaftaran pendakian secara *online* dan memudahkan pendaki dalam melakukan proses pencarian

informasi sehingga pendaki tidak perlu datang langsung ke tempat pendakian.

## **2.2 Landasan Teori**

Penelitian yang dilakukan dilandaskan pada pedoman teori yang berhubungan dengan inti dari penelitian. Berikut adalah pedoman teori yang digunakan sebagai acuan dalam penelitian :

### **2.2.1 Sistem Informasi**

Sistem informasi merupakan media yang diterapkan secara teknologi untuk keperluan perekaman, proses penyimpanan, dan pendistribusian ekspresi dan penalaran pendukung[8]. Sistem informasi bisa dikatakan sebagai kumpulan beberapa komponen yang digabungkan untuk dilakukan proses sehingga bisa mencapai suatu tujuan. Sistem informasi ini terdiri dari beberapa komponen yaitu *input* yang digunakan untuk mendeskripsikan kegiatan yang nantinya akan diproses. Komponen kedua yaitu proses yang juga mendeskripsikan dengan tujuan mendapatkan informasi. Komponen selanjutnya yaitu *output* yang digunakan untuk menghasilkan laporan dari informasi yang telah di proses.

### **2.2.2 Pendaftaran**

Pendaftaran adalah suatu alur, teknik, perilaku mendaftar (mendaftarkan). Penulisan untuk nama, alamat, dan sebagaimana dalam perihal mendaftar[9]. Pengertian dari pendaftaran pada dasarnya bukan hanya untuk mempermudah akan tetapi bisa mengarah pada keefisien dan keefektifan dalam hal waktu, tempat biaya maupun tenaga. Pendaftaran juga digunakan untuk tujuan pencatatan data, sehingga data dan informasi yang telah di daftarkan bisa di catat dan disimpan untuk keperluan administrasi lainnya. Pendaftaran sifatnya memang tidak wajib untuk semua orang akan tetapi pendaftaran dianjurkan agar nantinya memudahkan dalam pencarian data diri pendaftar.

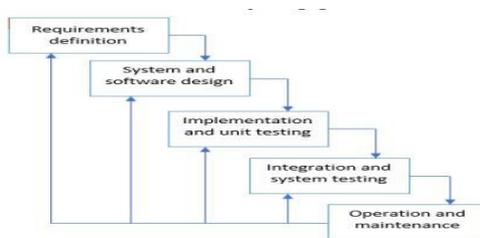
### **2.2.3 Pendakian**

Pendakian adalah perjalanan panjang dan kuat seringkali melibatkan rute atau lokasi yang sulit[10]. Pengertian pendakian di Indonesia mengarah pada perjalanan yang diharuskan melawati puncak gunung. Kegiatan seperti pendakian ini biasanya dilakukan oleh organisasi pencinta alam. Waktu yang digunakan untuk melakukan pendakian ke puncak gunung adalah lebih dari 1 hari, karena perjalanan

pendakian melewati hutan yang rindang, *vegetasi* yang rapat dan permukaan tanah yang tidak rata.

#### 2.2.4 Rekayasa Perangkat Lunak

Rekayasa Perangkat Lunak merupakan sebuah disiplin ilmu yang mencakup semua aspek yang berkaitan dengan proses pengembangan perangkat lunak, mulai dari tahap desain hingga tahapan implementasi sehingga siklus hidup perangkat lunak dapat dilakukan secara efisien dan terukur. Lingkup rekayasa perangkat lunak sangat luas namun masih banyak peserta RPL (Rekayasa Perangkat Lunak) yang mengatakan tidak, ada perbedaan antara RPL (Rekayasa Perangkat Lunak) dengan disiplin ilmu lain seperti ilmu komputer dan sistem informasi [11].



**Gambar 2. 1 Waterfall**

#### 2.2.5 Metode Pengembangan Sistem

Metode yang sering digunakan dalam penelitian adalah metode *waterfall*, karena metode ini sejalan dengan pengembangan sistem informasi. Metode *waterfall* merupakan proses yang digerakan oleh rencana dan prinsip sehingga harus dijadwalkan sebelum dikerjakan sehingga proses pengembangannya berurutan [4].

Berikut tahapan metode *waterfall* menurut Royce yang dilakukan pada pengembangan perangkat lunak :[4]

1. *Requirements analysis and definition*  
Layanan sistem, kendala, dan tujuan didefinisikan dalam konsultasi dengan pengguna yang kemudian didefinisikan sebagai spesifikasi sistem.
2. *System and software design*  
Desain Sistem dan Perangkat Lunak tahap desain sistem menetapkan persyaratan sistem perangkat keras dan perangkat

lunak yang membentuk arsitektur sistem secara keseluruhan. Desain perangkat lunak melibatkan pendefinisian dan penggambaran sistem abstrak dasar perangkat lunak dan hubungannya.

3. *Implementation and unit testing*

Tahap perancangan perangkat lunak ini berlangsung sebagai rangkaian program atau unit program. Pengujian pada tahap ini adalah untuk memverifikasi bahwa setiap unit memenuhi spesifikasinya.

4. *Integration and system testing*

Integrasi dan pengujian sistem Unit individu dari satu atau lebih program digabungkan dan diuji sebagai sistem yang lengkap untuk memastikan bahwa sistem akhir memenuhi atau tidak memenuhi persyaratan perangkat lunak. Setelah pengujian, perangkat lunak dapat dikirim ke pelanggan.

5. *Operation and maintenance*

Terkadang (Meskipun tidak selalu), tahap ini adalah tahap yang panjang. Sistem *diinstal* dan digunakan secara nyata. Pemeliharaan termasuk membangun *bug* yang tidak ditemukan pada fase sebelumnya, meningkatkan implementasi elemen sistem, dan meningkatkan layanan sistem sebagai persyaratan baru.

## 2.2.6 Metode Pengujian Sistem

Pengujian *Black-Box Testing* merupakan teknik pengujian perangkat lunak yang berfokus pada spesifikasi fungsional dari perangkat lunak, *Black-Box Testing* bekerja dengan menggunakan struktur kontrol sehingga difokuskan pada informasi domain[13].

## 2.2.7 Alat Bantu

1. *Flowchart*

*Flowchart* adalah alat pemetaan sederhana yang menampilkan urutan tindakan dalam proses yang mudah dibaca dan dikomunikasikan. Analisis *flowchart* memecah masalah menjadi bagian-bagian yang lebih kecil dan membantu menganalisis alternatif dalam operasi lain[14]. Simbol-simbol yang digunakan dalam *flowchart* sebagai berikut :

**Tabel 2. 1 Flowchart**

NAMA	SIMBOL	ARTI
<i>Input / Output</i>		Menjelaskan <i>input</i> dan <i>output</i> dari pengolahan data.
<b>Proses</b>		Menjelaskan operasi
<b>Anak Panah</b>		Menjelaskan alur kerja
<b>Keputusan</b>		Menjelaskan keputusan dalam program
<i>Terminal Port</i>		Menjelaskan awal dan akhir dari <i>flowchart</i> .
<b>Dokumen</b>		<i>Input dan Output</i> dalam format cetak
<i>Manual Operation</i>		Operasi manual
<i>On-Page Connector</i>		Menghubungkan bagian <i>flowchart</i> pada satu halaman
<i>Off-page Connector</i>		Menghubungkan bagian diagram alur pada halaman berbeda

## 2. UML

*Unified Modeling Language* (UML) adalah alat yang telah menjadi standar di bidang pengembangan sistem perangkat lunak berorientasi objek[15].

Macam-macam UML sebagai berikut :

### a. *Use case* Diagram

*Use Case* Diagram adalah deskripsi fungsionalitas sistem dari perspektif pengguna atau diagram yang menggambarkan actor, kasus pengguna dan hubungan mereka sebagai serangkaian tindakan yang memberikan nilai terukur untuk aktor. *Use case* bekerja dengan menggambarkan interaksi khas antara pengguna sistem dengan sistem itu sendiri, digambarkan dengan cerita bagaimana tentang bagaimana sistem digunakan.

Notasi *use case*:

Tabel 2. 2 Usecase Diagram

Simbol	Deskripsi
	<b>Actor</b> tokoh, alur atau sistem lain yang berhubungan dengan sistem informasi yang akan dibentuk selain sistem informasi yang akan dibuat itu sendirinya.
	<b>Use case</b> Fungsionalitas yang disediakan oleh sistem sebagai pertukaran pesan antar unit aktor, biasanya diekspresikan menggunakan kata kerja diawali dengan frase nama nama <i>use case</i> .
	<b>Association</b> menjelaskan navigasi antar kelas, berapa banyak obyek lain dapat diasosiasikan dengan satu objek, dan apakah satu kelas merupakan bagian dari kelas lainnya.
	<b>Generalization</b> menunjukkan hubungan antara unsur-unsur yang lebih umum dan unsur-unsur yang lebih khusus. Melalui generalisasi, kelas yang lebih spesifik ( <i>subclass</i> ) akan menurunkan properti dan operasi dari kelas yang lebih umum ( <i>superclass</i> ). Notasi ini digunakan pada konsep <i>inheritance</i> .

b. *Sequence Diagram*

*Sequence Diagram* menjelaskan interaksi antar objek yang diatur dalam urutan waktu. Pengertian *sequence diagram* adalah deskripsi langkah demi langkah, yang terdiri dari *timeline* (urutan) perubahan logis yang harus menghasilkan sesuatu sesuai dengan *use case diagram*

Tabel 2. 3 Sequence Diagram

No.	Simbol	Nama	Keterangan
1.		<i>LifeLine</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
2.		<i>Actor</i>	Menggambarkan <i>user</i> atau pengguna.
3.		<i>Message</i>	Spesifikasi dari komunikasi antar <i>objek</i> yang memuat informasi – informasi tentang aktivitas yang terjadi.

4.		<i>Boundary</i>	Menggambarkan sebuah <i>form</i> .
5.		<i>Control Class</i>	Menghubungkan <i>boundary</i> dengan Tabel.
6.		<i>Entity Class</i>	Menggambarkan hubungan kegiatan yang akan dilakukan.

### 2.2.8 Rekayasa Web

Rekayasa *web* adalah proses untuk membuat aplikasi berbasis *web*. Rekayasa *web* terkait dengan rekayasa perangkat lunak dengan konsep dasarnya diterapkan pada aktivitas rekayasa dan manajemen [11].

1. HTML (*Hyper Text Markup Language*)  
HTML adalah singkatan dari *HyperText Markup Language* merupakan bahasa pemrograman yang digunakan untuk membangun halaman *web*. HTML sering digunakan untuk mendesain halaman *web*, meskipun pada kenyataannya HTML tidak terisolasi, karena HTML digabungkan dengan CSS atau skrip lain, seperti *javascript*.
2. MySQL (*My Structured Query Language*)  
MySQL merupakan perangkat lunak sistem manajemen yang basis data SQL multi-pengguna (bahasa Inggris : *database management system*) atau DBMS yang *multithread*, *multi-user*, dengan sekitar 6 juta instalasi di seluruh dunia.  
MySQL AB menyediakan MySQL sebagai perangkat lunak gratis di bawah GNU *General Public License* (GPU), tetapi mereka juga menjualnya di bawah lisensi komersial jika penggunaannya tidak kompatibel dengan pengguna GPL.
3. PHP (*Hypertext Preprocessor*)  
PHP merupakan salah satu bahasa pemrograman digunakan membuat *website* dan merupakan salah satu bahasa pemrograman yang paling populer. Dinamis artinya sebuah *website* dapat berubah tampilan dan isinya sesuai kondisi tertentu. Misalnya PHP dapat menampilkan tanggal saat ini di halaman *web*.

Interaktif artinya bahasa PHP dapat memberikan *feedback* kepada penggunaanya (seperti menampilkan pencarian produk).

### 2.2.9 Basis Data

Basis Data merupakan sistem yang terdiri dari sekumpulan file atau tabel yang saling berhubungan yang memungkinkan banyak pengguna untuk mengakses dan manipulasi file-file ini. Setiap *database* dapat berisi seperti file atau tabel. Konsep sederhana dari *database* kumpulan catatan-catatan atau potongan dari pengetahuan.

#### 1. DBMS (*Database Management System*)

DBMS adalah perangkat lunak untuk mengelola dan memanggil kueri basis data yang jika diterjemahkan kedalam bahasa Indonesia berarti “ Sistem Manajemen Basis Data “. Ada dua tingkat perangkat lunak DBMS, perangkat lunak tingkat tinggi dan perangkat lunak tingkat rendah. Dari dua level tersebut, *High Level Software* adalah *software* yang berkembang dan sering digunakan sekarang antara lain: MySQL, *PostgreSQL* dan *SQLite*. Dua jenis bahasa komputer yang biasa digunakan untuk interaksi dengan DBMS, yaitu : [16]

##### a. DDL (*Data Definition Language*)

DDL atau yang disebut dengan *Data Definition Language* yaitu sekumpulan perintah untuk mendeskripsikan desain dari *database*. DDL merupakan sebuah metode *query* SQL yang digunakan untuk mendefinisikan data dalam *database*.

- 1) *CREATE* :Membuat *database* dan tabel.
- 2) *DROP* :Menghapus tabel dan *database*.
- 3) *ALTER* :Melakukan perubahan struktur tabel yang telah dibuat, baik menambah *field* (*Add*), mengganti nama *field* (*Change*) ataupun menamakannya kembali (*Rename*), dan menghapus *field* (*Drop*).

##### b. DML (*Data Manipulation Language*)

- 1) *INSERT* :Memasukan data
- 2) *UPDATE* :Mengubah data
- 3) *DELETE* :Menghapus data

2. Entity Relationship *Diagram* (ERD)  
*Entity Relationship Diagram* adalah suatu diagram yang biasa digunakan untuk menghubungkan pemodelan sistem. Pemodelan data pada *ERD* ini membantu dalam tahap perancangan dan analisis sebuah *database*.

#### **2.2.10 Framework**

*Framework* secara sederhana dapat diartikan sebagai kumpulan fungsi/prosedur yang dibuat khusus dan *class* yang siap digunakan, sehingga mempermudah dan mempercepat pekerjaan *programmer* tanpa harus membuat fungsi atau *class* dari awal[18].

*CodeIgniter* adalah aplikasi berupa framework PHP *open source* dengan pola MVC (*Model, View, Controller*) , yang dapat digunakan untuk membangun aplikasi *web* dinamis dengan cepat dan mudah. *CodeIgniter* juga memiliki desain, struktur file yang sederhana, sehingga dukungan dokumentasi lengkap membuat *framework codeigniter* ini lebih mudah dipelajari dan dipahami[18].