

## **BAB II**

### **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Sistem informasi berbasis *web* sebagai *portal* alumni telah banyak dikembangkan di institusi pendidikan. Salah satunya pada penelitian sebelumnya yang telah dilakukan dengan judul “Sistem Informasi Pengelolaan Data Alumni Berbasis Web Pada SMK Bina Mandiri 2 Sukabumi”. Sistem informasi ini digunakan untuk membantu dalam menyampaikan informasi kepada alumni sebagai sarana komunikasi antara alumni dan Pihak sekolah. Metode. Pengembangan sistem menggunakan *prototype* serta bahasa pemrograman yang gunakan PHP dan *SQL* serta menggunakan *framework codeigniter*. Proses pengembangan software menggunakan model *waterfall* [1].

Penelitian sebelumnya yang dibuat oleh Rafles Sebayang, Marlyna Infrianty Hutapea dan Roni Jhonson Simamora dengan judul “Sistem Informasi Pendataan Alumni Fakultas Ekonomi Universitas Methodist Indonesia Berbasib *Web*” dengan menggunakan bahasa pemograman PHP dan *MySQL* dan Metode Pengumpulan Data. Sebelum adanya sistem tersebut, pendataan alumni Fakultas Ekonomi Universitas Methodist Indonesia masih dilakukan secara manual dengan menggunakan pengarsipan dalam buku besar, maka diperlukan pengembangan sistem informasi alumni Fakultas Ekonomi Universitas Methodist Indonesia menggunakan fasilitas *media internet* ataupun *website* untuk memberikan kemudahan pengolahan data alumni atau pengarsipan dan kemudahan komunikasi antar alumni.

Penelitian dengan judul “Sistem Informasi Alumni Program Studi Ilmu Komputer FMIPA Universitas Halu Oleo dengan API Berbasis Web” Nur (2021). Metode pengembangan sistem yang digunakan pada penelitian ini adalah metode *Waterfall*. Sistem ini dibuat menggunakan bahasa pemrograman PHP dengan framework laravel dan *MySQL*

sebagai *database*. Metode pengujian dengan menggunakan sistem *black-box* dan *postman* metode pengujian API.

Pada penelitian ini penulis bermaksud merancang sistem informasi alumni di Akmakom *Training Center* dengan menggunakan metode *waterfall*, *framework* Laravel Bahasa pemrograman PHP, dan *database MySQL*. Yang berbeda dari penelitian sebelumnya pada sistem ini juga akan disediakan fitur informasi lowongan pekerjaan sehingga para alumni bisa mendapatkan informasi mengenai lowongan pekerjaan sesudah lulus dari Lembaga Pelatihan Kursus tersebut. Sistem informasi ini diharapkan dapat digunakan baik oleh pihak Akmakom *Training Center* maupun para alumninya.

## 2.2 Landasan Teori

Dalam penelitian ini diperlukan adanya teori-teori yang mendasar untuk menunjang proses penelitian ini. Teori-teori yang digunakan dalam penelitian ini adalah

### 2.2.1 Sistem

Menurut Cahya Lestari dan Arni Muarifah Amri (2020), sistem adalah sekumpulan subsistem, komponen ataupun *element* yang saling bekerja sama dengan tujuan yang sama untuk menghasilkan *output* yang sudah ditentukan sebelumnya [2]. Menurut Jonny Seah (2020), sistem adalah kumpulan/grup dari subsistem/bagian/komponen apapun, baik fisik ataupun nonfisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu [3]. Menurut Erawati (2019), sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu [4].

Menurut Tata Sutabri, sistem memiliki karakteristik yang terdiri dari Komponen Sistem (*Components*) yang saling berinteraksi dan bekerja sama membentuk satu kesatuan, Batasan Sistem (*Boundary*) yang membatasi antara sistem dengan sistem lainnya atau sistem

dengan lingkup luarnya (*Environment*), Penghubung Sistem (*Interface*), Masukan Sistem (*Input*), Keluaran Sistem (*Output*), Pengolahan Sistem (*Procces*) dan Sasaran Sistem (*Objective*) [5].

### **2.2.2 Informasi**

Informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang. Definisi tersebut merupakan definisi informasi dalam pemakaian sistem informasi. Informasi merupakan sesuatu yang dihasilkan dari pengolahan data. Data yang sudah ada dikemas dan diolah sedemikian rupa sehingga menjadi sebuah informasi yang berguna. Adapun karakteristik informasi yang berguna bagi si penerima yaitu relevan, akurat, tepat pada waktunya, lengkap, rangkuman, dapat diverifikasi [5].

### **2.2.3 Alumni**

Menurut Ariansyah, Fajriyah dan Prasetyo (2019) Alumni adalah siswa ataupun mahasiswa yang telah menyelesaikan jenjang pendidikan dengan segala aturannya pada sebuah institusi pembelajarannya, maka ini bisa dikatakan sebagai alumni [6].

### **2.2.4 Sistem Informasi Alumni**

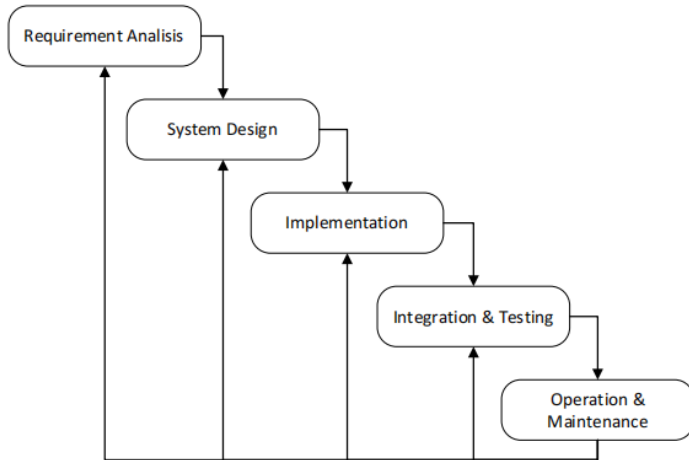
Sistem Informasi Alumni atau bisa disebut dengan *tracer study* adalah suatu sistem yang dirancang untuk mengumpulkan, menyimpan, dan mengelola informasi tentang para alumni (lulusan) suatu lembaga pendidikan atau organisasi tertentu. Tujuan dari sistem ini adalah untuk menjaga dan memelihara hubungan antara institusi atau organisasi dengan para alumni, serta memfasilitasi komunikasi dan pertukaran informasi di antara mereka. Sistem Informasi Alumni biasanya mencakup berbagai data dan informasi tentang para alumni, seperti data pribadi, riwayat pendidikan, pengalaman kerja, pencapaian, aktivitas dan partisipasi, alamat dan kontak lainnya. [7]

### 2.2.5 Website

Menurut Rachdian mengemukakan bahwa ”*website*” adalah suatu pengenalan ruang informasi dimana sumber-sumber daya yang berguna diidentifikasi global yang disebut *Uniform Resource Identifier* (URI) atau lebih dikenal dengan istilah yang lebih populer yaitu *Uniform Resource Locator* (URL)”. Perkembangan *World Wide Web* (WWW) yang sangat pesat ditandai dengan munculnya berbagai macam *website* dengan halaman *web* yang interaktif. Berdasarkan isinya *website* terdiri dari dua jenis yaitu Website Statis (*Static Website*) yang biasanya *user* tidak bisa mengubah *content* dari *web* tersebut secara langsung menggunakan *browser*. Dan Website Dinamis (*Dynamic Website*) yang biasanya *user* dapat mengubah *content* dari halaman tertentu dengan menggunakan *browser* [8].

### 2.2.6 Waterfall

Umesh Kumar Tiwari dan Santosh Kumar (2021) *Waterfall* adalah sebuah contoh dari proses perencanaan, dimana semua proses kegiatan harus terlebih dahulu direncanakan dan dijadwalkan sebelum dikerjakan. Penggunaan model *Waterfall* dalam pengembangan sistem diharapkan mampu memudahkan pembuatan sehingga pembangunan sistem bisa terstruktur. Dalam pengembangannya metode *waterfall* memiliki beberapa tahapan yang berurut yaitu *requirement analisis* (analisis kebutuhan), *system design* (desain sistem), *Implementation*, *Integration & Testing* (pengujian), *Operation and Maintenance* (operasi dan pemeliharaan) [9].



**Gambar 2. 1** Metode *Waterfall*

a. *Requirement Analysis*

Tahap ini pengembang sistem diperlukan komunikasi yang bertujuan untuk memahami perangkat lunak yang diharapkan oleh pengguna dan batasan perangkat lunak tersebut. Informasi ini biasanya dapat diperoleh melalui wawancara, diskusi atau survei langsung. Informasi dianalisis untuk mendapatkan data yang dibutuhkan oleh pengguna..

b. *System Design*

Spesifikasi kebutuhan dari tahap sebelumnya akan dipelajari dalam fase ini dan desain sistem disiapkan. Desain Sistem membantu dalam menentukan perangkat keras (*hardware*) dan sistem persyaratan dan juga membantu dalam mendefinisikan arsitektur sistem secara keseluruhan.

c. *Implementation*

Pada tahap ini, sistem pertama kali dikembangkan di program kecil yang disebut *unit*, yang terintegrasi dalam tahap selanjutnya. Setiap unit dikembangkan dan diuji untuk fungsionalitas yang disebut sebagai *unit testing*.

d. *Integration & Testing*

Seluruh unit yang dikembangkan dalam tahap implementasi diintegrasikan ke dalam sistem setelah pengujian yang dilakukan masing-masing unit. Setelah integrasi seluruh sistem diuji untuk mengecek setiap kegagalan maupun kesalahan.



e. *Operation & Maintenance*

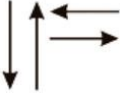


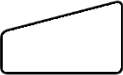
Tahap akhir dalam model *waterfall*. Perangkat lunak yang sudah jadi, dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaikan implementasi *unit* sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

### 2.2.7 Flowchart

*Flowchart* adalah bagan bagan yang mempunyai arus yang menggambarkan Langkah-langkah penyelesaian suatu masalah. *Flowchart* juga merupakan sebuah cara untuk menyajikan algoritma [10]. Berikut beberapa simbol dalam *flowchart*, yang dimuat dalam tabel 2.1

**Tabel 2. 1** Simbol-Simbol dalam *Flowchart*

No.	Simbol	Nama	Keterangan
1.		Terminal	Memulai & mengakhiri suatu program.
2.		<i>Input / Output</i>	Memasukan data dan mennjukkan hasil dari suatu proses tanpa bergantung pada jenis peralatan.

3.		<i>Flow</i>	Menghubungkan antara simbol yang satu dengan yang lain, dengan kata lain menyatakan jalannya suatu proses (disebut juga sebagai <i>connecting line</i> ).
4.		<i>Decision</i>	Memilih proses berdasar pada kondisi yang tersedia.
5.		<i>Process</i>	Pengolahan yang dilakukan oleh komputer
6.		<i>Manual Input</i>	Pemasukkan data secara <i>manual on-line keyboard</i>



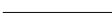

### 2.2.8 Unified Modeling Language

UML (*Unified Modeling Language*) merupakan salah satu standar Bahasa yang banyak digunakan di dunia industri untuk mendefinisikan requirement, membuat analisis dan desain serta menggambarkan arsitektur dalam pemrograman berorientasi objek [11].


#### a. Use Case Diagram

*Use Case Diagram* merupakan pemodelan untuk *behavior* sistem informasi yang akan dibuat. Secara garis besar *use case diagram* digunakan untuk mengetahui fungsi apa saja yang ada dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Simbol-simbol yang ada pada *use case diagram* diantaranya sebagai berikut, pada tabel 2.2.

**Tabel 2. 2** Simbol-simbol *Use Case Diagram*

No.	Simbol	Nama	Keterangan
1.		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil terukur bagi <i>Actor</i> .
2.		<i>Actor</i>	Membuat himpunan peran yang <i>user</i> mainkan. Ketika berinteraksi dengan <i>use case</i> menjadi lebih spesifik.
3.		<i>Association</i>	Menghubungkan objek satu dengan yang lainnya.
4.		<i>Include</i>	Menunjukkan <i>use case</i> sumber secara eksplisit dan spesifik.
5.		<i>Extend</i>	Menunjukkan secara spesifik bahwa <i>use case</i> target memperluas perilaku dari <i>use case</i> sumber pada suatu titik yang diberikan.






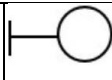
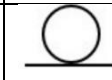
6.		<i>System</i>	Menunjukkan dengan spesifik paket yang menampilkan sistem secara terbatas.
----	---	---------------	--

**b. Sequence Diagram**

*Sequence Diagram* digunakan untuk menggambarkan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan pesan yang dikirimkan maupun diterima antar objek. Oleh karena itu untuk menggambarkan *sequence diagram* maka harus diketahui terlebih dahulu objek-objek yang terlibat dalam sebuah *use case* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu [12]. Simbol-simbol pada *sequence diagram* diantaranya tertera pada Tabel 2.3.

**Tabel 2. 3** Simbol-Simbol *Sequence Diagram*


No.	Simbol	Nama	Keterangan
1.		<i>LifeLine</i>	Objek entitas yang saling berinteraksi.
2.		<i>Actor</i>	Menggambarkan <i>user</i> yang menggunakan sistem.
3.		<i>Message</i>	Menggambarkan komunikasi antar objek yang memuat informasi-informasi tentang aktivitas yang terjadi.

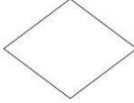

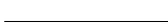
4.		<i>Boundary</i>	Menggambarkan sebuah <i>form</i> .
5.		<i>Control Class</i>	Menghubungkan <i>boundary</i> dengan Tabel.
6.		<i>Entity Class</i>	Menggambarkan hubungan kegiatan yang akan dilakukan.

**c. Entity Relationship Diagram**

*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 sistem. 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* [13]. Simbol-simbol ERD dapat di lihat pada Tabel 2.4 berikut.

**Tabel 2. 4** Simbol-simbol ERD

No.	Simbol	Nama	Keterangan
1.		<i>Entitas</i>	Entitas merupakan data inti yang akan disimpan, bakal tabel pada basis data, benda yang memiliki data dan harus disimpan datanya agar dapat diakses oleh aplikasi komputer.

2.		Relasi	Relasi yang menghubungkan antar entitas. Biasanya diawali dengan kata kerja.
3.		Atribut	<i>Field</i> atau kolom data yang butuh disimpan dalam suatu entitas
4.		Alur	Alur memiliki fungsi untuk menghubungkan atribut dengan entitas dan entitas dengan relasi.

### 2.2.9 PHP

PHP (*Hypertext Preprocessor*) merupakan suatu Bahasa pemrograman yang digunakan untuk menerjemahkan baris kode program menjadi kode mesin yang dapat dimengerti oleh komputer yang bersifat server-side dan dapat ditambahkan kedalam HTML [14].

### 2.2.10 OOP

*Object-Oriented Programming* atau OOP adalah suatu metode programming yang berorientasi pada objek. Program-program tersebut merupakan gabungan dari beberapa komponen kecil yang telah ada sebelumnya, sehingga dapat mempermudah pekerjaan seorang programmer dalam melakukan pengembangan [15].

### 2.2.11 Laravel

Laravel merupakan *framework* PHP yang *open-source* dan berisi banyak modul dasar untuk mengoptimalkan kinerja PHP dalam pengembangan aplikasi *web*. *Laravel* memberikan seperangkat alat dan sumber daya untuk membangun aplikasi berbasis PHP. *Laravel* memiliki ekosistem yang lengkap didukung oleh *package* dan ekstensi yang kompatibel.

Dengan mengikuti pola arsitektur *model-view-controller* (MVC), *development* bisa dilakukan dengan lebih cepat karena *developer* bisa fokus hanya ke salah satu bagian seperti model (bagian yang mengelola *database*), *view* (bagian yang mengelola tampilan kepada *user*), dan bagian *controller* (bagian yang menghubungkan model dan *view* jika seandainya ada permintaan dari *user*).

### **2.2.12 Basis Data**

Basis data adalah kumpulan data yang disimpan secara sistematis dalam komputer dan dapat diolah atau dimanipulasi menggunakan perangkat lunak untuk menghasilkan informasi. Definisi basis data meliputi spesifikasiberupa tipe data, struktur, dan juga batasan-batasan data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam sistem informasi dimana basis data merupakan gudang penyimpanan data yang akan diolah lebih lanjut. Basis data juga digunakan untuk menghindari duplikasi data, hubungan antar data yang tidak jelas, organisasi data, dan juga update data yang rumit [16].