



## **BAB II**

# **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

## **BAB II**

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

#### **2.1. Tinjauan Pustaka**

Sebagai landasan dari penyusunan penelitian ini, penulis melakukan review literatur terhadap artikel-artikel yang sudah ditulis terdahulu oleh peneliti, yaitu berupa hasil dari penelitian yang berhubungan dengan penelitian yang akan ditulis.

Penelitian sebelumnya telah dilakukan oleh Bustomi Abdul Goni [5], dengan judul Sistem Informasi Akademik Di Smk Purnama Bakti Berbasis Java. Sistem Informasi Akademik ini dikembangkan berbasis dekstop yang dapat memudahkan guru dalam menunjang kegiatan proses belajar mengajar. Kekurangan dari sistem akademik ini yaitu diperlukannya pengembangan lebih lanjut agar dapat digunakan lebih baik.

Penelitian lain telah dilakukan oleh Nelfira Nelfira [6], dengan judul Sistem Informasi Akademik Pada SMA N 1 Palembang Berbasis Web Menggunakan Framework Bootstrap. Sistem Informasi Akademik ini dapat mempermudah civitas sekolah dalam pengolahan data akademik menjadi sebuah informasi yang berguna.

Penelitian lain juga dilakukan oleh Fajar Dwi Alfionto [7], dengan judul Sistem Informasi Akademik Sekolah Dasar Negeri Kahuman Ngawen Klaten Berbasis Website. Sistem ini dapat membantu masyarakat maupun orangtua dalam menemukan informasi terkait dengan sekolah, seperti informasi pengambilan rapor, penerimaan siswa baru, perubahan data siswa maupun informasi tentang program sekolah setiap tahun yang ditampilkan pada halaman utama *landing page*.

Sistem yang akan dirancang oleh penulis tentu akan berbeda dengan yang sudah ada sebelumnya. Dibandingkan dengan penelitian sebelumnya, penelitian ini akan memiliki beberapa fitur yang berbeda yaitu fitur menampilkan grafik perkembangan nilai siswa pada SD Tiranus. Pada sistem ini akan berbasis website dan akan dibangun menggunakan bahasa pemrograman php serta menggunakan *framework laravel*. Kelebihan dari menggunakan *framework laravel* yaitu mendukung *MVC (Model-View-Controller)*.

## **2.2. Landasan Teori**

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**

Sistem merupakan susunan yang sengaja dirancang untuk suatu proses yang berjalan secara bertahap, yang memiliki alur jelas dalam penerapannya [8]. Konsep dasar sistem sendiri memiliki arti keseluruhan yang tersusun demi berjalannya suatu proses hingga akhir, antara komponen yang satu dengan komponen lainnya. Berikut ini beberapa pengertian menurut para ahli:

a. Menurut Cambridge Dictionary

Sistem adalah cara untuk melakukan sesuatu. Ini didefinisikan seperti satu set benda atau perangkat yang terhubung dan beroperasi bersama. Pengertian sistem juga bisa diartikan satu set peralatan komputer dan program yang digunakan untuk tujuan tertentu atau sekumpulan organ atau struktur dalam tubuh yang memiliki tujuan tertentu.

b. Menurut Meriam Webster

Sistem adalah interaksi secara teratur atau kelompok item yang saling bergantung membentuk satu kesatuan yang utuh. Sistem juga didefinisikan Meriam-Webster sebagai seperangkat ajaran, gagasan, atau asas yang terorganisasi biasanya dimaksudkan untuk menjelaskan pengaturan atau cara kerja dari keseluruhan yang sistematis.

c. Menurut Ludwig Von Bertalanffy

Sistem adalah suatu kumpulan unsur yang saling berhubungan dan saling mempengaruhi satu lingkup tertentu

### **2.2.2. Informasi**

Informasi adalah sekumpulan data mentah yang dikelola menjadi sesuatu yang bermanfaat bagi penerimanya. Informasi perlu diproses terlebih dahulu agar penerima mudah untuk memahami informasi yang diberikan [9]. Adapun pengertian informasi menurut para ahli:

a. Menurut George H. Bodnar

Pengertian Informasi menurut George H. bodnar, informasi adalah data yang telah diolah menjadi bentuk yang berguna bagi penerimanya dan nyata, berupa nilai yang dapat dipahami di dalam keputusan sekarang maupun masa depan [10].

- b. Menurut Robert N. Anthony dan John Dearden  
Pengertian Informasi menurut Robert N. Anthony dan John Dearden, informasi adalah data yang telah diolah menjadi bentuk yang lebih berguna dan berarti bagi yang menerimanya [11].

### **2.2.3. Sistem Informasi**

Menurut Rudy Tantra, sistem informasi adalah cara yang terorganisir untuk mengumpulkan, memasukan, dan memroses data dan penyimpanannya, mengelola, mengontrol dan melaporkannya sehingga dapat mendukung perusahaan atau organisasi untuk mencapai tujuan [12].

### **2.2.4. Sistem Informasi Akademik**

Sistem informasi akademik adalah suatu sistem untuk mengolah data akademik pada suatu instansi pendidikan, baik formal maupun informal, dari tingkat dasar hingga tingkat perguruan tinggi. Secara umum data yang diolah dalam sistem informasi akademik ini meliputi data guru, data siswa, data mata pelajaran dan jadwal pembelajaran, serta data umum lainnya berdasarkan kebutuhan masing-masing lembaga pendidikan. Secara singkat Sistem Informasi Akademik dapat dipahami sebagai suatu aplikasi yang membantu mempermudah pengelolaan data dan informasi yang berkaitan dengan lembaga pendidikan [13].

### **2.2.5. UML (*Unified Modeling Language*)**

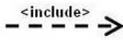
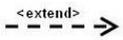
UML adalah sekumpulan alat yang digunakan untuk melakukan abstraksi terhadap sebuah sistem atau perangkat lunak berbasis objek. UML merupakan singkatan dari Unified Modeling Language. UML juga menjadi salah satu cara untuk mempermudah pengembangan aplikasi yang berkelanjutan. Aplikasi atau sistem yang tidak terdokumentasi biasanya dapat menghambat pengembangan karena developer harus melakukan penelusuran dan mempelajari kode program. UML juga dapat menjadi alat bantu untuk transfer ilmu tentang sistem atau aplikasi yang akan dikembangkan dari satu developer ke developer lainnya [14].

#### **1. *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 dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan

fungsi-fungsi itu. Simbol-simbol yang ada pada *use case diagram* dapat dilihat pada tabel 2.1.

**Tabel 2.1** Simbol-simbol *Use Case Diagram*

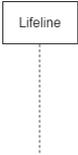
No.	Simbol	Nama	Keterangan
1.		<i>Use Case</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil terukur bagi <i>Actor</i> .
2.		<i>Actor</i>	Membuat Himpunan peran yang <i>user</i> mainkan ketika berintraksi dengan <i>use case</i> menjadi lebih spesifik.
3.		<i>Association</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 seara terbatas.
----	---	---------------	---

## 2. *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 diintansi menjadi objek itu. Simbol-simbol pada *sequence diagram* diantaranya tertera pada tabel 2.2.

**Tabel 2.2** Simbol-simbol *Sequence Diagram*

No.	Simbol	Nama	Keterangan
1.		<i>Lifeline</i>	Objek entitas yang saling berinteraksi.
2.		<i>Actor</i>	Menggambarkan user 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 form.
5.		<i>Control Class</i>	Menghubungkan <i>boundary</i> dengan tabel.
6.		<i>Entitas Class</i>	Menggambarkan hubungan kegiatan yang akan dilakukan

### 3. *Class Diagram*

*Class diagram* adalah jenis diagram struktur statis dalam UML yang menggambarkan struktur sistem dengan menunjukkan sistem *class*, atributnya, metode, dan hubungan antar objek. *Class diagram* disebut jenis diagram struktur karena menggambarkan apa yang harus ada dalam sistem yang dimodelkan dengan berbagai komponen. Berbagai komponen tersebut dapat mewakili *class* yang akan diprogram, objek utama, atau interaksi antara *class* dan objek. Berikut adalah simbol-simbol yang menyusun *class diagram* tertera pada tabel 2.3.

**Tabel 2.3** Simbol *Class Diagram*

No.	Simbol	Nama	Keterangan
1.		<i>Generalization</i>	Hubungan dimana objek anak (descendent) berbagi perilaku dan struktur data dari objek yang ada di atasnya

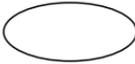
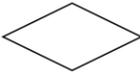
			objek induk.
2.		<i>Nary Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek.
3.		<i>Collaboration</i>	Deskripsi dari urutan aksi-aksi yang ditampilkan sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.
4.		<i>Realization</i>	Operasi yang benar-benar dilakukan oleh suatu objek
5.		<i>Dependency</i>	Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri ( <i>independent</i> ) akan mempengaruhi element yang bergantung pada elemen yang tidak mandiri
6.		<i>Class</i>	Himpunan dari objek-objek yang berbagi atribut serta operasi yang sama.
7.		<i>Association</i>	Yang menghubungkan

			anantara objek satu dengan objek lainnya
--	--	--	--

#### 4. *Entity Relationship Diagram (ERD)*

*Entity Relationship Diagram (ERD)* merupakan diagram yang digambarkan dalam bentuk grafis dalam pembuatan database yang menghubungkan hubungan antar data yang satu dengan data yang lain. Fungsi *Entity Relationship Diagram* adalah untuk membantu pembuatan *database* dan memberikan gambaran terhadap kerja database yang akan dibuat. Berikut adalah simbol-simbol yang menyusun ERD diantaranya tertera pada tabel .

**Tabel 2.4** Simbol-simbol *ERD*

No.	Simbol	Nama	Keterangan
1.		<i>Entity</i>	<i>Entity</i> , simbol yang menyatakan elemen yang didokumentasi menggunakan data.
2.		<i>Attribute</i>	<i>Attribute</i> , simbol yang digunakan untuk menunjukkan nama atribut pada suatu entitas.
3.		<i>Primary Key Attribute</i>	<i>Primary Key Attribute</i> , simbol yang digaris bawahi berfungsi untuk menjadi key diantara beberapa nama atribut yang ada pada suatu entitas.
4.		<i>Relationship</i>	<i>Relationship</i> , simbol yang menyatakan relasi berfungsi untuk menunjukkan hubungan antara entitas yang satu

			dengan entitas yang lain
5.	—————	<i>Link</i>	<i>Link</i> , simbol yang digunakan untuk menghubungkan relasi entitas yang satu dengan entitas dengan atribut yang lain

### 2.2.6. *Laravel*

Laravel adalah sebuah framework web berbasis PHP yang dapat digunakan secara gratis dan open source. Laravel dibuat oleh Otwell untuk pengembangan aplikasi berbasis website dengan menggunakan struktur pola MVC (*Model – View – Controller*) [15].

### 2.2.7. *PHP*

*PHP (Hypertext Preprocessor)* adalah bahasa serverside yang didesain khusus untuk aplikasi web. PHP dapat disisipkan diantara bahasa HTML dan karena bahasa server side, maka bahasa PHP akan dieksekusi di server, sehingga yang dikirimkan ke browser adalah “hasil jadi” dalam bentuk HTML, dan kode PHP anda tidak akan terlihat [16].

### 2.2.8. *Website*

Website adalah media publikasi elektronik yang terdiri dari halaman-halaman web yang saling terhubung satu dengan yang lain menggunakan link yang dilekatkan pada suatu teks atau image. Website dibuat pertama kali oleh Tim Barners Lee pada tahun 1990. Website dibangun dengan menggunakan bahasa *Hypertext Markup Language* [17].

### 2.2.9. *MySQL*

MySQL merupakan software yang tergolong sebagai DBMS (Database Management System) yang bersifat open source. Selain itu, tentu saja bentuk executable-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi dan bisa diperoleh secara gratis di internet [18].

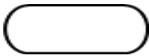
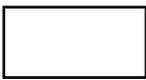
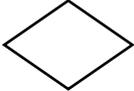
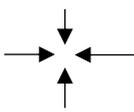
### 2.2.10. Framework

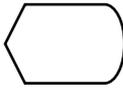
*Framework* adalah koleksi atau kumpulan potongan-potongan program yang disusun atau diorganisasikan sedemikian rupa, sehingga dapat digunakan untuk membantu membuat aplikasi utuh tanpa harus membuat semua kodenya dari awal. *Framework* sendiri berisikan fungsi dasar dan perintah yang lazim dipakai untuk membuat dan mengembangkan sebuah software atau aplikasi, dengan harapan aplikasi yang dibuat bisa dibangun secara lebih terstruktur, lebih cepat serta lebih tersusun dengan cukup rapi [19].

### 2.2.11. Flowchart

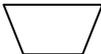
*Flowchart* adalah diagram dengan simbol-simbol grafis yang menunjukkan langkah-langkah yang diwakili oleh simbol-simbol, yang urutannya dihubungkan oleh panah. Diagram ini dapat memberikan solusi langkah demi langkah untuk masalah dalam suatu proses atau algoritma [20]. Berikut ini adalah simbol-simbol dari *flowchart*:

**Tabel 2.5** Tabel Flowchart

No	Simbol	Fungsi
1.		Terminal, untuk memulai dan mengakhiri suatu program.
2.		Proses, Suatu simbol yang menunjukkan setiap pengolahan yang dilakukan oleh komputer.
3.		<i>Input-Output</i> , untuk memasukan data ataupun menunjukkan hasil dari suatu proses.
4.		<i>Decision</i> , suatu kondisi yang akan menghasilkan beberapa kemungkinan jawaban ataupun pilihan (iya atau tidak).
5.		Arus atau <i>flow</i> , prosedur yang dapat dilakukan dari atas ke bawah, bawah ke atas, kiri ke kanan dan kanan ke kiri.

No	Simbol	Fungsi
6.		<i>Display</i> , simbol <i>display</i> digunakan untuk menyatakan <i>output</i> yang dicetak melalui monitor atau layar. Apabila sebuah sistem memiliki keluaran yang tampil dalam sebuah layar.
7.		<i>Preparation</i> , simbol <i>preparation</i> Menggambarkan proses inialisasi data.
8.		<i>Predefined Process</i> , Melakukan permulaan sub program atau proses pengolahan data.

**Tabel 2.6** Simbol-Simbol Aliran Sistem Informasi

NO	Simbol	Nama	Keterangan
1.		<i>Document</i>	Menunjukkan dokumen input dan output untuk manual maupun mekanik atau komputer
2.		<i>Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada satu halaman
3.		<i>Offline Connector</i>	Penghubung bagian-bagian <i>flowchart</i> yang berada pada halaman yang lain
4.		<i>Manual Operation</i>	Menunjukkan pengolahan yang dilakukan secara manual
5.		<i>Manual Input</i>	Memasukan data secara manual <i>on-line keyboard</i> .

*~ Halaman ini sengaja dikosongkan ~*