

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Penelitian Terkait**

Penelitian yang telah dilakukan oleh (Sutisna and Rachman 2021) tentang Sistem Informasi Penerimaan Siswa Baru Berbasis *Website* Dengan Model *Prototype* ini sudah berhasil membantu calon peserta didik baru dalam melakukan pendaftaran dan mempermudah kinerja panitia dalam proses mengelola data siswa baru, namun masih memiliki kekurangan yaitu tidak adanya fitur ujian seleksi *online*.

Penelitian yang telah dilakukan oleh (Badrul and Ardy 2021) dalam jurnal penelitiannya yang berjudul Penerapan Metode *Waterfall* pada Perancangan Sistem Informasi Penerimaan Siswa Baru ini sudah mengurangi kesalahan yang terjadi pada proses penerimaan mulai dari pengisian formulir hingga penerimaan peserta didik baru. Pengembangan untuk kedepannya dapat ditambahkan fitur demo atau panduan untuk peserta didik baru sebelum melakukan pendaftaran.

Selain itu, *website* sistem informasi penerimaan siswa baru yang dikembangkan oleh (Dian Fajri, Wirentake, and M. Julkarnain 2020) masih adanya kekurangan fitur-fitur. Maka dari itu, penambahan fitur-fitur seperti *input* foto dan Penambahan button cari pada menu pengumuman di halaman *user* agar mempermudah dalam mencari data calon mahasiswa baru yang telah lulus atau diterima.

(Supriyadi and Lutfiyana 2020) melakukan Perancangan Sistem Informasi Penerimaan Siswa Baru pada SMA Pusaka 1 Jakarta Berbasis *Website* agar dapat memudahkan calon siswa dalam proses pendaftaran maupun panitia dalam penyimpanan data-data calon peserta didik baru dan lebih cepat untuk mendapatkan informasi-informasi tentang sekolah, namun perlu adanya pengembangan pada desain tampilan dan fasilitas yang ada agar lebih nyaman dan juga memudahkan para pengguna.

## **2.2 Landasan Teori**

### **2.2.1. Sistem Informasi**

Sistem Informasi Sistem informasi merupakan kumpulan dari komponen-komponen yang mengumpulkan, memproses, menyimpan dan menyediakan *output* dari setiap informasi yang dibutuhkan dalam proses bisnis serta aplikasi yang digunakan melalui perangkat lunak, *database* dan bahkan proses manual yang terkait[4].

### **2.2.2. Sekolah**

Sekolah merupakan suatu lembaga pendidikan formal yang sistematis yang melaksanakan pengajaran, bimbingan dan pelatihan dalam rangka membantu peserta didik supaya mampu mengembangkan potensinya secara optimal yang menyangkut berbagai aspek mulai dari moral, spritual, intelektual, emosional, sosial, maupun fisikmotoriknya[5].

### 2.2.3. Penerimaan

Menurut Waridah, “Penerimaan berasal dari kata daftar yang artinya catatan sejumlah nama, angka dan sebagainya yang disusun secara berderet dari atas ke bawah misalnya catatan dan *register*”[6].

### 2.2.4. Penerimaan peserta didik baru

Penerimaan peserta didik baru (PPDB) adalah salah satu proses pendaftaran yang ada di instansi/lembaga yang tentunya harus melalui persyaratan yang telah ditentukan[7].

### 2.2.5. Website

*Website* adalah sekumpulan halaman yang terdiri dari beberapa laman yang berisi informasi dalam bentuk data digital, baik berupa teks, gambar, video, audio dan animasi lainnya yang disediakan melalui jalur koneksi internet[8].

### 2.2.6. Visual Studio Code

Visual Studio Code (VS Code) adalah sebuah teks editor multiplatform yang komplit dan handal buatan *Microsoft*. Selain tersedia untuk *Windows*, juga tersedia untuk versi *Linux* dan *Mac*. Teks editor mendukung banyak bahasa pemrograman seperti JavaScript, *Typescript*, dan Node.js[3].



Gambar 2.1 Visual Studio Code

### 2.2.7. Xampp

XAMPP merupakan sebuah perangkat lunak atau *software* yang bebas (*open source*) yang dapat digunakan di berbagai sistem operasi. XAMPP merupakan kompilasi dari beberapa program. XAMPP juga merupakan *software web server Apache* dan juga terdapat MySQL untuk penyimpanan *database* suatu aplikasi[3].



Gambar 2.2 Xampp

### 2.2.8. Bootstrap

Bootstrap adalah sebuah library *framework* CSS yang dibuat khusus untuk bagian pengembangan front-end *website*. Bootstrap merupakan salah satu *framework* HTML, CSS dan Javascript yang paling populer di kalangan *web developer*, pada saat ini hampir semua *web developer* telah menggunakan bootstrap untuk membuat tampilan *front-end* menjadi lebih mudah dan sangat cepat[3].



Gambar 2.3 Bootstrap

### 2.2.9. Basis Data (*Database*)

Sebuah perangkat lunak sistem manajemen basis data SQL. SQL (*Structured Query Language*) adalah sekumpulan perintah khusus yang digunakan untuk mengakses data dalam *database* relational[3].

### 2.2.10. CodeIgneter

CodeIgneter merupakan *framework* pengembangan aplikasi dengan menggunakan PHP dan *framework* untuk pemrograman menggunakan PHP[9].



Gambar 2.4 CodeIgneter

### 2.2.11. HTML

HTML atau HyperText Markup Language adalah bahasa markup yang digunakan untuk membuat sebuah halaman web. HTML ini merupakan pondasi dasar dari halaman web. File HTML disimpan dengan ekstensi .html dan dapat diakses melalui web browser[10].

### 2.2.12. Pagination

Pagination adalah proses pemisahan konten cetak atau digital ke dalam halaman-halaman terpisah. Untuk dokumen cetak dan beberapa konten daring, paginasi juga merujuk pada proses otomatis penambahan nomor berurutan untuk mengidentifikasi urutan halaman.

### 2.2.13. MySQL

“MySQL adalah program yang mengakses suatu *database* MySQL yang berposisi sebagai *server*” (Haqi, 2019). MySQL adalah sebuah *software database* yang dapat menyimpan data yang sudah dibuat di dalam komputer, *database* dapat dihubungkan ke media internet sehingga dapat diakses dengan jarak jauh, *database* yang bersifat *free* atau bebas langganan yang dapat dilakukan dengan siapa saja tanpa membayar dan membeli kepada pembuatnya[11].



Gambar 2.5 MySQL

### 2.2.14. PhpMyAdmin

PhpMyAdmin adalah perangkat lunak yang ditulis dalam bahasa pemrograman PHP yang digunakan untuk membuat *database* yang diperintahkan untuk membuat tabel menggunakan form yang sudah tersedia PhpMyAdmin atau dapat langsung menuliskan script pada *server* SQL PhpmyAdmin sebagai aplikasi berbasis *web* yang mendukung berbagai operasi MYSQL[11].



Gambar 2.6 PhpMyAdmin

### 2.2.15. CSS

CSS atau *Cascading Style Sheet* adalah bahasa yang digunakan untuk memberikan gaya atau *style* pada halaman *web* yang tidak bisa dilakukan oleh bahasa markup HTML. Dengan CSS, halaman *web* menjadi lebih indah dan membuat *web* menjadi responsif menyesuaikan ukuran layar desktop, tablet dan *smartphone*[12].

### 2.2.16. Javascript

Javascript adalah bahasa pemrograman yang dapat digunakan untuk menangani logika program yang dijalankan pada halaman *web*. Dengan javascript, halaman *web* menjadi lebih interaktif. Rilis terbaru dari javascript pada bulan juni 2018 yaitu ECMAScript 2018[10].

### 2.2.17. Personal Home Page (PHP)

PHP atau Hypertext Preprocessor adalah bahasa pemrograman yang dapat disisipkan ke dalam HTML. PHP pertama kali dibuat pada tahun 1995 oleh Rasmus Lerdorf. PHP merupakan *server-side-programming*, yaitu bahasa pemrograman yang dijalankan di sisi *server*[10].



Gambar 2.7 PHP

### 2.2.18. Figma

Figma adalah aplikasi desain berbasis cloud dan alat prototyping untuk proyek digital. Figma dibuat untuk dapat membantu para penggunanya agar bisa berkolaborasi dalam proyek dan bekerja dalam bentuk tim sekaligus di mana saja.

### 2.2.19. DOMPDF

DOMPDF adalah salah satu library PHP yang berfungsi untuk meng-*convert* HTML menjadi PDF. Biasanya, library ini digunakan ketika ingin menampilkan data *website* yang berupa laporan *HTML* ke dalam bentuk PDF.

### 2.2.20. Unified Modeling Language (UML)

*Unified Modelling Language* (UML) adalah sebuah "bahasa" yg telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem piranti lunak. UML menawarkan sebuah standar untuk merancang model sebuah *system*[13]. Alat bantu yang digunakan dalam perancangan berorientasi objek berbasis UML adalah sebagai berikut:

#### 1. Class Diagram

*Class* diagram merupakan diagram yang umum digunakan dalam pemodelan berbasis UML. *Class* diagram ini bertujuan untuk menggambarkan interaksi antar *class-class* yang ada dalam suatu sistem[1].




Tabel 2.1 *Class Diagram*

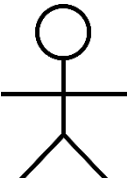
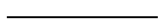
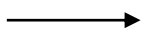
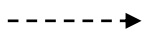
No	Simbol	Pengertian	Keterangan
1.	1	Hanya Satu	Satu dan hanya satu.
2.	0..*	Nol Atau Lebih	Boleh tidak ada satu atau 1 atau lebih
3.	1..*	Satu Atau Lebih	1 atau lebih
4.	0....1	Nola Atau Satu	Boleh tidak ada, maksimal 1.
5.	n...n	n Sampai n	Batasan antara contoh 2..4 mempunyai arti minimal 2 maksimal 4.

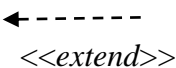
2. *Use Case Diagram*

Sasaran dari pemodelan *use case* diantaranya adalah menerangkan atau mendefinisikan kebutuhan fungsional dan operasional sistem dengan menjelaskan scenario penggunaan yang disepakati oleh *user* dan pengembang. *Use case Diagram* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi – fungsi itu[14].

Tabel 2.2 *Use Case Diagram*

No	Simbol	Pengertian	Keterangan
1.		<i>Use Case</i>	Menggambarkan fungsionalitas yang disediakan sistem sebagai unit-unit yang bertukar pesan antar unit dengan aktif, yang dinyatakan dengan menggunakan kata kerja.

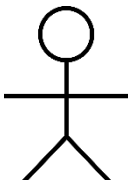

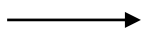



No	Simbol	Pengertian	Keterangan
2.		<i>Actor/Aktor</i>	<i>Abstraction</i> dari orang atau sistem yang lain yang mengaktifkan fungsi dari target sistem. Untuk mengidentifikasikan aktor, harus ditentukan pembagian tenaga kerja dan tugas-tugas yang berkaitan dengan peran pada konteks target sistem.
3.		<i>Association/ Assosiasi</i>	Asosiasi antara aktor dan <i>use case</i> , digambarkan dengan garis tanpa panah yang mengindikasikan siapa atau apa yang meminta interaksi secara langsung dan bukannya mengindikasikan data.
4.		<i>Generalization/ Generalisasi</i>	Asosiasi antara aktor dan <i>use case</i> yang menggunakan panah terbuka untuk Mengindikasikan bila aktor berinteraksi secara pasif dengan sistem
5.	 <<include>>	<i>Include/ Uses</i>	Merupakan di dalam <i>use case</i> lain (required) atau pemanggilan <i>use case</i> oleh <i>use case</i> lain, contohnya adalah pemanggilan sebuah fungsi program.

No	Simbol	Pengertian	Keterangan
6.		<i>Extend/ Ekstensi</i>	Merupakan perluasan dari <i>use case</i> lain jika kondisi atau syarat terpenuhi.

### 3. Sequence Diagram

*Sequence* diagram adalah gambaran visual yang menggambarkan interaksi antara objek-objek dalam urutan waktu. Fungsinya ialah untuk menunjukkan urutan pada *system*[1]. Simbol-simbol yang digunakan dalam *Sequence* Diagram yaitu:





Tabel 2.3 *Sequence* Diagram

No	Simbol	Pengertian	Keterangan
1.		<i>Aktor</i>	Menggambarkan orang yang sedang berinteraksi dengan sistem.
2.		<i>Entity Class</i>	Menggambarkan penghubung antara boundary dengan tabel.
3.		<i>Asynchronous Message</i>	Menggambarkan pengiriman pesan.
4.		<i>Control Class</i>	Menggambarkan penghubung antara boundary dengan tabel.
5.		<i>Boundary Class</i>	Menggambarkan sebuah gambaran dari foem.
6.		<i>A Focus Of Control &amp; A Life Line</i>	Menggambarkan penghubung antara boundary dengan tabel.

#### 4. Activity Diagram

*Activity* diagram menggambarkan *workflow* (aliran kerja) atau aktivitas diari sebuah sistem dan *user*[15]. Simbol-simbol yang digunakan dalam *activity* diagram yaitu:

Tabel 2.4 *Activity* Diagram

No	Simbol	Pengertian	Keterangan
1.		<i>Activity</i>	Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain.
2.		<i>Activity Final Mode</i>	bagaimana objek dibentuk dan dihancurkan.
3.		<i>Initial Node</i>	bagaimana objek dibentuk atau diawali.
4.		<i>Fork Node</i>	Satu aliran yang tahap tertentu berubah menjadi beberapa aliran.