

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka sebuah penelitian sebelumnya yang dapat digunakan sebagai pendukung penelitian yang sedang dilakukan sekarang. Peneliti telah mengumpulkan beberapa tinjauan pustaka yang dapat dilihat pada tabel 2.1 yang ada dibawah.

Tabel 2. 1 Studi Literatur

Nomor	Detail Jurnal	
1	Judul	Implementasi Penerimaan Peserta Didik Baru (PPDB) Berbasis Web pada Sekolah Dasar Alam Semangat Bangsa dan Taman Kanak-kanak Kartini Kecamatan Medan Johor
	Tahun Terbit	2020
	Penulis	Al-Khowarizmi, Halim Maulana , Lutfi Basit
	Metode pengembangan	Metode Waterfall
	Hasil	Dengan diberikannya aplikasi mengenai Penerimaan Peserta Didik Baru (PPDB) berbasis web sekolah untuk mengatasi permasalahan sekolah dalam adanya Pandemi COVID-19 dengan mengurangi aktifitas pertemuan atau dengan arahan pemerintan dengan adanya Sosial Distancing sehingga proses PPDB dilakukan secara online agar tidak berkurangnya minat calon PPDB mendaftar di sekolah dasar alam Semangat Bangsa dan TK Kartini.

Tabel 2. 2 Studi Literatur (Lanjutan)

Nomor	Detail Jurnal	
2	Judul	Sistem Informasi Penerimaan Peserta Didik Baru Online Berbasis Website di Yayasan Pendidikan Arya jaya Sentika
	Penulis	Taufik Hidayat, Mahmudin Muttaqin, Djamaludin
	Tahun Terbit	2020
	Metode Pengembangan	Metode Waterfall
	Hasil	Hasil penelitian ini memungkinkan sistem informasi PPDB untuk melakukan pendaftaran secara online; orang tua calon siswa baru dapat mengetahui biaya dan mendapatkan informasi tentang biaya dana sumbangan Pendidikan, dan orang tua siswa baru akan mendapatkan username dan password untuk login ke sistem.
3	Judul	Rancang Bangun Aplikasi Penerimaan Peserta Didik Baru Berbasis Web Di SMA IT Miftahul Huda Bojongasih
	Penulis	Gian Ramdani, Sulidar Fitri, Alfadl Habibie
	Tahun Terbit	2022
	Metode Penelitian	Metode Waterfall
	Hasil	Hasil penelitian ialah rancangan produk atau rancangan bangun PPDB berbasis web. Penelitian menghasilkan aplikasi PPDB Online berbasis web. Pengujian aplikasi PPDB online berbasis Web menggunakan black box testing.

Tabel 2. 3 Studi Literatur (Lanjutan)

No	Detail Jurnal	
4	Judul	Penerapan <i>E-Learning</i> Dengan Pendekatan <i>Customer Relationship Management (CRM)</i> Berbasis Website Pada Sekolah Menengah Pertama
	Penulis	Wiyanto Wiyanto, Muhammad Najmuddin Dwi Miharja
	Tahun Terbit	2023
	Metode Penelitian	Waterfall
	Hasil	Penelitian ini menghasilkan suatu sistem E-learning berbasis website dengan pendekatan <i>Customer Relationship Management (CRM)</i> dalam proses belajar mengajar antara guru, siswa dan orang tua/wali murid pada Sekolah Menengah Pertama (SMP). Pengujian sistem yang dilakukan peneliti untuk memeriksa apakah aplikasi atau sistem yang dibuat dengan model ISO 9126. Hal ini guna memenuhi persyaratan dan fungsionalitas yang dimaksudkan sesuai dengan desain yang dibuat.
5	Judul	Penerapan <i>Customer Relationship Management</i> Penerimaan Siswa Baru Di Era Pandemi
	Penulis	Noer Azni Septiani, Cut Rusda Mallinda
	Tahun Terbit	2023
	Metode penelitian	Extreme Programming
	Hasil	Hasil pada penelitian yang penulis lakukan tentang penerimaan siswa baru Pada Penerimaan Siswa Baru Di Era Pandemi yang

		diharapkan dapat membantu pihak sekolah dalam memberikan pelayanan kepada calon siswa semaksimal mungkin di era pandemi ini dan memudahkan calon siswa dan wali calon siswa untuk melakukan pendaftaran dan proses pendaftaran lainnya hingga selesai dari mana saja,kapan saja.
--	--	--

Berdasarkan lima *literature* pada Tabel 2.1, maka penelitian ini memiliki perbedaan dari penelitian-penelitian terdahulu, diantaranya adalah sebagai berikut:

1. Pada penelitian ini sistem yang di rancang dan dibangun terdapat berbagai kegiatan para siswa yang bisa diakses oleh para calon peserta dan fitur promosi untuk para calon peserta didik baru dengan memberikannya biaya gratis SPP sampai lulus sekolah ketika melakukan pendaftaran disekolah tersebut.
2. Pada penelitian sebelumnya dalam proses pengembangan sistem tidak ada yang menggunakan metode *Extreme Programming* (XP), dengan menggunakan metode ini karena selain fleksibel metode ini juga dapat menyederhanakan proses saat pengembangan sistem.
3. Pada penelitian sebelumnya hanya menggunakan pengujian *black box* dan ISO 9126. Dalam pengujian sebelumnya belum ada yang menggunakan Standar *ISO/IEC 25010* atau Standart terbaru.

2.2 Penerimaan Peserta Didik Baru (PPDB)

Sebagai bagian dari proses penyelenggaraan pendidikan menjelang tahun ajaran baru, Penerimaan Peserta Didik Baru (PPDB) adalah proses di mana satuan pendidikan memilih calon peserta didik. Sistem informasi PPDB online adalah

aplikasi berbasis website yang digunakan demi menunjang proses pendataan penerimaan peserta didik baru dengan melalui proses entri, dengan menggunakan sistem database terintegrasi, seleksi dilakukan secara digitalisasi dengan program komputer, dan hasil seleksi dapat diakses setiap waktu secara online pada website. Tujuan dari sistem informasi PPDB online ini adalah untuk menyediakan layanan yang cepat, akurat, efektif, efisien, transparan, terintegrasi, dan dapat dipertanggungjawabkan. (Walean & Supit, 2023)

2.3 Pengertian Sistem *E-Enrollment*

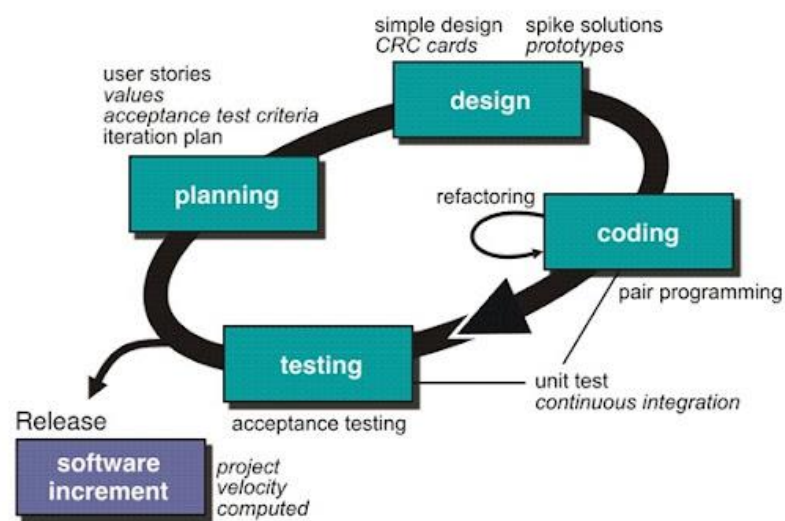
Sistem *E-Enrollment* adalah aplikasi Pendaftaran Online secara elektronik yang digunakan dalam proses pendaftaran peserta didik baru. Penggunaan aplikasi *E-Enrollment*, calon peserta cukup membuka situs web kemudian mendaftarkan diri dengan mengisi formulir pendaftaran, meng-upload data diri, dan melakukan bukti *online payment*. Sistem juga dapat memperbarui data ataupun informasi secara real time sehingga calon peserta didik baru tidak perlu menghubungi secara manual untuk menanyakan ketersediaan kuota pendaftar disekolah tersebut. (Bahri & Dalis, 2018)

2.4 Metode pengembangan sistem

Metode pengembangan perangkat lunak yang dikenal sebagai *Extreme Programming (XP)* bertujuan untuk meningkatkan kualitas perangkat lunak agar sesuai dengan perubahan dan tuntutan pelanggan. Jenis pengembangan perangkat lunak semacam ini dimaksudkan untuk meningkatkan produktivitas dan memperkenalkan pro pemeriksaan dimana persyaratan pelanggan baru dapat diadopsi. *Extreme Programming* memiliki beberapa tahapan, seperti perencanaan (*planning*), seperti mengetahui kriteria pengguna dan merencanakan

pengembangan, desain, seperti membuat prototype dan tampilan, pengkodean, yang juga termasuk dalam pengintegrasian dan yang terakhir pengujian. (Ariyanti et al., 2020).

Pada metode ini terdapat empat tahapan yang harus dilakukan oleh penulis sebelum mengerjakan sebuah perangkat lunak, empat tahapan tersebut adalah sebagai berikut :



Gambar 2.1 Metode Extreme Programming

(Ismail, 2017)

Tahapan-tahapan metode extreme programming menurut (Yusnia Budiarti, 2020) :

1. Tahapan perencanaan (perencanaan) dimulai dengan mendengarkan kebutuhan aktifitas sistem. Ini membantu pengguna memahami proses bisnis sistem dan memahami fitur, fungsionalitas, dan keluaran yang diinginkan. Dalam perancangan sistem pendaftaran peserta didik baru pada tahap ini dimulai dari mengidentifikasi permasalahan yang timbul pada sistem yang

sedang berjalan, kemudian dilakukan analisa kebutuhan pengguna terhadap sistem yang akan dibangun.

2. Design (Perancangan) Pada tahap perancangan, pemodelan sistem dibuat berdasarkan temuan analisis kebutuhan. Selanjutnya, model basis data dibuat untuk menunjukkan hubungan antar data. Unified Modelling Language (UML) adalah bahasa pemodelan sistem yang digunakan. UML terdiri dari *Activity Diagram* dan *Use Case Diagram*. Serta CRC Card yang digunakan untuk mengidentifikasi dan mengorganisasikan object-oriented classes.
3. Coding (pengkodean), perancangan model sistem dimasukkan ke dalam kode program, yang menghasilkan prototipe perangkat lunak. Dalam pembangunan system informasi pendaftaran siswa baru menggunakan bahasa pemrograman PHP, HTML, CSS dan Javascript. Namun, untuk menjalankan basis data, database manajemen sistem yang digunakan adalah MySQL.
4. Testing (Pengujian) Tahapan ini merupakan tahapan pengujian terhadap aplikasi yang sudah dibangun, pada tahapan ini ditentukan oleh pengguna sistem dan berfokus pada fitur dan fungsionalitas dari keseluruhan sistem kemudian ditinjau oleh pengguna sistem. Pada tahapan ini penulis melakukan pengujian terhadap sistem yang sudah dibuat untuk mengetahui apakah ada kesalahan atau yang sering disebut Bug pada sistem saat sistem tersebut dijalankan.

2.5 *Unified Modelling Language*




Unified Modeling Language (UML) adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat

analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek. (Hasanah, 2020)

2.6 Use Case Diagram



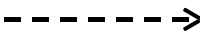
Menurut (Hasanah, 2020), *Use case* diagram adalah teknik untuk merekam persyaratan fungsional sebuah system, menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. *Use case* diagram menekankan kepada “apa” yang diperbuat oleh sistem, dan bukan “bagaimana”. Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. Berikut simbol atau node yang akan digunakan dalam menggambarkan *Use Case Diagram* :

Tabel 2. 4 Use Case Diagram

No.	Simbol	Deskripsi
1	<i>Use case</i> 	Fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antara unit atau aktor.
2	Aktor / <i>actor</i> 	Orang, proses atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
3	Asosiasi / <i>association</i> 	Komunikasi antar aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan aktor.

Sumber : (Hasanah, 2020)

Tabel 2. 5 Use Case Diagram (Lanjutan)



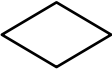


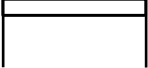
4	Ekstensi / <i>extend</i> <<extend>> 	Relasi <i>use case</i> tambahan kesebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu.
5	Generalisasi / <i>generalization</i> 	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.
6	<i>Include</i> <<include>> 	Relasi <i>use case</i> tambahan ke <i>use case</i> dimana <i>use case</i> yang ditambahkan memerlukan <i>use case</i> ini menjalankan fungsionalnya atau sebagai syarat dijalankan <i>use case</i> ini.

2.7 Activity Diagram

Activity Diagram adalah diagram yang menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Hal yang perlu diperhatikan adalah bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor jadi aktivitas yang dapat dilakukan oleh system. (Hasanah, 2020)

Berikut simbol-simbol yang akan digunakan dalam menggambarkan *activity diagram* dapat dilihat pada tabel 2.4 berikut ini:

Tabel 2. 6 *Activity Diagram*

No.	Simbol	Deskripsi
1	Status awal 	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2	Aktivitas 	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3	Percabangan / <i>decision</i> 	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
4	Penggabungan / <i>join</i> 	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
5	Status Akhir 	Status akhir yang dilakukan sistem sebuah diagram aktivitas memiliki sebuah status akhir.
6	Swimlane 	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

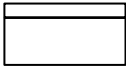

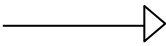
Sumber : (Hasanah, 2020)

2.8 Class Diagram

Class diagram adalah sebuah *class* yang menggambarkan struktur sistem dari segi pendefinisian kelas kelas yang akan dibuat untuk membangun sistem. *Class diagram* juga menjelaskan hubungan antar *class* dalam sebuah sistem yang sedang

dibuat dan bagaimana caranya agar mereka saling berkolaborasi untuk mencapai sebuah tujuan. (Hasanah, 2020)

Tabel 2. 7 *Class Diagram*

No.	Simbol	Deskripsi
1	Kelas / <i>class</i> 	Kelas pada Struktur Sistem
2	Antarmuka / <i>interface</i> 	Sama dengan konsep <i>interface</i> dalam pemrograman berorientasi objek.
3	Asosiasi / <i>association</i> 	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
4	Asosiasi berarah / <i>directed association</i> 	Relasi antar kelas dengan makna kelas kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
5	Generalisasi 	Relasi antar kelas dengan makna generalisasi-spesialisai (umum-khusus).
6	Agresi / <i>aggregation</i> 	Relasi antar kelas dengan makna semua bagian (<i>whole -part</i>).

Sumber : (Hasanah, 2020)

2.9 *Class Responsibility Collaborator Card (CRC Card)*

Menurut (Sari & Cahyani, 2022), CRC-card untuk menjelaskan kelas-kelas yang akan digunakan, beserta fungsi mereka dan interaksi dengan kelas lain. CRC-card (*Class-Responsibility-Collaborator card*) adalah alat yang digunakan dalam pengembangan perangkat lunak berorientasi objek untuk merancang kelas dan menggambarkan tanggung jawab serta kolaborasi antara kelas-kelas yang ada dalam sistem. Setiap kartu CRC mewakili kelas dalam sistem dan berisi informasi tentang peran (*responsibility*) dan hubungan kolaboratif (*collaborator*) kelas tersebut. Berikut ini merupakan gambar bagian-bagian CRC Card :

Class Name	
Responsibilities	Collaborators

Gambar 2. 2 *Class Responsibility Collaborator Card (CRC Card)*

2.10 *PHP Language*

PHP merupakan singkatan dari *Hypertext Preprocessor* yang, *PHP* sendiri dapat dikatakan sebagai bahasa pemrograman *server-side programming*, yaitu bahasa pemrograman yang diproses di sisi server. Fungsi utama *PHP* dalam membangun *website* adalah untuk melakukan pengolahan data pada *database*. Data *website* akan dimasukkan ke *database*, diedit, dihapus, dan ditampilkan pada *website* yang diatur oleh *PHP*. *PHP* berasal dari kata *Hypertext Preprocessor*, yaitu bahasa

pemrograman universal untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan *HTML*. (Nestary, 2020)

2.11 *MySql*

MySQL merupakan sebuah *Relational Database Management System (RDBMS)* yang bersifat *open source*. Perangkat lunak database pada umumnya disandingkan dengan bahasa pemrograman *JAVA*, dan server web seperti *PHP* atau *JSP*. *MySQL (My Structured Query Language)* adalah sebuah program pembuat dan pengelola database atau yang sering disebut dengan *DBMS (Database Management System)*. Selain itu *MySQL* juga merupakan program pengakses database yang bersifat jaringan, sehingga bisa digunakan untuk aplikasi *Multi User*. (Nestary, 2020)

2.12 *HTML Language*

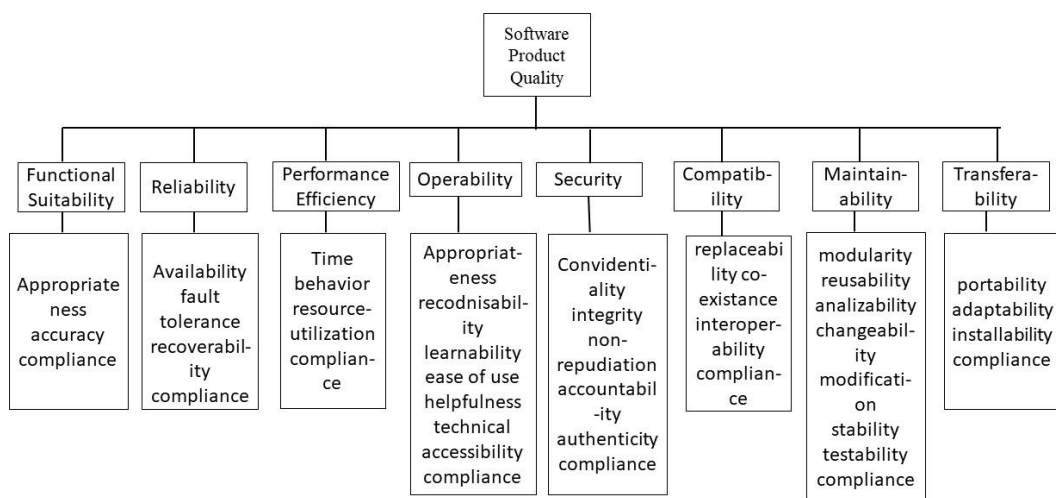
HTML merupakan singkatan dari *Hyper Text Markup Language*, yaitu kode program yang berupa tag-tag untuk membuat dan mengatur struktur atau tampilan dari halaman *website*. Tugas utama dari *HTML* sendiri yaitu sebagai desain dari *website*, misalnya menentukan layout, memformat text dan paragraf, membuat list dan tabel, menyisipkan video maupun audio, dan lain lain. *HTML* bisa digunakan bersamaan dengan *CSS*. (Tumini, 2021)

2.13 Pengujian ISO 25010

Menurut (Tyas et al., 2021) Pengujian ISO 25010 merupakan bagian dari *Systems and Software Quality Requirements and Evaluation (SQuaRE)* yang merupakan versi lanjutan dari ISO 91261, yang telah direvisi secara teknis dengan menambahkan beberapa struktur dan bagian dari standar model kualitas. Tujuan

dari penggunaan kualitas ini adalah untuk mengukur sejauh mana produk atau sistem tersebut bisa digunakan oleh pengguna untuk memenuhi kebutuhan dalam mencapai tujuan yang diinginkan dengan efisiensi, efektivitas, kepuasan dalam konteks penggunaan yang spesifik, dan bebas dari resiko.

Terdiri dari delapan karakteristik yang dibagi menjadi beberapa bagian yang berhubungan dengan sifat-sifat statis perangkat lunak dan sifat dinamis dari sistem komputer, yang dapat ditunjukkan pada gambar dibawah ini :



Gambar 2. 3 Model ISO 25010

Model ISO 25010 adalah Model kualitas adalah landasan dari sistem evaluasi kualitas produk. Model kualitas menentukan karakteristik kualitas mana yang akan diperhitungkan saat mengevaluasi properti produk perangkat lunak.

Berdasarkan gambar diatas, dapat dijelaskan mengenai delapan karakteristik tersebut, sebagai berikut:

1. *Functional Suitability*, Pengujian *functional suitability* merupakan tingkat dimana perangkat lunak dapat menyediakan fungsionalitas yang dibutuhkan ketika perangkat lunak digunakan pada kondisi yang spesifik.

2. *Reliability*, Pengujian *reliability* ini merupakan kemampuan perangkat lunak untuk mempertahankan tingkat kinerja tertentu, ketika digunakan dalam kondisi tertentu.
3. *Performance Efficiency*, merupakan tingkat dimana sistem menyediakan performa yang baik dengan sejumlah *resource* yang akan digunakan pada sistem atau produk.
4. *Usability*, merupakan tingkat dimana pada suatu sistem atau produk mudah dimengerti, mudah dipakai, dan menarik untuk digunakan.
5. *Security*, merupakan tingkat dimana pada suatu sistem menyediakan layanan untuk melindungi akses, ataupun pengungkapan yang berbahaya.
6. *Compatibility*, merupakan kemampuan pada suatu komponen atau sistem untuk bertukar informasi.
7. *Maintainability*, merupakan tingkat dimana pada suatu sistem dapat dimodifikasi, yang meliputi perbaikan, pengembangan untuk menyesuaikan dengan lingkungan, modifikasi pada kriteria, dan spesifikasi fungsi.
8. *Portability*, merupakan tingkat dimana pada suatu sistem dapat dipindahkan dari satu ruang ke ruang lainnya.