

BAB II LANDASAN TEORI

2.1 Tinjauan Pustaka

Tinjauan pustaka yang diambil dari penelitian sebelumnya digunakan sebagai rujukan atau pendukung penelitian yang sedang dilakukan, diantaranya dapat dilihat pada table 2.1 berikut ini:

Tabel 2. 1 Daftar Literatur

No	Detail Jurnal Tinajauan Pustaka	
1	Judul Penelitian	Implementasi Media Visualisasi 360 Pada Platform Android Untuk Promosi Penjualan Kendaraan Bekas
	Penulis Penelitian	(Rokhim dan Lestari, 2019)
	Metode Penelitian	MDLC (Multimedia Development Life Cycle)
	Tujuan Penelitian	Membantu konsumen dalam melihat detail part sepeda motor dan mendapat visulisasi sepeda motor dan mobil yang di pasarkan.
	Hasil Penelitian	Menghasilkan aplikasi katalog motor dan mobil bekas berbasis virtual reality yang dapat membantu salas dalam penjualan produk secara detail.
2	Judul Penelitian	Rancang Bangun Sistem Informasi Pada Bengkel Motor Savana Berbasis Web Menggunakan Metode Xp (Extreme Programming)
	Penulis Penelitian	(Zulianti, Nabyla dan Syauqi, 2020)
	Metode Penelitian	Extreme Programming (XP)
	Tujuan Penelitian	Membangun sistem terkomputerisasi stok barang secara keseluruhan dengan cepat dan tepat.

	Hasil Penelitian	Berhasil membangun sistem informasi bengkel motor berbasis web untuk mempermudah dan mempercepat proses pengolahan data bengkel savanna.
3	Judul Penelitian	Penerapan React JS Pada Pengembangan FrontEnd.
	Penulis Penelitian	(Nasution dan Iswari, 2021)
	Metode Penelitian	Metodelogi SCRUM
	Tujuan Penelitian	Meningkatkan performa pengaksesan data menjadi lebih cepat dan memberikan kemudahan pengembang dalam pembuatan tampilan dengan bantuan package dan library.
	Hasil Penelitian	Berhasil menerapkan React JS dalam pengembangan Front end startup ubaform yang berbasis Single Page Application (SPA) mampu memberikan kemudahan bagi pengembang dan menghemat waktu karena penggunaan package sehingga pengembang tidak perlu melakukan pemrograman setiap komponen dari nol.
4	Judul Penelitian	Sistem Informasi Penjualan Mobil Bekas Berbasis Web Pada Pt. Umi Ford Banjarmasin
	Penulis Penelitian	(Wibowo, 2018)
	Metode Penelitian	Metode Arima Box-Jenkins
	Tujuan Penelitian	Merancang sistem informasi penjualan mobil bekas berbasis web sebagai sarana promosi penjualan pada PT UMI, Ford Banjarmasin.
	Hasil Penelitian	Menghasilkan sistem informasi penjualan mobil berbasis web dan membantu meningkatkan proses promosi penjualan pada PT UMI, Ford Banjarmasin.

5	Judul Penelitian	Evaluasi Functional Suitability , Performance Efficiency , Usability , dan Portability Berdasarkan ISO 25010 pada Aplikasi VR Gamelan Slenthem
	Penulis Penelitian	(Lianto <i>et al.</i> , 2023)
	Metode Penelitian	Metode USE Questionnaire
	Tujuan Penelitian	menjamin kualitas dari aplikasi VR Gamelan Slenthem agar menghindari error serta menjamin berjalannya seluruh fungsi yang terdapat di dalamnya dengan menggunakan standar ISO 25010 selama pengujiannya.
	Hasil Penelitian	Aplikasi VR Gamelan Slenthem telah diuji dan memenuhi standar ISO 25010 pada aspek Functional suitability, Performance efficiency, Usability dan Portability.

2.1.1 Tinjauan Terhadap Literatur

Tinjauan literatur ini menguraikan sejumlah penelitian yang relevan dalam konteks implementasi teknologi dan sistem informasi dalam industri otomotif. Penelitian pertama, yang dilakukan oleh Rokhim dan Lestari (2019), berfokus pada pengembangan aplikasi katalog motor dan mobil bekas berbasis virtual reality dengan menggunakan metode MDLC. Aplikasi tersebut bertujuan untuk membantu konsumen dalam melihat detail produk sepeda motor dan mobil bekas yang dijual.

Penelitian kedua, yang dilakukan oleh Zulianti, Nabyala, dan Syauqi (2020), menggunakan metode Extreme Programming (XP) dalam membangun sistem informasi bengkel motor berbasis web. Tujuan utama dari penelitian ini adalah mempercepat dan memudahkan proses pengolahan data bengkel Savana secara keseluruhan.

Penelitian ketiga, yang dilakukan oleh Nasution dan Iswari (2021), terkait dengan penggunaan React JS dalam pengembangan frontend startup Ubaform. Dalam penelitian ini, metode SCRUM diterapkan untuk meningkatkan performa pengaksesan data dan menyediakan kemudahan bagi pengembang dalam merancang tampilan.

Penelitian keempat yang dilakukan oleh Wibowo (2018) berupaya merancang sistem informasi penjualan mobil bekas berbasis web di PT UMI, Ford Banjarmasin, dengan memanfaatkan metode Arima Box-Jenkins. Sistem ini memiliki peran penting dalam meningkatkan proses promosi penjualan mobil bekas.

Penelitian kelima oleh Lianto et al. (2023) menitikberatkan pada evaluasi aplikasi VR Gamelan Slenthem dengan menggunakan metode USE Questionnaire serta mengacu pada standar ISO 25010. Evaluasi ini dilakukan untuk memastikan bahwa aplikasi VR tersebut memenuhi kriteria kualitas, termasuk Functional suitability, Performance efficiency, Usability, dan Portability.

Dengan mengurai berbagai metode dan tujuan penelitian dalam tinjauan literatur ini, kita dapat memahami beragam pendekatan yang telah digunakan dalam penerapan teknologi dan sistem informasi dalam industri otomotif.

2.2 Keaslian Penelitian

Dari tinjauan literatur di atas, perbedaan utama antara penelitian ini dengan penelitian sebelumnya dapat diidentifikasi sebagai berikut:

- 1) Pendekatan yang diadopsi dalam penelitian ini adalah dengan menggunakan metode Extreme Programming (XP).

- 2) Tahap pengembangan aplikasi melibatkan penggunaan Framework ReactJS.
- 3) Pengujian sistem dilakukan dengan mengacu pada standar ISO 25010.

Selain itu, penelitian ini juga menonjolkan aspek implementasi teknologi Realitas Virtual (VR) dalam pengembangan platform penjualan mobil bekas berbasis web, dengan fokus pada pengalaman pengguna, kepuasan pelanggan, dan tingkat kepercayaan konsumen terhadap transaksi online.

2.3 Sistem Informasi

Sistem informasi adalah suatu struktur yang terdiri dari berbagai bagian yang saling terkait, bekerja bersama, dan berintegrasi satu sama lain dengan tujuan untuk memproses data, menerima data masukan, mengolahnya, dan menghasilkan informasi yang digunakan sebagai dasar untuk pengambilan keputusan (Jannah dan Gevanne, 2021).

2.4 Virtual Reality

Virtual Reality (VR) adalah teknologi simulasi komputer interaktif yang menciptakan lingkungan maya, memengaruhi pengguna dengan informasi indera, sehingga mereka merasa berada dalam simulasi (Gea dan Maulany, 2020). Ini memungkinkan observasi, interaksi dengan objek, dan menciptakan kehadiran virtual. VR merupakan bidang ilmiah dan teknis yang mensimulasikan entitas 3D yang berinteraksi dengan pengguna, memungkinkan aktivitas kognitif dalam dunia buatan digital yang menyerupai dunia nyata melalui perangkat keras dan perangkat lunak komputer.

2.5 Penjualan Mobil Bekas

Mobil bekas adalah kendaraan bermotor yang telah digunakan sebelumnya oleh pemilik sebelumnya dan saat ini ditawarkan untuk dijual kembali. Mobil bekas ini memiliki sejarah pemakaian dan umur yang lebih tua dibandingkan dengan mobil baru, yang membuatnya biasanya dijual dengan harga yang lebih rendah. Pilihan untuk membeli mobil bekas dapat menghemat biaya pembelian, tetapi juga memerlukan perhatian ekstra untuk memastikan keadaan mobil dan riwayat pemeliharannya agar pengguna dapat membuat keputusan yang tepat dalam pembelian mobil bekas (Muhammad, 2018).

2.6 ReactJS

ReactJS adalah pustaka JavaScript yang digunakan untuk membangun komponen antarmuka pengguna (UI) yang modular dan dapat digunakan kembali. React memungkinkan pengembang untuk membuat UI yang efisien dan responsif dengan memisahkan komponen-komponen dalam aplikasi web (Murti dan Sujarwo Badan, 2018). Hal ini membuat pengembangan lebih terstruktur dan memudahkan pemeliharaan serta pengembangan fitur-fitur baru dalam aplikasi.

2.7 CSS

CSS (Cascading Style Sheets) adalah bahasa pemrograman yang digunakan untuk mengatur tampilan dan gaya halaman web. CSS memungkinkan pengembang web untuk mengontrol tata letak, warna, jenis huruf, dan elemen desain lainnya pada halaman web (Andri Nata, Muhammad Iqbal dan Nasrun Marpaung, 2023).

2.8 HTML

HTML (HyperText Markup Language) adalah bahasa yang digunakan untuk mendefinisikan struktur halaman web, termasuk judul, teks, tabel, gambar, dan lainnya. Ini memungkinkan publikasi dokumen online, navigasi melalui tautan hypertext, pembuatan formulir, dan penyisipan konten multimedia langsung ke dalam halaman. Dalam HTML, struktur halaman dijelaskan menggunakan markup atau penanda (Saputra dan Astuti, 2018).

2.9 Visual Studio Code

Visual Studio Code adalah sebuah perangkat lunak editor kode sumber yang ringan namun kuat yang beroperasi di desktop. Ini memiliki built-in dukungan untuk bahasa seperti JavaScript, TypeScript, dan Node.js, serta menyediakan berbagai ekstensi untuk bahasa pemrograman lainnya seperti C++, C#, dan Python. Dengan Visual Studio Code, pengembang dapat membuat dan mengedit kode program dengan efisien (Hartati, 2020).

2.10 Figma

Figma adalah sebuah perangkat lunak desain grafis berbasis web yang digunakan untuk membuat desain antarmuka pengguna (UI), prototipe, dan kolaborasi dalam proyek desain. Ini memungkinkan tim desain untuk bekerja bersama secara online dalam waktu nyata, membuat mockup, menghasilkan prototipe interaktif, dan berbagi proyek dengan mudah (Lunak, Garuda dan Sejahtera, 2022).

2.11 Metode Extreme Programming

Extreme Programming (XP) adalah sebuah metodologi pengembangan perangkat lunak yang sederhana dan fleksibel, cocok untuk proyek-proyek skala

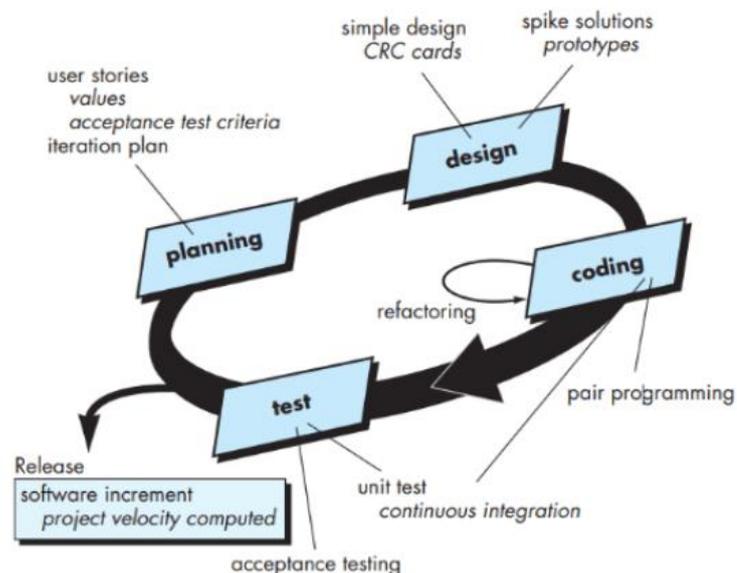
kecil. Metode ini berfokus pada pengkodean sebagai aktivitas utama dalam pengembangan perangkat lunak, dan memungkinkan perubahan yang cepat dan responsif terhadap requirement yang tidak jelas. XP menerapkan prinsip-prinsip agile software development dan melibatkan iterasi berulang untuk memenuhi kebutuhan pengguna serta menghasilkan perangkat lunak berkualitas tinggi (Setiawan dan Ardhiansyah, 2022).

Extreme Programming melibatkan beberapa tahapan dalam rangka kegiatan pengembangan perangkat lunak, termasuk:

- 1) Tahap Perencanaan (Planning): Tahap awal pengembangan perangkat lunak dimulai dengan mengidentifikasi dan merencanakan kebutuhan perangkat lunak yang akan dikembangkan. Ini melibatkan pemahaman konteks bisnis, visi umum mengenai output, dan fungsi inti dari perangkat lunak.
- 2) Tahap Perancangan (Design): Pada tahap perancangan, dipilih pendekatan perancangan yang sederhana daripada yang kompleks. Penggunaan CRC (class responsibility card) digunakan sebagai mekanisme efektif untuk mempertimbangkan perangkat lunak dalam konteks berorientasi objek.
- 3) Tahap Pengkodean (Coding): Ini adalah tahap utama di mana sistem dibangun dengan menggunakan bahasa pemrograman.
- 4) Tahap Pengujian (Testing): Tahap pengujian dilakukan untuk mengidentifikasi kesalahan saat aplikasi berjalan dan memastikan bahwa perangkat lunak yang dibangun sesuai dengan rencana penggunaan awal.

2.11.1 Tahapan Extreme Programming

Tahap XP terdiri dari empat langkah utama, yaitu pemrograman berpasangan, pengujian otomatis, integrasi berkelanjutan, dan pengiriman berulang. Dalam pengiriman berulang, perubahan kode yang melewati proses integrasi terus-menerus dapat segera diterapkan kepada pengguna. Ini memungkinkan pengguna untuk dengan cepat mendapatkan manfaat dari fitur baru dan perbaikan yang diterapkan, sehingga meningkatkan kepuasan pengguna. Selain itu, pengiriman berulang juga membantu mengurangi risiko kesalahan dengan memungkinkan tim pengembang untuk secara teratur memperbarui perangkat lunak. Hal ini membantu dalam mengatasi hambatan yang mungkin muncul selama proses penerbitan perangkat lunak dan mendukung perkembangan yang responsif serta efisien dalam pengembangan perangkat lunak.



Gambar 2. 1 Tahapan Extreme Programming

(Sumber : https://www.researchgate.net/figure/Gambar-1-Alur-Metode-Extreme-Programming-XP_fig1_363662318)

1) Perencanaan (*Planning*)

Tahapan ini adalah tahap yang harus diselesaikan sebelum pengembang mulai membuat sistem. Tahap ini memiliki signifikansi besar karena melibatkan perencanaan atau analisis kebutuhan pengguna saat merancang sistem. Pengembang harus menentukan bagaimana mewujudkan cerita pengguna sesuai dengan kesepakatan yang telah dibuat. Proses pembuatan cerita pengguna dilakukan dengan berbagai langkah, seperti:

- a. Pengguna menceritakan apa permasalahan pada sistem yang digunakan dan sistem seperti apa yang akan dibangun.
- b. Berdasarkan hasil cerita pengguna maka peneliti menentukan poin pada bagian value untuk memutuskan apa saja yang akan dibangun.
- c. Dari hasil kesepakatan tersebut maka peneliti menentukan *acceptance test criteria* yaitu menentukan kriteria apa saja yang nantinya sebagai acuan terhadap sistem yang akan diuji.
- d. Sehingga hasil peneliti menyimpulkan berapa kali akan dilakukan release dan perbaikan pada tahap *iteration* plan merencanakan berapa kali akan dilakukan uji terhadap sistem yang dibangun.

2) Perancangan (*Design*)

Setelah tahap perencanaan selesai, langkah selanjutnya adalah tahap desain. Pada tahap ini, pengembang melakukan perancangan dengan membuat berbagai model, yang dimulai dengan pemodelan sistem, diikuti dengan pemodelan arsitektur, dan pemodelan basis data. Hal ini bertujuan untuk memberikan gambaran yang jelas mengenai sistem yang akan dibangun.

- a. Simple Design adalah pengembang membangun perangkat lunak dengan desain yang sederhana. Dimulai dengan desain yang sederhana dilakukan menggunakan UML seperti Use Case Diagram, Activity Diagram dan Class Diagram.
- b. CRC Card digunakan untuk mengenali dan mengatur object oriented class yang sesuai dengan pengembangan. Jika pada saat perancangan terdapat ketidak sesuaian maupun perbaikan maka akan dilakukan.
- c. *Prototype* adalah bagian perancangan berupa user interface dalam bentuk *wireframing* untuk mempermudah pengguna melihat desain sistem.

3) Pengkodean (Coding)

Dalam proses pengkodean, peneliti melakukan penyesuaian terhadap cerita pengguna (user story) untuk memastikan bahwa sistem yang sedang dibangun sesuai dengan kebutuhan. Proses pengkodean ini mencakup langkah-langkah berikut:

- a. Pair Programming merupakan tahap sistem dibangun dengan bahasa pemrograman dan media penyimpanan yang telah disepakati.
- b. Refactory merupakan tahapan yang dilakukan ketika terjadi ketidak sesuaian kode program sehingga dilakukan perbaikan guna mendapatkan hasil yang sesuai.

4) Pengujian (Testing)

Pada tahap pengujian, pengguna sebagai pengguna akhir melakukan pengujian sesuai dengan acceptance test yang telah ditetapkan dan disetujui sebelumnya. Unit test yang telah dibuat memfokuskan pada semua fitur dan

fungsi sistem secara keseluruhan. Dengan demikian, sistem dapat dinilai telah sesuai dengan standar yang ditentukan dan siap untuk dirilis.

2.12 UML (Unified Modeling Language)

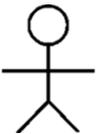
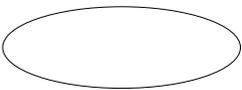
UML (Unified Modeling Language) adalah bahasa standar untuk merancang dan mendokumentasikan perangkat lunak dan sistem. Ini menggunakan berbagai diagram untuk menggambarkan struktur, fungsi, dan interaksi dalam proyek pengembangan perangkat lunak, memfasilitasi komunikasi tim dan dokumentasi proyek.

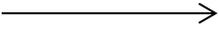
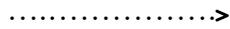
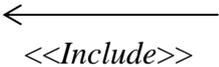
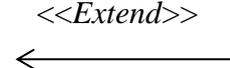
Ada beragam jenis diagram yang dapat ditemui dalam Unified Modeling Language (UML), seperti yang berikut:

2.12.1. Use Case Diagram

Diagram use case mengilustrasikan fungsionalitas yang diantisipasi dari sistem, dengan fokus pada apa yang dilakukan sistem, bukan bagaimana melakukannya. Ini memiliki kegunaan yang besar dalam mengklarifikasi persyaratan sistem, berkomunikasi dengan klien, dan merancang uji kasus untuk semua fitur sistem. Dalam proses internal, satu use case juga dapat mencakup fungsionalitas dari use case lain. Simbol-simbol yang digunakan dalam diagram use case dapat dirujuk dalam tabel 2.2 berikut.

Tabel 2. 2 Usecase Diagram

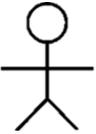
Simbol	Nama	Keterangan
	Aktor	Mewakili individu, organisasi atau sistem lain yang berinteraksi dengan sistem yang dibangun.
	Usecase	Aktivitas atau interaksi antara aktor dengan sistem yang dibangun.

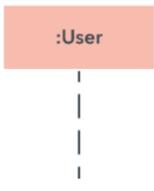
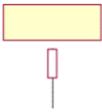
	Association	Keterkaitan antara dua atau lebih elemen dalam diagram usecase.
	Generalitation	Menunjukkan kemampuan aktor untuk berpartisipasi dalam use case
	Include	Menunjukkan bahwa satu <i>usecase</i> sepenuhnya bergantung pada <i>usecase</i> dan saling terkait.
	Extend	Menunjukkan bahwa <i>usecase</i> dapat memperluas atau menambah aktivitas pada <i>usecase</i> lain.

2.12.2 Sequence Diagram

Diagram urutan (Sequence Diagram) mengilustrasikan komunikasi antara berbagai objek dalam sistem dan sekitarnya, termasuk interaksi dengan pengguna, layar, dan lainnya, melalui pesan yang menggambarkan perilaku dalam penggunaan kasus tertentu. Diagram ini menjelaskan bagaimana objek hidup dalam waktu tertentu serta pesan yang mereka kirim atau terima. Simbol-simbol yang digunakan dalam Sequence Diagram dapat diacu dalam tabel 2.3 berikut.

Tabel 2. 3 Sequence Diagram

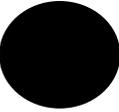
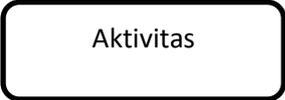
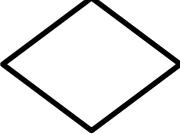
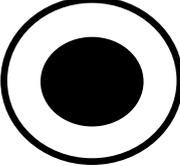
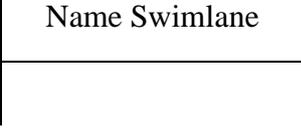
Simbol	Nama	Keterangan
	Aktor	Mewakili peran orang, sistem, atau alat dalam berinteraksi di dalam dan di luar
	<i>Activation Box</i>	Merepresentasikan waktu yang dibutuhkan suatu objek untuk menyelesaikan tugas tertentu
	Objek	Mendemonstrasikan bagaimana sebuah objek akan berperilaku dalam sistem

	<i>Lifelines</i>	Menunjukkan kejadian berurutan yang terjadipada sebuah objek
	<i>Synchronous Message</i>	Menunjukkan bahwa pengirim harus menunggu respon pesan sebelum melanjutkan aktivitas
	<i>Asynchronous Message</i>	Menunjukkan bahwa pengirim tidak memerlukan respon pesan sebelum melanjutkan aktivitas
	<i>Reply Message</i>	Menunjukkan balasan untuk sebuah panggilan tertentu
	<i>Delete Message</i>	Menghapus sebuah objek
	<i>General</i>	Merepresentasikan entitas tunggal berupa kelas
	<i>Message Return</i>	Penggambaran hasil dari pengiriman pesan

2.12.3 Activity Diagram

Diagram Aktivitas mengilustrasikan berbagai kegiatan yang terjadi dalam sistem yang sedang dibangun. Ini mencakup awal masing-masing alur, keputusan yang mungkin diambil, serta bagaimana alur-alur tersebut berakhir. Selain itu, Diagram Aktivitas juga memvisualisasikan tahapan-tahapan yang dapat terjadi secara paralel dalam beberapa eksekusi. Simbol-simbol yang digunakan dalam Diagram Aktivitas dapat ditemukan dalam tabel 2.4 di bawah ini.

Tabel 2. 4 Activity Diagram

Nama	Simbol	Keterangan
Status awal		Sebuah diagram aktivitas yang memiliki status awal.
Aktivitas		Aktivitas yang dilakukan oleh system yang biasanya diawali dengan kata kerja.
Decision		Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
Join		Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
Final State		Menunjukkan bagian akhir dari aktivitas.
Swimlane		Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

2.12.4 Class Diagram

Diagram Kelas adalah sebuah representasi yang berfungsi sebagai panduan untuk menciptakan objek dan merupakan elemen sentral dalam perancangan dan pengembangan berorientasi objek. Diagram ini menggambarkan kondisi sistem dan juga menyediakan metode untuk mengubah kondisi tersebut. Diagram Kelas memiliki tiga komponen utama, yaitu nama, atribut, dan metode. Simbol-simbol yang digunakan dalam Diagram Kelas dapat ditemukan dalam tabel 2.5 di bawah ini.

Tabel 2. 5 Class Diagram

Simbol	Nama	Keterangan
	<i>Association</i>	Yang menghubungkan satu objek dengan objek lainnya
	<i>Association</i>	Upaya untuk menghindari asosiasi dengan lebih dari 2 objek
	<i>Class</i>	Kumpulan objek dengan atribut dan operasi yang sama
	Collaboration	Deskripsi urutan kegiatan yang dilakukan oleh sistem, yang memberikan hasil yang terukur bagi aktor
	Realization	Operasi yang sebenarnya dilakukan oleh objek
	Dependency	Hubungan dimana perubahan pada elemen independen akan mempengaruhi elemen yang bergantung pada elemen yang tidak independen.

2.13 MySql

MySQL adalah sebuah sistem manajemen basis data (DBMS) relasional yang open-source. Ini digunakan untuk menyimpan, mengelola, dan mengakses data dalam berbagai aplikasi, terutama aplikasi web. MySQL menggunakan bahasa SQL (Structured Query Language) untuk mengelola database dan

memungkinkan pengguna untuk melakukan berbagai operasi seperti penyimpanan, pengambilan, pembaruan, dan penghapusan data (Mulyati, 2019).

2.14 ISO 25010

ISO 25010 adalah standar internasional yang mengatur kualitas perangkat lunak dan sistem informasi. Ini memberikan panduan untuk mengukur dan mengevaluasi berbagai aspek kualitas perangkat lunak, seperti fungsionalitas, keamanan, kinerja, keandalan, dan kegunaan, untuk memastikan produk memenuhi standar yang ditetapkan dalam pengembangannya. Standar ini penting dalam memastikan perangkat lunak berkualitas tinggi (Lianto *et al.*, 2023).