

BAB II LANDASAN TEORI

2.1. Tinjauan Pustaka

Dalam penelitian ini, menggunakan beberapa penelitian sebelumnya untuk mendukung penelitian yang sedang dilakukan. Berikut merupakan penelitian sebelumnya yang dapat dijadikan tinjauan pustaka pada penelitian ini, adapun referensi dalam penelitian ini :

2.1.1. Tinjauan Terhadap Literatur 1

(Zia Rizki Saputri *et al.*, 2019) Rancang Bangun Sistem Informasi Pemesanan Makanan Berbasis Web Pada Cafe Surabiku. Aplikasi ini dapat mengolah data produk, stok, konfirmasi transaksi, dan melihat laporan penjualan yang dilakukan secara online. Serta mempermudah konsumen dalam mencari produk, mendapatkan informasi produk, melakukan transaksi, dan melihat riwayat pesanan. Pembuatan aplikasi pada toko ini menguntukkan metode *Waterfall* dan diuji dengan metode *Black-Box Testing* serta menggunakan *framework Codeigniter* yang berbasis web. Hasil dari penelitian dan pembuatan aplikasi ini yaitu Dengan adanya rancang bangun aplikasi pemesanan berbasis web ini dapat mempermudah antara pihak pembeli dan pihak petugas. Sistem ini mempunyai jenis pelayanan diantaranya pemesanan makanan dan minuman melalui Web yang sudah disediakan. Hal ini akan membantu efektifitas waktu pekerja menjadi lebih cepat dan efisien.

2.1.2. Tinjauan Terhadap Literatur 2

(Miyadi, 2019) Membangun sistem informasi pemesanan meja dan makanan berbasis web (studi kasus rumah makan mbak tuti). Sistem informasi ini dibuat dengan menggunakan metode Extreme Programming untuk tahap pengujian. Sistem informasi berbasis web ini dibangun dengan PHP, CodeIgniter, MYSQL dan Apache. Tujuan dari analisis ini

yaitu untuk memesan meja dan makanan dan memudahkan pelanggan melakukan pesanan tanpa harus datang ke tempat.

2.1.3. Tinjauan Terhadap Literatur 3

(Pratama *et al.*, 2022) membuat Pemanfaatan Qr Code Pada Sistem Transaksi Kuliner Berbasis Web (Studi Kasus : Cafe Society Probolinggo) *Utilization of Qr Code in Web-Based Culinary Transaction System* (Case Study: Cafe Society Probolinggo), yang bertujuan untuk merancang sistem penjualan secara merata dan kemudahan transaksi pelanggan tanpa harus datang langsung ke tokonya. Perancangan ini menguntukkan metode *PIECES* yang diuji dengan metode Black-Box Testing dan menguntukkan framework Laravel. Hasil dari penelitian ini adalah Aplikasi Pemesanan Makanan Online di Café Society ini dapat diakses secara online oleh admin, bagian kasir, dan pelanggan dengan menggunakan email dan password (bagi pelanggan tidak perlu).

2.1.4. Tinjauan Terhadap Literatur 4

(Arta, Sudana and Putri, 2022) *Workshop Marketplace System Using Rajaongkir API, Leaflet API and Midtrans Payment Gateway*. Sistem ini dikembangkan dengan berbasis web yang menggunakan bahasa pemrograman PHP dengan framework Laravel serta diintegrasikan dengan *Leaflet API, RajaOngkir API, dan Midtrans Payment Gateway* yang bertujuan untuk mengatasi permasalahan pasar saat ini, seperti kurangnya fokus pada bidang pasar dan sulitnya mendapatkan informasi tentang bengkel-bengkel sekitar. Hasil penelitian berupa sistem marketplace bengkel yang diuji dengan metode black box menunjukkan bahwa seluruh menu dapat berjalan normal tanpa error/bug.

2.1.5. Tinjauan Terhadap Literatur 5

(Kurniawan, 2021) Pengembangan Aplikasi *E-Order Service* Berbasis Website Dengan Memanfaatkan Payment gateway (Studi Kasus: Hotel Lava View), Proses pengembangan Aplikasi E-Order service yang dibuat megunakan framework *codeinighter* dengan mengguakan teknologi *payment gateway* sebagai API pembayaran sehingga dapat menjadi solusi dengan pencegahan penyebaran virus corona sesuai dengan protokol kesehatan.

2.1.6. Tinjauan Terhadap Literatur 6

(Lawrenxius Benny, Joseph Eric Samodra and Yonathan Dri Handarkho, 2022) Pengembangan Sistem Pemesanan Makanan di Roemah Soto Berbasis Web Service Dengan Penerapan Payment Gateway Metode waterfall digunakan dalam pengembangan sistem di Roemah Soto. Hasil dari penelitian ini adalah sebuah sistem dengan aplikasi mobile Android yang memungkinkan pelanggan untuk mengolah makanan, terhubung dengan payment gateway dengan sistem pembayaran *cashless* dan website administrasi untuk dikelola. Informasi dari Roemah Soto agar dapat mendukung proses bisnis.

2.2. Keaslian Peneliti

Adapun beberapa hal yang menjadi pembeda antara penelitian yang dilakukan penulis dengan penelitian yang dilakukan sebelumnya sebagaimana sudah terlampir pada tinjauan pustaka, diantaranya adalah :

- 1) Penempatan pada penelitian ini terletak di Eyzell Kaffe yang dimana belum ada yang melakukan penelitian di Eyzell Kaffe.

2.3. Sistem Informasi

Menurut Darmawan Deni dan K.N Fauzi yang dikutip oleh Wahyu Nugraha dan Muhamad Syarif (2018) menyatakan bahwa sistem Informasi merupakan kumpulan sub-sub sistem yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan, yaitu mengelola data menjadi informasi yang berguna.

Hidayat dan Muttaqin (2018) mengatakan bahwa sistem Informasi merupakan suatu asosiasi terdiri dari beberapa modul yang saling terintegrasi dimana menyajikan informasi dan pengolahan data untuk disajikan sesuai dengan kebutuhan user, database, source code dan model design mewakili dari untaian suatu rancangan sistem informasi.

2.4. Kafe

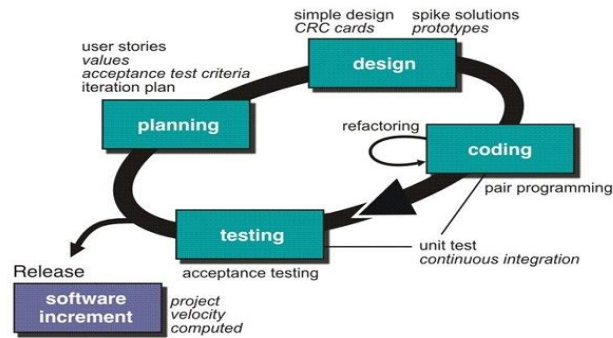
Menurut (Zia Rizki Saputri *et al.*, 2019) Kafe (atau kopi cafe) adalah tempat usaha yang menyajikan berbagai jenis minuman dan makanan, terutama kopi, untuk dikonsumsi di tempat atau untuk dibawa pulang. Biasanya kafe juga menyediakan tempat duduk dan area yang nyaman untuk para pelanggannya bersantai atau bekerja sambil menikmati minuman atau makanan yang disajikan Extreme Programming.

2.5. Extreme Programming (XP)

Extreme Programming adalah merupakan salah satu metodologi dalam rekayasa perangkat lunak dan juga merupakan satu dari beberapa asile software development methodologies yang berfokus pada coding sebagai aktivitas utama di semua tahap pada siklus pengembangan perangkat lunak (software development lifecycle) (Miyadi, 2019).

2.6. Tahapan Metode Extreme Programming (XP)

Tahapan Metode Extreme Programming (XP) ada empat langkah adalah sebagai berikut :



Gambar 2.1 Tahap Metode Extreme Programming.

Sumber (machlizadevi.blog.binusian.org, Akses pada tanggal 11/02/2023)

Menurut (Borman, Priandika and Edison, 2020) ada 4 tahapan yaitu :

1. Perencanaan (*Planning*)

Pada tahap ini, beberapa kegiatan perencanaan dilakukan, mulai dari mengidentifikasi permasalahan, menganalisis kebutuhan hingga menentukan jadwal pelaksanaan pembangunan sistem. Pengembang memutuskan bagaimana membangun hasil cerita pengguna sesuai dengan janji yang disepakati.

2. Perancangan (*Design*)

Pada tahap ini, beberapa kegiatan perencanaan dilakukan, mulai dari mengidentifikasi permasalahan, menganalisis kebutuhan hingga menentukan jadwal pelaksanaan pembangunan sistem.

3. Pengkodean (*Coding*)

Pada tahap ini, pengembang mulai membuat kode program dengan menyesuaikan kebutuhan fungsional dan desain detail yang telah ditetapkan sebelumnya. XP mendorong pengembang untuk mengembangkan kode dengan teknik pengkodean yang bersih dan terstruktur, menggunakan metode pengujian otomatis untuk

memastikan kualitas kode dan meningkatkan kemudahan pengujian di masa depan.

4. Pengujian (Testing)

Pada tahap ini, pengembang mulai membuat kode program dengan menyesuaikan kebutuhan fungsional dan desain detail yang telah ditetapkan sebelumnya. XP mendorong pengembang untuk mengembangkan kode dengan teknik pengkodean yang bersih dan terstruktur, menggunakan metode pengujian otomatis untuk memastikan kualitas kode dan meningkatkan kemudahan pengujian di masa depan.

2.7. UML (Unified Modeling Language)

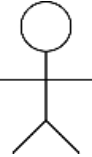
UML merupakan standar bahasa yang banyak dipakai di industri untuk menggambarkan kebutuhan, melakukan analisis dan desain, serta menggambarkan arsitektur dalam pengembangan perangkat lunak berorientasi objek (Putra and Andriani, 2019).

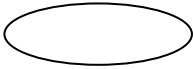

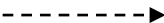
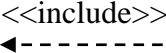
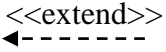
2.7.1. Use Case Diagram

Use case diagram merupakan pemodelan untuk kelakuan sistem informasi yang akan dibuat. Use case bekerja dengan mendeskripsikan tipikal interaksi antara user sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sistem itu dipakai (Putra and Andriani, 2019).

Simbol yang digunakan dalam use case diagram dapat dilihat pada 2.1

Tabel 2.1 Use Case Diagram.

No	Simbol	Nama	Keterangan
1		Aktor	Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan use case

2		Use Case	Abstraksi dan interaksi antara sistem dan aktor
3		Association	Abstraksi dari penghubung antara aktor dan use case
4		Generalisasi	Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan use case
5		Include	Menunjukkan bahwa suatu use case seluruhnya merupakan fungsionalitas dari use case lain
6		Extend	Menunjukkan bahwa suatu use case merupakan tambahan fungsi

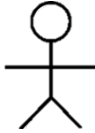


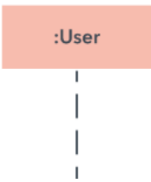

Tabel 2. 1 Use Case Diagram (lanjutan)

2.7.2. Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku objek dalam use case dengan menggambarkan urutan waktu hidup objek dan pesan yang dikirimkan dan diterima antara objek. Diagram ini dibuat setidaknya untuk setiap *use case* yang memiliki proses sendiri atau penting, dan harus mencakup interaksi pesan dari semua use case yang telah didefinisikan. Sehingga, semakin banyak *use case* yang didefinisikan, semakin banyak pula *Sequence diagram* yang harus dibuat (Putra and Andriani, 2019).

Simbol yang digunakan dalam *sequence diagram* dapat dilihat pada tabel 2.2






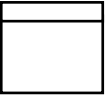
Tabel 2.2 Sequence Diagram

No	Simbol	Nama	Keterangan
1		Aktor	Mewakili peran orang, sistem, atau alat dalam berinteraksi di dalam dandi luar
2		<i>Activation Box</i>	Merepresentasikan waktu yang dibutuhkan suatu objek untuk menyelesaikan tugas tertentu
3		Objek	Mendemonstrasikan bagaimana sebuah objek akan berperilaku dalam sistem
4		<i>Lifelines</i>	Menunjukkan kejadian berurutan yang terjadi pada sebuah objek
5		<i>Synchronous Message</i>	Menunjukkan bahwa pengirim harus menunggu respon pesan sebelum melanjutkan aktivitas

2.7.3. Activity Diagram

Activity diagram adalah jenis diagram yang digunakan untuk menggambarkan alur kerja atau aktivitas sistem yang ada dalam perangkat lunak. (Putra and Andriani, 2019). Adapun beberapa simbol yang terdapat pada *activity diagram* dapat dilihat pada 2.3

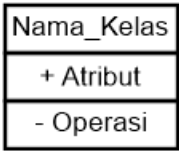


Tabel 2.3 *Activity Diagram*


No	Simbol	Nama	Keterangan
1		Aktivitas	Aktivitas yang dilakukan system, aktivitas biasanya diawali dengan kata kerja
2		Percabangan	Percabangan dimana ada pilihan aktivitas yang lebih dari satu
3		Status Awal	Sebuah diagram memiliki sebuah status awal
4		Status Akhir	Bagaimana benda dibuat dan dihancurkan
5		Penggabungan	Penggabungan dimana lebih dari satu aktivitas digabungkan jadi satu
6		Swimlane	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi

2.7.4. Class Diagram

Class diagram merupakan sebuah spesifikasi yang jika diinstansikan akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut. *Class diagram* memiliki tiga area pokok yaitu nama, atribut dan metode. Berikut merupakan simbol-simbol yang terdapat pada *class diagram* dapat dilihat pada gambar 2.4

Tabel 2.4 Class Diagram

No	Simbol	Nama	Keterangan
1		Kelas	Kelas yang terdapat pada struktur
2		Interface	Interface sama dengan konsep interface dalam pemrograman berorientasi objek
3		Asosiasi Berarah	Relasi antar kelas dengan makna kelas yang satu digunakan oleh kelas yang lain, asosiasi biasanya disertai dengan simbol
4		Generalisasi	Relasi antar kelas dengan makna generalisasi spesialisasi
5		Dependency	Relasi antar kelas dengan makna ketergantungan antar kelas
6		Agregasi	Relasi antar kelas dengan makna ketergantungan antar kelas

7		Asosiasi	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity

Tabel 2.4 Class Diagram (lanjutan)

2.8. Laravel

Laravel adalah suatu kerangka kerja (*framework*) yang dibangun di atas bahasa pemrograman PHP dengan sumber terbuka (open source) dan menggunakan konsep Model-View-Controller (MVC). Laravel didistribusikan di bawah lisensi MIT License dan menggunakan platform Github sebagai tempat untuk berbagi kode sumber dan menjalankannya. *Framework* ini awalnya dikembangkan oleh Taylor Otwell untuk menyediakan alternatif yang lebih baik dari framework PHP lainnya seperti Codeigniter. Studi pustaka dilakukan untuk memahami lebih dalam tentang Laravel, kemudian dilakukan analisis penggunaan framework dan observasi untuk mengimplementasikan program menggunakan Laravel. Setelah implementasi program selesai, maka tahap penelitian juga selesai (Arta, Sudana and Putri, 2022).

2.9. Bootstrap

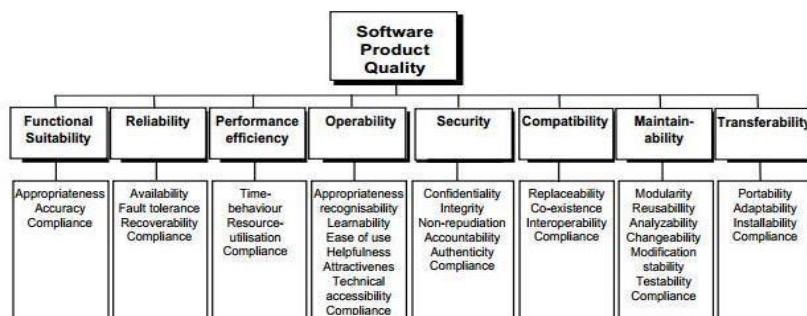
Bootstrap merupakan salah satu jenis *framework* Framework adalah sebuah perangkat lunak atau aplikasi yang berfungsi sebagai kerangka kerja yang dapat mempermudah pengembang dalam mengembangkan aplikasi website. Bootstrap merupakan salah satu jenis framework yang menggabungkan CSS dan JavaScript. Framework ini dikembangkan oleh Mark Otto dan Jacob Thornton di kantor Twitter, dengan tujuan untuk memberikan konsistensi dalam tahapan pengembangan antarmuka pada pembuatan website. Bootstrap ditawarkan sebagai alternatif untuk berbagai jenis framework lainnya (Khana Wijaya, Rishi Suprianto and Endi Istiawan, 2022).

2.10. MySQL

MySQL adalah sistem manajemen basis data SQL atau DBMS yang bersifat multi-threaded dan multi-user. Karena ukurannya yang kecil dan kecepatannya yang tinggi, MySQL sangat populer di kalangan pengembang web. Selain itu, MySQL juga merupakan perangkat lunak sumber terbuka (open source) yang artinya dapat digunakan secara gratis. Faktor ini juga menjadi alasan mengapa MySQL sangat ideal untuk digunakan pada website.

2.11. ISO 25010

ISO 9126 dan ISO 25010 adalah dua standar Internasional yang digunakan untuk pengujian perangkat lunak (Lamada, Miru and Amalia, 2020) ISO 25010 dikembangkan sebagai alternatif dari standar ISO 9126 dengan mempertimbangkan perkembangan teknologi informasi dan komunikasi (ICT). Standar ISO 25010 terdiri dari delapan karakteristik utama, yaitu kompatibilitas fitur, keandalan, efisiensi, ketersediaan, keamanan, kompatibilitas, pemeliharaan, dan probabilitas. Dua standar ini digunakan sebagai acuan dalam melakukan pengujian terhadap perangkat lunak untuk memastikan kualitas dan kesesuaian fungsionalitasnya.



Gambar 2.2 Karakteristik ISO 25010

Berikut adalah penjelasan dari delapan karakteristik ISO 25010 yang terlihat pada gambar di atas:

1. **Functionality Suitability:** Pengujian ini menggunakan test case dengan Guttman scale, yang memberikan jawaban unik untuk pertanyaan yang diujikan. Tujuan dari pengujian ini adalah untuk memastikan kesesuaian fungsionalitas aplikasi.
2. **Performance Efficiency:** Pengujian ini dilakukan untuk menguji kinerja aplikasi yang telah dikembangkan, memastikan bahwa aplikasi berjalan dengan efisien dan responsif.
3. **Compatibility:** Pengujian ini menguji kemampuan suatu komponen atau sistem untuk bertukar informasi dengan baik dengan sistem lainnya.
4. **Usability:** Pengujian ini dilakukan untuk menganalisis umpan balik dari pengguna menggunakan skala dengan 5 pilihan, memastikan bahwa aplikasi mudah digunakan dan sesuai dengan kebutuhan pengguna.
5. **Reliability:** Pengujian ini bertujuan untuk menguji keandalan atau keterpercayaan sistem, memastikan bahwa aplikasi berfungsi secara konsisten dan dapat diandalkan.
6. **Security:** Pengujian ini dilakukan untuk menguji sejauh mana sistem atau produk yang menyediakan layanan melindungi dari akses, penggunaan, modifikasi, gangguan, atau pengungkapan bahaya yang tidak diinginkan.
7. **Maintainability:** Pengujian ini melibatkan alat ukur yang telah diuji oleh peneliti langsung di lapangan kegiatan, dengan tiga aspek utama, yaitu alat ukur, perhitungan konsistensi, dan kesederhanaan dalam pemeliharaan aplikasi.
8. **Portability:** Pengujian ini bertujuan untuk menguji sejauh mana suatu sistem atau produk dapat dipindahkan dari satu ruangan ke ruangan lainnya tanpa mengalami masalah.

Semua karakteristik ini penting untuk memastikan bahwa aplikasi memiliki kualitas yang baik, berfungsi dengan baik, dan memenuhi kebutuhan pengguna dengan tepat. Pengujian berdasarkan ISO 25010 membantu dalam mengidentifikasi dan mengatasi potensi masalah yang mungkin muncul dalam pengembangan aplikasi.

2.12. Payment Gateway

payment gateway merupakan sebuah layanan elektronik yang digunakan dalam melakukan proses transaksi pembayaran dengan menggunakan alat pembayaran dengan media kartu, uang elektronik, dan/atau Proprietary Channel. Payment gateway merupakan sistem yang membantu efisiensi proses pembayaran pembeli kepada penjual. Sistem pembayaran seperti ini sering dipakai dalam e-commerce atau toko online yang terpercaya.

2.13. Midtrans

Midtrans adalah salah satu *gateway* pembayaran yang menyediakan layanan bagi para pelaku bisnis online dengan berbagai metode seperti pembayaran kartu, transfer bank, debit langsung, e-wallet, over the counter, dan lain sebagainya. Selain itu, Midtrans juga menyediakan fitur sistem analisis data untuk mengelola risiko serta sistem chat untuk memfasilitasi komunikasi antara pelanggan dan penjual online dalam meningkatkan volume transaksi.