

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Pada penelitian ini, penulis melakukan tinjauan pustaka pada penelitian sebelumnya dan serupa. Sebagai pendukung penelitian yang dilakukan oleh penulis, dibawah ini merupakan tinjauan pustaka yang sudah di lakukan oleh peneliti sebelumnya.

Tabel 2.1 Tinjauan Pustaka

No	Penulis, Tahun	Judul	Metode	Hasil
1	Sukma Fitria Putri dan Widi Fathiyah Rahim, 2022	Perancangan Dan Implementasi Sistem Informasi Penjualan Konsinyasi Berbasis Web Pada Umkm Permata Kurma	SDLC	Memudahkan entri dan pengolahan data penjualan konsinyasi serta laporan yang diperlukan.
2	Herlina Widiastuti dan Azizah Fatmawati, 2019	Sistem Informasi Produksi Usaha Mikro Kecil Menengah Pada Zahroh Barokah	Waterfall	Mempermudah pemilik dalam pencatatan produksi dan transaksi
3	Farizan Luthfi, 2017	Penggunaan Framework Laravel	Prototype	Berhasil mengembangkan

No	Penulis, Tahun	Judul	Metode	Hasil
		Dalam Rancang Bangun Modul Back-End Artikel Website Bisnisbisnis.ID		modul Back-end untuk pembuatan artikel dan berhasil mengintegrasikannya dengan frontend.
4	Mardi Yudhi Putra dan Rayhan Wahyudin Ratu Lolly, 2021	Sistem Aplikasi Penjualan Souvenir Berbasis Web Menggunakan Metode Rapid Application Development (RAD)	RAD	memudahkan pencarian dan pencatatan data, menyimpan data pemesan dan produk, menampilkan laporan penjualan, dan memperluas jangkauan pemasaran.
5	Puteri Noviandhiny, 2018	Rancang Bangun Aplikasi Penjualan dan Pembelian Berbasis Web pada Apotek Neofarma Sanggau	Waterfall	Mempermudah penjualan dan pembelian obat. Aplikasi berfungsi dengan baik dalam transaksi, mencetak laporan, serta

No	Penulis, Tahun	Judul	Metode	Hasil
				memberikan pemberitahuan tentang obat

2.1.1 Tinjauan Terhadap Literatur 1

Oleh (Fitria Putri & Rahim, 2022) membahas perancangan dan implementasi sistem informasi penjualan konsinyasi berbasis web pada UMKM Permata Kurma. Penelitian ini menggunakan metode deskriptif dengan pengumpulan data melalui pengamatan, wawancara, dan kepustakaan. Dengan menggunakan bahasa pemrograman web dan database MySQL, sistem informasi ini dikembangkan berdasarkan pendekatan SDLC (System Development Life Cycle) dengan pengujian black box. Sistem ini mengatasi masalah sistem sebelumnya yang dilakukan secara manual dengan menyediakan entri data penjualan, pengolahan data, serta laporan yang diperlukan oleh UMKM. Fitur pengendalian hak akses pada menu login juga memastikan keamanan data dengan menampilkan menu utama yang sesuai dengan jabatan pengguna. Hasil penelitian menunjukkan bahwa sistem yang dikembangkan memenuhi kebutuhan UMKM Permata Kurma dalam hal entri dan pengolahan data penjualan konsinyasi, serta menyajikan laporan yang diperlukan.

2.1.2 Tinjauan Terhadap Literatur II

Oleh (Widiastuti & Fatmawati, 2019) mengembangkan sistem informasi produksi UMKM di Zahroh Barokah yang bergerak dalam makanan ringan.

Kendala dalam pendataan produksi dan pelaporan manual menggunakan media kertas yang rentan hilang atau rusak diidentifikasi. Metode pengembangan sistem yang digunakan adalah pendekatan waterfall dengan analisis kebutuhan, desain, pemrograman, pengujian, dan implementasi. Hasilnya adalah kemampuan mengelola data pemasukan dan pengeluaran produksi dagang dan memudahkan pencetakan laporan. Sistem ini diuji menggunakan metode black box dan terbukti berjalan dengan baik. Pengujian dengan System Usability Scale (SUS) mendapatkan rata-rata nilai 74, menunjukkan kualitas sistem yang baik dan diterima oleh pengguna.

2.1.3 Tinjauan Terhadap Literatur III

Oleh (Luthfi, 2017) dibahas penggunaan Framework Laravel dalam rancang bangun modul Back-End artikel pada website Bisnisbisnis.ID, sebuah direktori bisnis yang memuat daftar bisnis dalam kategori tertentu. Penelitian ini fokus pada pembuatan modul Back-End Artikel dengan menggunakan Framework Laravel dan MySQL sebagai database. Hasil penelitian menunjukkan bahwa modul artikel berhasil dibuat, dan melalui pengujian menggunakan metode Black-Box testing, diperoleh kesimpulan bahwa fungsi yang dibuat telah sesuai dengan persyaratan sistem. Dengan adanya penggunaan Framework Laravel, website Bisnisbisnis.ID dapat menghadirkan modul Back-End Artikel yang sesuai dengan kebutuhan pengguna.

2.1.4 Tinjauan Terhadap Literatur IV

Oleh (Putra & RWR, 2021) penelitian ini membahas penggunaan metode Rapid Application Development (RAD) dalam pengembangan sistem aplikasi penjualan souvenir berbasis web. Fokus penelitian ini adalah Maju Jaya Shop, sebuah toko

penjualan souvenir yang menghadapi kendala dalam pencarian dan pencatatan data penjualan yang masih dilakukan secara manual, serta terbatasnya jangkauan pemasaran souvenir. Dalam pengembangan aplikasi, metode RAD digunakan dengan database MySQL serta menggunakan bahasa pemrograman PHP dan diuji menggunakan Black-Box Testing. Hasil penelitian ini menghasilkan sistem aplikasi penjualan souvenir berbasis web yang membantu dalam pencarian dan pencatatan data penjualan, serta menyediakan laporan transaksi yang dapat diakses melalui jaringan internet. Dengan adanya aplikasi ini, jangkauan pemasaran souvenir dapat diperluas, meningkatkan omset penjualan di Maju Jaya Shop.

2.1.5 Tinjauan Terhadap Literatur V

Oleh (Noviandhiny, 2018) membahas rancangan dan implementasi Aplikasi Penjualan dan Pembelian Berbasis Web pada Apotek Neofarma Sanggau yang mengalami kendala dalam pengolahan data obat secara manual, termasuk pencatatan stok obat kadaluarsa dan laporan penjualan. Dalam penelitian ini, dirancang aplikasi penjualan dan pembelian obat berbasis web menggunakan metode pengembangan waterfall dan pengujian Black-Box Testing. Tujuannya adalah menciptakan aplikasi yang mampu memfasilitasi penjualan dan pembelian obat secara terkomputerisasi, memberikan notifikasi obat kadaluarsa atau habis, serta memudahkan pengelolaan data obat. Hasil pengujian menunjukkan bahwa semua kelas uji berjalan dengan baik, dan pengujian kuesioner penggunaan aplikasi memperoleh nilai persentase Likert sebesar 81,5.

2.2 Keaslian Penelitian

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

1. Menerapkan desain responsif untuk berdasarkan dari penelitian sebelumnya yang terkendala dengan berbagai komponen di dalamnya.
2. Terdapat grafik batang *vertikal* penjualan yang menyajikan data visual tentang tren dan perubahan dalam penjualan produk.

2.3 Landasan Teori

Berdasarkan penelitian yang akan dilakukan, maka penulis menyusun sebuah landasan teori yang berkaitan dengan penelitian ini, antara lain:

2.3.1 Konsep Dasar Web

Menurut Sastradipraja dan Munawar (2022) Web adalah cara mengakses informasi melalui media internet. Ini adalah model berbagi informasi yang dibangun di atas Internet. Web menggunakan protokol HTTP, hanya salah satu bahasa yang digunakan melalui internet, untuk mengirimkan data. Web juga menggunakan browser, seperti Internet Explorer atau Firefox, untuk mengakses dokumen web yang disebut halaman web yang terhubung satu sama lain melalui *hyperlink*. Dokumen web juga berisi grafik, suara, teks dan video.

Berdasarkan dari teori tersebut, penulis menarik kesimpulan web adalah sebuah model untuk mengakses dan berbagi informasi melalui internet. Dokumen-dokumen web terhubung melalui *hyperlink* dan disebut halaman web. Pengguna dapat menjelajahi halaman web yang saling terhubung untuk mengakses teks,

gambar, animasi, dan video. Untuk mengakses web, digunakan perangkat lunak yaitu browser.

2.3.2 Pengertian Website

Menurut Kusumawardani et al., (2023) Website merupakan kumpulan halaman digital yang berisi informasi berupa teks, animasi, gambar, suara dan video atau gabungan dari semuanya yang terkoneksi melalui internet, sehingga dapat diakses oleh seluruh ataupun siapapun orang yang bisa terkoneksi jaringan internet. Terdapat dua jenis website berdasarkan sifatnya, yaitu:

1. Website Statis

Merupakan jenis website yang umumnya memiliki halaman-halaman yang tidak berubah. Artinya, kontennya tetap dan tidak berubah dari waktu ke waktu. Setiap perubahan atau penambahan informasi pada halaman-halaman tersebut harus dilakukan secara manual dengan mengubah kode yang menjadi struktur dari website tersebut. Pada website statis, pengguna biasanya hanya dapat melihat informasi yang telah ditentukan sebelumnya tanpa ada interaksi dinamis.

2. Website Dinamis

Merupakan jenis website yang didesain secara terstruktur untuk sering diperbarui atau mengizinkan perubahan konten secara berulang tanpa perlu mengakses kode dari struktur pembuatannya.

2.3.3 Penjualan *Konsinyasi*

Sistem penjualan *konsinyasi* adalah suatu cara transaksi di mana pemilik barang menyerahkan barangnya kepada pihak lain, yang bertindak sebagai agen penjualan. Dalam perjanjian *konsinyasi*, pihak kedua (agen penjualan) akan menjual barang tersebut kepada konsumen atas nama pemilik barang. Pemilik

barang akan memberikan komisi kepada pihak kedua atas barang yang berhasil terjual. Dalam hal ini, pemilik barang disebut sebagai *consignor* (konsinyor), sementara pihak yang menerima barang titipan disebut sebagai *consignee* (*konsinyi*) atau agen penjualan (Mesra et al., 2021).

Dalam sistem penjualan *konsinyasi*, agen penjual memiliki peran penting sebagai perantara antara pemilik barang dan konsumen. Salah satu aspek yang menjadi daya tarik bagi agen penjual adalah adanya komisi yang diterimanya dari pemilik barang atas setiap barang yang berhasil terjual.

2.3.4 *HyperText Markup Language* (HTML)

Pengertian HTML Menurut Kusumawardani et al., (2023) merupakan sebuah bahasa pemrograman standar yang berfungsi untuk membuat halaman *website* agar dapat diakses dan menampilkan berbagai jenis konten lewat perantara *browser* internet. Sedangkan menurut Sastradipraja dan Munawar (2022) HTML merupakan bahasa markup bukan bahasa pemrograman, istilah HTML adalah singkatan dari *Hypertext Markup Language*. Kami dapat menerapkan bahasa markup ini ke halaman anda untuk menampilkan teks, gambar, file suara dan film, dan hampir semua jenis informasi elektronik lainnya.

Berdasarkan kedua pendapat tersebut, dapat disimpulkan bahwa HTML merupakan sebuah standar bahasa *markup* yang berfungsi untuk membuat halaman *website*. Dengan menggunakan HTML, pengguna dapat menciptakan halaman *website* yang dapat diakses melalui *browser* internet dan menampilkan berbagai jenis konten seperti teks, gambar, file suara, dan film.

2.3.5 Cascading Style Sheet (CSS)

Menurut Maryam dan Supriyanti, (2023) pengertian CSS adalah bahasa yang dapat digunakan untuk mendefinisikan bagaimana suatu bahasa *markup* ditampilkan pada suatu media, salah satunya adalah HTML, atau dengan kata lain bahwa CSS merupakan kumpulan kode yang dapat digunakan untuk mendesain halaman website agar lebih menarik saat dilihat atau dikunjungi.

2.3.6 Javascript

Pengertian *Javascript* menurut Siahaan dan Sianipar (2018) merupakan bahasa skript populer yang dipakai untuk menciptakan halaman web yang dapat berinteraksi dengan pengguna dan dapat merespons *event* yang terjadi pada halaman. Javascript merupakan perekat yang menyatukan halaman-halaman *web*. Akan sangat susah menjumpai halaman *web* komersial yang tidak memuat kode *Javascript*. Sedangkan menurut Kusumawardani et al., (2023) JavaScript merupakan bahasa pemrograman murni yang biasanya dimanfaatkan untuk mendesain halaman website supaya terlihat lebih interaktif hingga atau animasi.

Berdasarkan kedua pendapat tersebut, dapat disimpulkan bahwa JavaScript adalah bahasa pemrograman yang digunakan untuk menciptakan halaman web yang interaktif, merespons peristiwa, dan meningkatkan tampilan halaman website dengan elemen-elemen interaktif dan animasi.

2.3.7 Hypertext Preprocessor (PHP)

PHP menurut Maryam dan Supriyanti, (2023) adalah sebuah skrip berbasis *server (server-side)* yang mampu menerjemahkan kode php menjadi tampilan website yang dinamis di sisi *client (browser)*. Kode PHP tidak diberikan langsung ketika ada permintaan dari client, melainkan dimasukkan ke dalam kode HTML.

Sedangkan menurut Mardani et al., (2021) PHP merupakan sebuah bahasa *server-side scripting* yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Karena PHP merupakan *server-side scripting* maka sintaks dan perintah-perintah PHP akan dieksekusi di server kemudian hasilnya dikirimkan ke browser dalam format HTML. Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh user sehingga keamanan halaman web lebih terjamin.

Berdasarkan kedua pendapat tersebut, dapat disimpulkan bahwa PHP adalah bahasa skrip *server-side* yang digunakan untuk membuat halaman web dinamis. PHP menerjemahkan kode PHP menjadi tampilan website dinamis di sisi *client* (*browser*). Kode PHP dimasukkan ke dalam kode HTML dan dieksekusi di *server*. Hasilnya dikirimkan ke browser dalam format HTML, sehingga kode program PHP tidak terlihat oleh pengguna, meningkatkan keamanan halaman web.

Salah satu kelebihan utama PHP adalah mendukung berbagai jenis database populer seperti MySQL, PostgreSQL, dan Oracle, sehingga memungkinkan pengembang untuk melakukan manipulasi data. Dengan kepopulerannya dan dukungan yang luas dari komunitas pengembang, PHP telah menjadi salah satu bahasa pemrograman yang sangat populer dalam pengembangan aplikasi web.

2.3.8 Framework

Menurut Habibi et al., (2020) *Framework* adalah sekumpulan fungsi yang memudahkan *programmer* atau pengembang aplikasi web. Fungsi-fungsi yang berada di dalam *framework* seperti *plugin* dan konsep. Konsep yang dimaksud bertujuan untuk membangun sistem supaya dapat dikembangkan dengan rapi dan terstruktur. Selain menyediakan fungsi, *framework* juga menyediakan *library* dan *tools* lain yang anda butuhkan ketika mengembangkan aplikasi web.

Sedangkan Menurut Sholihin et al., (2022) *Framework* merupakan kerangka kerja yang berisi kumpulan kode-kode program yang dikumpulkan dan disusun secara rapi pada folder yang terstruktur dengan baik agar mudah digunakan. Laravel fokus di bagian *end-user*, yang berarti fokus pada kejelasan dan kesederhanaan, baik penulisan maupun tampilan, serta menghasilkan fungsionalitas aplikasi web yang bekerja sebagaimana mestinya.

Berdasarkan dua pendapat tersebut, dapat disimpulkan bahwa *framework* adalah sekumpulan fungsi, *plugin*, dan konsep yang memudahkan pengembang aplikasi web dalam membangun sistem secara rapi dan terstruktur. *Framework* juga menyediakan *library* dan *tools* lain yang dibutuhkan dalam pengembangan aplikasi web. Fokus dalam pengembangan *framework* adalah pada kejelasan, kesederhanaan, dan fungsionalitas aplikasi web yang bekerja dengan baik.

2.3.9 Laravel

Menurut Habibi et al., (2020) Laravel adalah PHP *open-source framework* yang dibangun dengan *model view controller* dan dibekali dengan berbagai macam sintaks. Sejak diluncurkan tahun 2011, membuat website dengan Laravel banyak digemari oleh berbagai komunitas programmer Github, sebelum kemudian menyebar ke seluruh dunia. Framework ini menyediakan beberapa jenis PHP library dan beberapa fungsi lain yang bisa memudahkan anda menuliskan baris kode. Framework Laravel dibuat dengan tujuan memperindah cara untuk membuat website. Laravel terkenal sederhana dan elegan karena pembuatannya memang ditujukan untuk *end-user*. Sedangkan menurut Sholihin et al., (2022) Laravel dikembangkan oleh programmer asal amerika yang bernama Taylor Otwell dan diluncurkan sejak tahun 2011, kemudian mengalami pertumbuhan yang cukup

eksponensial. Di tahun 2015, Laravel adalah framework yang paling banyak mendapatkan bintang di Github. Sekarang framework ini menjadi salah satu yang populer di dunia, tidak terkecuali di Indonesia.

2.3.10 MySQL

Menurut Rusli et al., (2019) MySQL dibaca mai-ess-que-ell bukan mai-sequel adalah suatu sistem manajemen database (Database Management System) atau DBMS, jadi MySQL adalah sistem yang berguna untuk melakukan proses pengaturan koleksi-koleksi struktur data (database) baik yang meliputi proses pembuatan atau proses pengelolaan database.

MySQL digunakan untuk mengelola dan membuat database sistem pada server. Database ini berisi berbagai informasi dan dikelola menggunakan perintah-perintah dasar SQL. MySQL juga memungkinkan pengguna untuk dengan mudah mengakses data dalam bentuk teks (*string*) baik secara personal maupun melalui sistem web yang bersifat publik.

2.3.11 Metode Personal Extreme Programming

Metodologi ialah bagian utama dalam perencanaan dan pengembangan perangkat lunak yang bertujuan untuk menghasilkan sistem informasi sesuai kebutuhan bisnis suatu organisasi (Bolung & HRK, 2017). Terdapat beberapa metodologi pengembangan sistem yang dapat digunakan, antara lain: *waterfall*, RAD, *prototype* serta *agile development*. Pendekatan *agile development* telah diperkenalkan sebagai upaya untuk membuat rekayasa perangkat lunak yang fleksibel dan efisien (Borman et al., 2020). Salah satu metode *agile development* yang cocok untuk pengembangan sistem yang cepat adalah *Extreme Programming*

(XP). Metode ini dapat mengatasi perubahan berdasarkan desain dan fitur dengan penanganan yang fleksibel (Faizal dan Nanda, 2021).

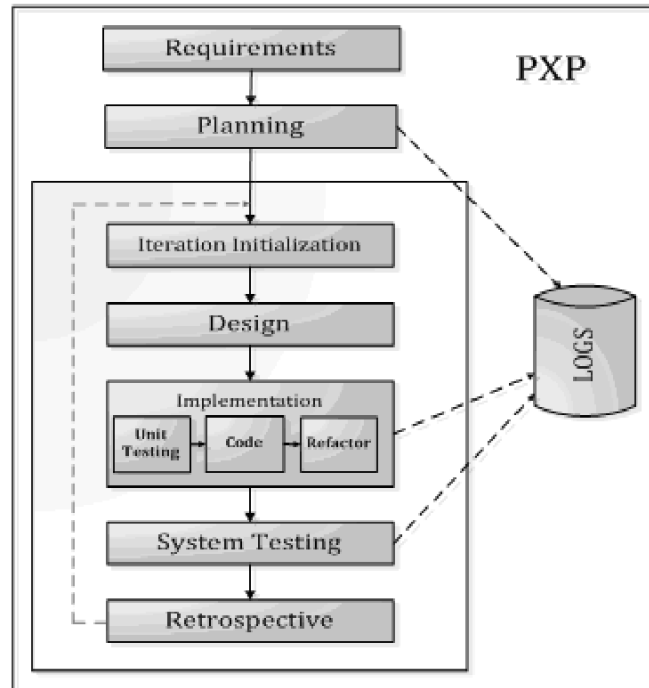
XP biasanya dikerjakan oleh sebuah tim kecil di mana pengembang bekerja berpasangan untuk mengembangkan sistem. Namun, metode ini mungkin kurang cocok jika pengembangan dilakukan oleh satu orang saja. Untuk mengatasi hal tersebut, telah dikembangkan sebuah variasi *metode* yang disebut *Personal Extreme Programming* (PXP). PXP memungkinkan seorang pengembang tunggal untuk menerapkan prinsip-prinsip XP dalam pengembangan perangkat lunak.

PXP suatu proses pembuatan software yang dirancang dan diimplementasikan oleh pengembang secara individu. PXP mempertahankan prinsip-prinsip dengan mengurangi beban kerja dokumentasi dan pemeliharaan. Proses pengembangan PXP bersifat iteratif, dan praktik implementasi memungkinkan pengembang untuk merespons perubahan dengan lebih fleksibel (Sita Eriana & Zein, 2021).

Dalam PXP, seorang pengembang akan melakukan semua tugas pengembangan sendiri, termasuk perencanaan, perancangan, pengkodean, dan pengujian. Meskipun bekerja sendiri, pengembang masih dapat menerapkan prinsip-prinsip XP seperti perencanaan adaptif, pengujian berkelanjutan, dan *refactoring* (Asri & Setiawan, 2017).

2.3.12 Tahapan Metode PXP

Adapun proses dan tahapan PXP yaitu:



Gambar 2.1 Sumber: *(Sita Eriana & Zein, 2021)*

1. Pengumpulan Kebutuhan (Requirements): Pada langkah ini, pengembang bekerja sama dengan pemilik UMKM Toko Kripik Pisang Asa Cipto Roso untuk mengumpulkan kebutuhan yang diperlukan. Pengumpulan kebutuhan ini dilakukan melalui proses wawancara dan diskusi dengan tujuan memahami kebutuhan pengguna secara mendalam. Hasil dari proses ini kemudian dituliskan dalam bentuk user story yang mengikuti format berikut: "Sebagai <jenis pengguna>, saya ingin <melakukan tindakan tertentu> untuk <mendapatkan manfaat dari tindakan tersebut>".
2. Perencanaan (Planning): Tahap pengembang akan membuat rangkaian tugas yang akan dikerjakan dalam setiap iterasi berdasarkan User Stories yang telah diperoleh. merencanakan langkah-langkah yang akan diambil

dalam pengembangan perangkat lunak, menentukan prioritas, dan mengestimasi waktu yang diperlukan untuk setiap tugas.

3. Tahap Awal Iterasi (*Iteration Initialization*): Tahap awal setiap iterasi dalam pengembangan perangkat lunak disebut *Iteration Initialization*. Pada tahap ini, dilakukan pemilihan tugas yang akan menjadi fokus utama dalam iterasi tersebut. Tugas-tugas ini dipilih berdasarkan hasil perencanaan yang dilakukan pada tahap planning.
4. Perancangan (*Design*): Pada tahap Desain, pengembang membuat model modul sistem yang akan diimplementasikan selama iterasi. Desain sistem ini bertujuan untuk memenuhi kebutuhan pengguna yang telah diperoleh selama proses kebutuhan. Desain yang dibuat oleh pengembang merupakan prototipe solusi awal yang mengikuti skema desain prototipe berdasarkan Class Responsibilities Collaboration (CRC) Cards dari iterasi yang sedang berlangsung.
5. Implementasi (*Implementation*): Dalam fase Implementasi, pengembang melakukan pembuatan kode pemrograman untuk mengimplementasikan point-point yang telah dirancang pada tahap perancangan. Fase Implementasi terdiri dari tiga tahap yang penting, yaitu *Unit Testing*, *Code Generation*, dan *Code Refactoring*.
6. Pengujian Sistem (*System Testing*): Setelah kode program selesai, tahap pengujian dilakukan untuk memverifikasi keberfungsian perangkat lunak dan apakah sesuai dengan keinginan user.
7. *Retrospective*, menganalisis jalannya setiap fase pengembangan modul, mengevaluasi kesesuaian estimasi waktu pengerjaan, dan mencari

penyebab terjadinya keterlambatan dalam proses pengembangan. Tujuannya adalah untuk mencegah hal-hal tersebut terulang kembali di iterasi selanjutnya.

2.3.13 Analisis PIECES

Menurut Soufitri, (2022) menyatakan bahwa analisis PIECES (*Performance, Information, Economy, Control, Eficiency, dan Service*) merupakan teknik untuk mengidentifikasi dan memecahkan permasalahan yang terjadi pada sistem informasi.

1. Analisis Kinerja Sistem (*Performance*)

Kinerja adalah suatu kemampuan sistem dalam menyelesaikan tugas dengan cepat sehingga sasaran dapat segera tercapai. Kinerja diukur dengan jumlah produksi (*throughput*) dan waktu yang digunakan untuk menyesuaikan perpindahan pekerjaan (*response time*). *Throughput* adalah jumlah dari pekerjaan yang dapat dilakukan.

2. Analisis Informasi (*Information*)

Informasi merupakan hal yang penting karena dengan informasi tersebut pihak manajemen (*marketing*) dan user dapat melakukan langkah selanjutnya. Apabila kemampuan sistem informasi baik, maka *user* akan mendapatkan informasi yang akurat, tepat waktu dan relevan sesuai dengan yang diharapkan. Evaluasi kemampuan sistem informasi dalam menghasilkan nilai atau produk yang bermanfaat untuk menyikapi peluang dalam menangani masalah yang muncul.

3. Analisis Ekonomi (*Economy*)

Adalah penilaian sistem atas biaya dan keuntungan yang akan didapatkan dari sistem yang diterapkan. Sistem ini akan memberikan penghematan operasional dan

keuntungan bagi instansi atau perusahaan. Hal yang diperlukan dalam analisis ini meliputi biaya dan keuntungan. Peningkatan terhadap kebutuhan ekonomis mempengaruhi pengendalian biaya dan peningkatan manfaat.

4. Analisis Pengendalian (*Control*)

Adalah sistem keamanan yang digunakan harus dapat mengamankan data dari kerusakan, misalnya dengan memback-up data. Selain itu sistem keamanan juga harus dapat mengamankan data dari akses yang tidak diizinkan. Analisis ini meliputi pengawasan dan pengendalian. Kontrol berkaitan erat dengan pengendalian dan keamanan baik dari segi *hardware*, *software*, dan *brainware*.

5. Analisis Efisiensi (*Efficiency*)

Efisiensi berhubungan dengan bagaimana sumber tersebut dapat digunakan secara optimal. Operasi pada suatu perusahaan dikatakan efisien atau tidak biasanya didasarkan pada tugas dan tanggung jawab dalam melaksanakan kegiatan.

6. Analisis Pelayanan (*Service*)

Untuk menilai kualitas dari sebuah sistem salah satunya dapat kita lihat dari pelayanannya. Pada sistem informasi perpustakaan peningkatan pelayanan terhadap anggota merupakan bagian dari tujuan umum diadakannya sistem informasi. Untuk menilai dampak sistem informasi terhadap kualitas pelayanan dari suatu sistem dapat dilihat dari kriteria-kriteria berikut ini:

1. Sistem menghasilkan produk yang tidak akurat
2. Sistem menghasilkan produk yang tidak konsisten
3. Sistem menghasilkan produk yang tidak dipercaya
4. Sistem tidak mudah dipelajari
5. Sistem tidak mudah digunakan

6. Sistem canggung untuk digunakan
7. Sistem tidak fleksibel

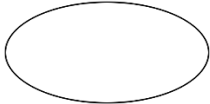

2.3.14 *Unified Modeling Language (UML)*


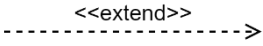
Unified Modeling Language (UML) menurut Rosa dan Shalahuddin, (2016) adalah salah standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.


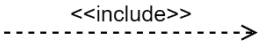
2.3.15 *Use Case Diagram*

Menurut Rosa dan Shalahuddin, (2016) Use case atau diagram use case merupakan pemodelan untuk kelakuan (behavior) sistem informasi yang akan dibuat. Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Berikut dibawah ini simbol-simbol yang digunakan pada Use Case Diagram dapat dilihat pada tabel 2.2 berikut

Tabel 2.2 Simbol-Simbol Use Case Diagram

No	Simbol	Nama	Keterangan
1		Use Case	Fungsionalitas yang disediakan sistem sebagai unit-unit atau actor, yang biasanya dinyatakan dengan kata kerja diawal frase nama use case.
2		Aktor	Orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan

No	Simbol	Nama	Keterangan
			dibuat diluar sistem informasi yang akan dibuat itu sendiri, walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu orang, yang biasanya dinyatakan menggunakan kata benda di awal frase nama actor.
3		Association	komunikasi antara aktor dan use case yang berpartisipasi pada use case atau use case memiliki interaksi dengan aktor.
4		Extend	Relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan dapat berdiri sendiri walau tanpa use case tambahan itu; mirip dengan prinsip inheritance pada PBO; biasanya use case tambahan memiliki nama depan yang sama dengan use





No	Simbol	Nama	Keterangan
			case yang ditambahkan, misal arah panah mengarah pada use case yang ditambahkan; biasanya use case yang menjadi extend-nya merupakan jenis yang sama dengan use case yang menjadi induknya.
5		Generalisation	Hubungan generalisasi dan spesialisasi (umum-khusus) antara dua buah use case dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya, misalnya: Arah panah mengarah pada use case yang menjadi generalisasinya (umum)
6		Include	Relasi use case tambahan ke sebuah use case di mana use case yang ditambahkan memerlukan use case ini untuk menjalankan fungsinya


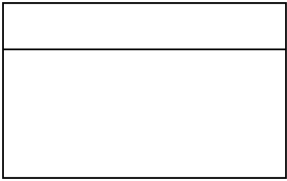
No	Simbol	Nama	Keterangan
			atau sebagai syarat dijalankan use case

2.3.16 Activity Diagram

Activity Diagram atau diagram kerja menurut Rosa dan Shalahuddin, (2016) menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Adapun beberapa simbol yang terdapat pada activity diagram dapat dilihat pada tabel 2.3 di bawah ini.

Tabel 2.3 Simbol-Simbol Activity Diagram

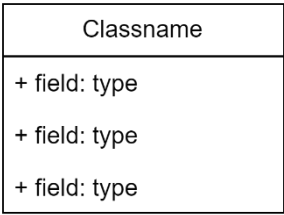





No	Simbol	Nama	Keterangan
1		Status Awal	Status awal aktivitas sistem, sebuah diagram aktivitas memiliki sebuah status awal.
2		Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
3		Percabangan	Percabangan dimana ada pilihan aktivitas yang lebih dari satu
4		Penggabungan	Asosiasi penggabungan dimana lebih dari satu

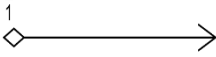
No	Simbol	Nama	Keterangan
			aktivitas digabungkan menjadi satu
5		Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
6		Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggungjawab terhadap aktivitas yang terjadi.

2.3.17 Class Diagram

Diagram Kelas atau class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem (Rosa & Shalahuddin, 2016). Berikut merupakan symbol - simbol yang terdapat pada class diagram dapat dilihat pada tabel 2.4.

Tabel 2.4 Simbol-Simbol Class Diagram

No	Simbol	Nama	Keterangan
1		Kelas	Kelas yang terdapat pada struktur sistem
2		Interface	Interface sama dengan konsep interface dalam pemrogramman berorientasi objek
3		Asosiasi	Relasi antar kelas dengan makna umum, asosiasi biasanya juga disertai dengan multiplicity
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.

No	Simbol	Nama	Keterangan
6		Agregasi	Relasi antar kelas dengan makna semua bagian.

2.3.18 User Acceptance Test

User Acceptance Testing (UAT) merupakan proses verifikasi bahwa solusi yang dibuat dalam sistem sudah sesuai untuk pengguna (Suprpto, 2021). UAT dilaksanakan pada akhir proses pengujian saat sistem siap digunakan. Tujuan utamanya adalah untuk mengembangkan perangkat lunak yang mampu memenuhi kebutuhan pengguna. Bukan hanya sekedar memenuhi spesifikasi sistem dan dapat digunakan saja, tetapi juga untuk memvalidasi apakah sistem dapat diterima atau tidak (Hady et al., 2020).

2.3.19 Skala Likert

Metode Skala Likert merupakan metode yang digunakan untuk mengukur tingkat kepuasan pengguna (Setyawan & Atapukan, 2018). Skala Likert biasanya mengenai persetujuan seseorang terhadap sebuah pernyataan, mulai dari “Sangat Setuju”, sampai kepada “Sangat Tidak Setuju”, dengan variasi di antara kedua jawaban ekstrim tersebut misalnya berupa pilihan “Setuju”, “Netral”, dan “Tidak Setuju”. Namun, dalam artikel aslinya, Likert menyiapkan beberapa jenis skala. Bukan hanya dalam bentuk seperti yang disampaikan di atas, namun juga bentuk skala mulai dari yang berbentuk pilihan “Ya” dan “Tidak”, serta pilihan ganda. Terhadap pilihan-pilihan respon tersebut selanjutnya diberikan angka, misalnya 1, 2, 3, 4 dan 5, jika pilihan responnya berjumlah 5 mulai dari Sangat Setuju, Setuju, Netral, Tidak Setuju hingga Sangat Tidak Setuju” (Suasapha, 2020). Berikut

menggambarkan skor pada Skala Likert yang digunakan untuk menilai tingkat persetujuan terhadap pernyataan yang diberikan dapat dilihat pada tabel 2.5.

Tabel 2.5 Score Skala Likert

Keterangan	Score
Sangat Setuju/Selalu/Sangat Positif Diberi Skor	5
Setuju/Sering/Positif Diberi Skor	4
Ragu-Ragu/Kadang-Kadang/Netral Diberi Skor	3
Tidak Setuju/Hampir Tidak Pernah/Negatif Diberi Skor	2
Sangat Tidak Setuju/Tidak Pernah Diberi Skor	1

Sumber: (Suasapha, 2020).

2.3.20 Black Box Testing

Black box testing atau dapat disebut juga *Behavioral Testing* adalah pengujian yang dilakukan untuk mengamati hasil input dan output dari perangkat lunak tanpa mengetahui struktur kode dari perangkat lunak. Pengujian ini dilakukan di akhir pembuatan perangkat lunak untuk mengetahui apakah perangkat lunak dapat berfungsi dengan baik (Yogie et al., 2021).