

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Tinjauan literatur membahas penelitian sebelumnya atau yang dilakukan sebelumnya untuk membandingkannya dengan penelitian yang sedang berlangsung dan penelitian yang akan datang. Tinjauan literatur berfungsi sebagai bahan informatif agar peneliti tidak plagiat saat melakukan penelitian ini. Selain itu, tinjauan literatur diperlukan untuk menemukan perbedaan isi pada setiap karya ilmiah. Beberapa penelitian terdahulu berkaitan dengan penelitian ini. Penelitian sebelumnya terdiri dari judul, perbandingan (*comparing*), kontras (*contrasing*), kritik (*criticize*), sintesis (*synthesize*), dan ringkasan (*summary*).

Adapun penelitian-penelitian tersebut, antara lain:

Paper utama diambil dari jurnal dengan judul “*Design and Build a Quality Assurance Document Archiving Application Using the Rapid Application Development*” yang terbit pada tahun 2023 oleh Abdullah. Jurnal ini membahas tentang pembangunan aplikasi pengarsipan dokumen jaminan kualitas di sebuah perusahaan manufaktur menggunakan metode RAD. Aplikasi ini memiliki fitur-fitur utama, yaitu tampilan yang user-friendly, fitur pencarian dan akses dokumen yang mudah, serta fitur keamanan penyimpanan dokumen. Aplikasi ini telah berhasil memenuhi kebutuhan pengguna.

Jurnal dengan judul “*Design And Build Company Profile Website Bagaskara Art Store Wonosobo*” yang disusun oleh Alfa Hardiyansyah dan Bitu Parga Zen pada tahun 2023. *Website* yang ada didalam jurnal dibangun menggunakan *framework* Laravel dan Bootstrap, serta *database* MySQL. Fitur-fiturnya meliputi informasi lengkap tentang Bagaskara Art Store, layanan, dan portofolio, serta formulir kontak untuk memudahkan komunikasi dengan pelanggan. *Website* dioptimalkan untuk SEO dan *mobile*, menghasilkan peningkatan visibilitas *online*, kredibilitas, dan penjualan bagi Bagaskara Art Store.

Jurnal “*Designing A Web-Based Career System Using The Laravel Framework*“ yang dibuat oleh Abdul Khaliq, Supina Batubara, Maya Syaula, dan Sahrial pada tahun 2023. Memaparkan desain dan pembangunan sistem karir *online* yang membantu pencari kerja dan perusahaan. Sistem dibangun dengan *Laravel framework*, PHP, dan *MySQL database*. Fitur-fitur yang ada meliputi pembuatan profil, pencarian kerja, pelamaran, *posting* lowongan, penyaringan pelamar, dan pengelolaan rekrutmen. Sistem terintegrasi dengan media sosial, menyediakan notifikasi, dan dasbor analitik. Hasilnya, sistem ini meningkatkan efisiensi, mempercepat rekrutmen, dan meningkatkan peluang kerja.

Jurnal “Implementasi *Framework* Laravel Pada Rancang Bangun *Website* IAKN Palangka Raya Dengan Metode *Prototype*” oleh Agri Apriliando pada tahun 2021. Pembangunan *website* IAKN Palangka Raya dengan *framework* Laravel dan metode *prototype*. *Website* dirancang untuk meningkatkan layanan informasi dan komunikasi IAKN, dengan fitur seperti tampilan modern dan responsif, informasi lengkap tentang IAKN, sistem berita dan pengumuman, portal akademik, layanan *online*, dan galeri foto. Pembangunannya meliputi analisis kebutuhan, perancangan, pembangunan dengan Laravel, dan pengujian. Hasilnya, *website* yang dibuat dapat meningkatkan aksesibilitas informasi, mempermudah komunikasi publik, dan meningkatkan citra IAKN.

Jurnal “Rancang Bangun Sistem Informasi Desa Berbasis Web Menggunakan Metode *Rapid Application Development*(RAD) Pada Desa Sukatani, Kecamatan Cikande, Kabupaten Serang – Banten” oleh Puspa Arundini, Rudhy Ho Purabaya, dan Ati Zaidiah tahun 2021. Perancangan dan pembangunan sistem informasi desa Sukatani dengan metode RAD. Sistem bertujuan untuk meningkatkan transparansi dan akses informasi bagi masyarakat desa. Dilakukan analisis kebutuhan, perancangan, dan pembangunan sistem dengan PHP dan MySQL menggunakan RAD. Hasilnya, sistem memiliki fitur profil desa, pemerintahan, lembaga, data kependudukan, potensi desa, dan informasi kegiatan. Sistem ini meningkatkan transparansi, mempermudah pelayanan publik, dan membantu promosi potensi desa.

Jurnal yang berjudul "Implementasi Metode Waterfall Dalam Pembangunan Company Profile Website Akademi Komunitas Dharma Bhakti Bangka (Ak Dbb)" tahun 2020 disusun oleh Ahmat Josi dan Sidhiq Andriyanto. Pembangunan *website* profil AKDBB dengan metode *Waterfall* bertujuan untuk mempublikasikan informasi dan meningkatkan citra AKDBB. Analisis kebutuhan, perancangan, dan pembangunan *website* dengan bahasa pemrograman dan *database* yang sesuai mengikuti tahapan-tahapan *Waterfall*. Hasilnya, *website* memiliki fitur profil institusi, program studi, prestasi, berita, dan informasi kontak. *Website* ini meningkatkan akses informasi bagi publik, mempermudah promosi program studi, dan meningkatkan citra AKDBB.

Jurnal "Perancangan Sistem Pendaftaran Berbasis Web Menggunakan Metode RAD Terhadap Peserta Training Industri Kopi Dan Aneka Minuman Pengelola UMKM (Penyandang Disabilitas)" oleh Aprian Yudistyra, Tatu Mariam Sari, Maulana Alfarisi, dan Roeslan Djatalov pada tahun 2023. Perancangan sistem pendaftaran berbasis web untuk *training* industri kopi dan aneka minuman bagi pengelola UMKM penyandang disabilitas. Sistem dirancang dengan metode RAD untuk mempermudah proses pendaftaran dan seleksi peserta. Dilakukan analisis kebutuhan, perancangan, dan pembangunan sistem dengan bahasa pemrograman dan *database* yang sesuai mengikuti prinsip-prinsip RAD. Hasilnya, sistem memiliki fitur registrasi akun, pengisian formulir pendaftaran, *upload* dokumen, dan pengumuman seleksi. Sistem ini membantu penyelenggara training dalam mengelola data peserta dan meningkatkan aksesibilitas bagi penyandang disabilitas.

Jurnal yang dibuat oleh Deri Susanti dan Elmiyati pada tahun 2020 dengan judul "Perancangan Website Media Informasi dan Pemesanan pada PT.Trita Musi Prasada dengan Metode RAD". Perancangan *website* media informasi dan pemesanan produk bagi PT.Trita Musi Prasada, perusahaan air minum dalam kemasan, menggunakan metode RAD. *Website* memiliki fitur profil perusahaan, katalog produk, formulir pemesanan, sistem pembayaran *online*, *tracking* pemesanan, dan artikel tentang AMDK. *Website* ini membantu

perusahaan dalam meningkatkan jangkauan pasar, mempermudah proses pemesanan, meningkatkan efisiensi, dan meningkatkan citra perusahaan.

Jurnal dengan judul “Sistem Informasi Pendaftaran PMI Berbasis *Website* (Studi Kasus Pada PT.Hendrarta Argaraya Kota Malang)” oleh Ajeng Rizka Silvia, Ratna Indah Safitri, Indra Dharma Wijaya, dan Chandrasena Setiadi tahun 2023. Sistem informasi pendaftaran PMI berbasis *website* yang dirancang untuk membantu PT.Hendrarta Argaraya dalam proses pendaftaran dan seleksi calon PMI. Sistem ini memiliki fitur registrasi akun, pengisian formulir, *upload* dokumen, pembayaran, seleksi, pengumuman, dan cetak kartu identitas PMI. Sistem dapat membantu perusahaan dalam meningkatkan efisiensi, transparansi, dan pelayanan kepada calon PMI.

Tabel 2. 1 Tinjauan Pustaka

No.	Judul	Comparing	Contrasting	Critisize	Synthesize	Summarize
1.	<i>Design and Build a Quality Assurance Document Archiving Application Using the Rapid Application Development</i> [13].	Sistem Informasi Pengarsipan Dokumen berbasis Website untuk Jaminan Kualitas Program Studi Teknik Informatika, Universitas PGRI Madiun, dibuat menggunakan <i>framework</i> Laravel dengan metode penelitian Rapid Application Development	Pengujian sistem hanya menggunakan metode <i>Blackbox Testing</i>	Pada bagian analisis kebutuhan sistem dapat dilengkapi dengan melakukan wawancara dan observasi kepada pihak-pihak terkait, seperti perusahaan, pengguna, dan instansi terkait.	Dapat dilakukan penelitian lain dalam hal pengembangan fitur yang telah dibuat agar tampilan antarmuka pengguna dan usabilitynya lebih optimal dan mudah digunakan. Situs web yang telah dibuat dapat dikembangkan dengan menggabungkan sistem informasi manajemen (MIS) yang sudah ada.	Penelitian menggunakan metode RAD untuk Sistem Informasi Pengarsipan Dokumen Jaminan Kualitas berbasis Web dengan Laravel. Sistem ini memfasilitasi pengarsipan dokumen dengan pengelompokan kriteria, fitur pencarian, dan arsip digital yang dapat direkapitulasi. Hasil pengujian <i>blackbox</i> menunjukkan sistem berjalan tanpa kesalahan.

No.	Judul	Comparing	Contrasting	Critisize	Synthesize	Summarize
2.	<i>Design And Build Company Profile Website Bagaskara Art Store Wonosobo</i> [14].	Rancang bangun <i>website company profile</i> Sanggar Seni Bagaskara Wonosobo untuk memberikan gambaran tentang perusahaan, visi dan misi, juga produk atau jasa yang ditawarkan.	Menggunakan metode penelitian <i>Extreme Programming</i>	Pada kesimpulan, penulis lebih fokus menjelaskan bagaimana pengaruh Extreme Programming terhadap rancang bangun <i>website company profile</i> Sanggar Seni Bagaskara Wonosobo sehingga belum mencakup keseluruhan isi dari penelitian yang telah dilakukan. Selain itu masih ada kesalahan penulisan pada beberapa kata seperti "aabout".	Pada penelitian ini dapat dilakukan penelitian lain didalam rancang bangun <i>website</i> Sanggar Seni Bagaskara Wonosobo, seperti menyelaraskan tampilan <i>User Interface(UI) frontend</i> dan <i>backend</i> agar sesuai dengan tema warna yang dimiliki oleh Sanggar Seni Bagaskara Wonosobo.	<i>Website</i> ini punya dua tampilan, yaitu admin dan pelanggan. Admin dapat mengelola data dan melihat pesanan dari semua pelanggan. Pelanggan dapat mengakses halaman utama, tentang, berita, portofolio, memesan pelatihan, dan mengedit akun.
3.	<i>Designing a web-based career system using the laravel framework</i> [15].	Membuat pengaturan aplikasi lowongan pekerjaan berbasis <i>website</i> yang disebut	Menggunakan metode penelitian Waterfall	Aktor yang ada didalam diagram use case maupun <i>diagram activity</i> hanya melibatkan <i>Employee</i> dan <i>Company</i> . Aktor	Dapat dilakukan penelitian lain dalam hal melakukan perbandingan dengan aplikasi serupa yang sudah	Aplikasi ini memberikan berbagai manfaat, seperti memudahkan perolehan data tentang industri yang membuka lowongan

No.	Judul	Comparing	Contrasting	Critisize	Synthesize	Summarize
		program pencarian web, untuk memudahkan pelamar didalam menemukan data lowongan pekerjaan dan mempermudah industri yang mencari karyawan dengan cepat dan keterampilan yang sesuai.		Admin yang memiliki fungsi untuk mengelola dan mengontrol user masih belum ada.	ada untuk menentukan kelebihan dan kekurangan dari aplikasi yang sudah ada.	pekerjaan, mendistribusikan surat pernyataan pembebasan tanggung jawab bagi pelamar tanpa harus membawa banyak dokumen fisik, memungkinkan pelamar profesional langsung melamar ke industri pilihan tanpa janji terlebih dahulu, dan memberikan peluang rekrutmen langsung dari industri yang tertarik.
4.	Implementasi <i>Framework</i> Laravel Pada Rancang Bangun <i>Website</i> IAKN Palangka Raya Dengan Metode Prototype[16].	Membuat <i>website</i> institusi sebagai media pengelolaan dan penyampaian informasi terkait IAKN Palangka Raya kepada	Mengimplementasikan <i>framework</i> Laravel pada rancang bangun <i>website</i> IAKN Palangkaraya	Dalam presentasi <i>website</i> yang disediakan oleh penulis, hanya menampilkan bagian depan (front-end) tanpa adanya sistem pengelolaan (back-	Tersedia potensi untuk melakukan penelitian lebih lanjut dalam upaya meningkatkan fungsionalitas <i>website</i> .	Semua fungsi dan fitur pada <i>website</i> berjalan optimal sebagai media institusi IAKN Palangka Raya. Informasi terkini disajikan menarik,

No.	Judul	Comparing	Contrasting	Critisize	Synthesize	Summarize
		masyarakat secara mudah, tepat dan cepat sesuai dengan perkembangan teknologi.	dengan metode <i>Prototype</i>	end) yang berfungsi untuk memperbarui konten yang ditampilkan di bagian depan tersebut. Oleh karena itu, dapat disimpulkan bahwa website tersebut belum memiliki fungsionalitas back-end untuk melakukan pembaruan isi kontennya.	Pengembangan ini melibatkan peningkatan melalui penambahan fitur-fitur seperti optimisasi mesin pencari (SEO) dengan pengelolaan metadata dan integrasi Google Analytics, pengoptimalan gambar, implementasi sistem pengelolaan konten di sisi backend, kemampuan berbagi halaman, penyediaan galeri album, serta penyusunan repositori berkas.	meningkatkan kualitas layanan bagi masyarakat dan civitas akademika secara mudah, akurat, dan efisien.

No.	Judul	Comparing	Contrasting	Critisize	Synthesize	Summarize
5.	Rancang Bangun Sistem Informasi Desa Berbasis Web Menggunakan Metode <i>Rapid Application Development</i> (RAD) Pada Desa Sukatani, Kecamatan Cikande, Kabupaten Serang – Banten[17].	Merancang dan membangun sistem informasi desa berbasis web dengan menggunakan Metode <i>Rapid Application Development</i> (RAD)	Sistem informasi desa berbasis web dengan penerapan fitur informasi yang ditampilkan dalam web Desa Sukatani meliputi Profil Desa, Pemerintahan Desa, Lembaga Masyarakat, Data Kependudukan, Sarana & Prasarana, dan Kegiatan Desa yang diharapkan dapat membantu mewujudkan peran desa, Sukatani, Kecamatan Cikande, Kabupaten Serang – Banten.	Pada poin 1.2 Teori, penjelasan yang diberikan masih terlalu singkat dan belum memberikan informasi pendukung yang kuat.	Dapat dilakukan penelitian lain dalam hal melakukan pengembangan dengan menambah fitur-fitur dan tampilan yang sesuai dengan kebutuhan ke depannya.	Sistem informasi desa berbasis <i>website</i> yang telah dirancang dan dibangun oleh penulis ini dapat menjadi media publikasi informasi desa, sehingga masyarakat mendapatkan kemudahan dalam hal mengakses informasi desa.

No.	Judul	Comparing	Contrasting	Critisize	Synthesize	Summarize
6.	Implementasi Metode <i>Waterfall</i> Dalam Pembangunan <i>Company Profile Website</i> Akademi Komunitas Dharma Bhakti Bangka (Ak Dbb)[18].	<i>Website</i> Akademi Komunitas Dharma Bhakti Bangka (AK DBB) merupakan sebuah media untuk mempublikasikan data atau informasi terkait dengan kampus AKDBB seperti informasi profil, program studi, prestasi dosen serta mahasiswa.	Pembangunan <i>Website</i> Akademi Komunitas Dharma Bhakti Bangka (AK DBB) menggunakan metode <i>waterfall model</i> .	Penggunaan metode <i>Waterfall</i> pada pembangunan <i>website</i> Akademi Komunitas Dharma Bhakti Bangka mungkin tidak sepenuhnya tepat, karena metode ini cenderung bersifat linier dan kurang fleksibel terhadap perubahan informasi dan kebutuhan yang mungkin terus berkembang di masa mendatang.	Dapat dilakukan penelitian lain dalam hal melakukan analisis ke-efektifan <i>website</i> bagi Akademi Komunitas Dharma Bhakti Bangka didalam menyebarkan informasi kepada masyarakat.	<i>Website</i> AKDBB ini telah di bangun menggunakan <i>framework</i> bootstrap agar tampilan <i>website</i> menjadi <i>responsive</i> mengikuti platform yang digunakan pengunjung. Mempermudah masyarakat dalam mencari informasi yang berhubungan dengan AKDBB
7.	Perancangan Sistem Pendaftaran Berbasis Web Menggunakan Metode RAD Terhadap Peserta <i>Training</i> Industri Kopi Dan Aneka Minuman	Aplikasi berbasis web dengan menggunakan metode RAD yang bermanfaat bagi peserta disabilitas untuk memudahkan dalam proses pendaftaran dan	Aplikasi berbasis web yang dibangun menggunakan bahasa pemrograman PHP dan database MySQL	Pada poin hasil dan pembahasan, belum ada penjelasan tentang fitur apa saja yang dapat membantu para peserta disabilitas. Fitur dan tampilan yang diberikan hanya memberitahu secara	Dapat dilakukan penelitian lain dalam hal mengetahui apakah aplikasi ini sudah diterima oleh peserta disabilitas maupun masyarakat, seta	Aplikasi pendaftaran pelatihan kerja dirancang untuk memberikan kemudahan kepada peserta, termasuk penyandang disabilitas, serta masyarakat umum. Aplikasi ini

No.	Judul	Comparing	Contrasting	Critisize	Synthesize	Summarize
	Pengelola Umkm (Penyandang Disabilitas)[19].	pelaksanaan program menjadi peserta pelatihan kerja disabilitas.		umum dan belum terperinci.	apakah aplikasinya sudah dapat memberikan kepuasan terhadap penggunaannya.	menyediakan informasi lengkap mengenai jadwal dan prosedur pelatihan, mempermudah proses pendaftaran, dan memandu peserta dalam mengikuti pelatihan kerja.
8.	Perancangan Website Media Informasi dan Pemesanan pada PT.Trita Musi Prasada dengan Metode RAD[20].	<i>Website</i> media informasi ini dirancang menggunakan metode RAD dan dibuat untuk mempermudah perusahaan dalam mengenalkan perusahaan kepada masyarakat dan menawarkan produk sekaligus sebagai media	Alur RAD yang digunakan adalah Requirements Planning, RAD Design Workshop, dan Implementation.	Tampilan <i>website</i> yang diberikan masih terbilang sederhana dan kurang menarik minat konsumen umum untuk mengunjungi <i>website</i> tersebut. Serta penulis tidak menampilkan tampilan yang lengkap pada setiap menu yang ada didalam <i>front-end</i> nya.	Dapat dilakukan penelitian lain dalam hal membuat <i>website</i> PT.Tirta Musi Prasada dengan menggunakan sebuah <i>framework</i> PHP seperti Laravel misalnya untuk memberikan tampilan yang lebih menarik untuk memikat konsumen yang	Dengan adanya <i>Website</i> Media Informasi dan Pemesanan pada PT.Trita Musi Prasada ini dapat membantu bagian <i>marketing</i> untuk meningkatkan dan memperluas jangkauan pemasaran produk dan memperkenalkan perusahaan secara tidak langsung.

No.	Judul	Comparing	Contrasting	Critisize	Synthesize	Summarize
		pemesanan produk.			mengunjungi <i>website</i> .	
9.	Sistem Informasi Pendaftaran PMI Berbasis Website (Studi Kasus Pada PT.Hendrarta Argaraya Kota Malang)[21].	Sistem Informasi Pendaftaran PMI dengan basis Web untuk membantu mengatur data pendaftaran, dalam mempermudah pemrosesan PMI dengan lebih cepat, tepat, dan efisien.	Sistem Informasi Pendaftaran PMI dengan basis Web dibuat menggunakan metode Waterfall dan diuji menggunakan kuisioner.	Pada bagian analisis kebutuhan sistem hanya menjelaskan secara singkat kebutuhan sistem dari pihak PT.Hendrarta Argaraya dan calon PMI. Analisis kebutuhan sistem yang lebih lengkap akan membantu peneliti untuk merancang sistem yang lebih sesuai dengan kebutuhan pengguna.	Dapat dilakukan penelitian lain dalam hal mengembangkan sistem informasi untuk platform mobile dan menambahkan fitur lupa password untuk user, fitur ini berguna untuk user yang lupa atau tidak mengingat password yang dimiliki.	Sistem pendaftaran PMI berhasil melakukan input data user, mengelola data negara tujuan dan pendaftar, memberikan notifikasi status proses tahap pertama dan kedua calon PMI, mengelola berkas-berkas dan sertifikasi wawancara, serta mengelola tes kemampuan bahasa calon PMI dengan hasil tes yang sesuai.

Berdasarkan tinjauan literatur yang dilakukan, dapat disimpulkan bahwa penelitian sebelumnya telah menunjukkan pentingnya membangun *website* yang dapat menyediakan sebuah informasi yang dimiliki oleh suatu perusahaan, dengan menggunakan berbagai metode, fitur, dan metode testing yang digunakan untuk meningkatkan hasil pembangunan *website*. Selain itu, metode *Rapid Application Development* (RAD) telah terbukti cocok digunakan untuk proyek berukuran kecil hingga menengah, karena fokus pada kecepatan pengembangan dan keterlibatan pengguna secara berulang dengan tujuan dapat menyediakan informasi terkait prosedur dan langkah-langkah menjadi Calon Pekerja Migran Indonesia (CPMI) melalui PT.Karyatama Mitra Sejati cabang Banyumas. Selain itu, pemilihan metode pengujian *Blackbox* untuk sistem membuktikan efektivitasnya dalam mengevaluasi fungsionalitas sistem secara keseluruhan tanpa memerlukan pengetahuan mendalam mengenai struktur internal kode. Sementara itu, penggunaan *System Usability Scale* (SUS) sebagai metode evaluasi *usability* menunjukkan keunggulan dalam menyediakan pengukuran subjektif terhadap pengalaman pengguna. Kombinasi kedua metode ini diharapkan dapat memberikan gambaran menyeluruh tentang kualitas dan performa *website* yang dikembangkan.

2.2. Landasan Teori

2.2.1. Hypertext Preprocessor (PHP)

Hypertext Preprocessor (PHP) adalah bahasa pemrograman skrip sisi server yang dirancang khusus untuk pengembangan *website*. PHP digunakan untuk membuat halaman web dinamis dengan kemampuan berinteraksi dengan *database* dan menghasilkan konten yang responsif berdasarkan permintaan pengguna[22]. Selain berfungsi sebagai bahasa pemrograman *server-side*, PHP juga dapat disematkan ke dalam HTML untuk mengontrol aspek-aspek dinamis pada halaman *website*. Dikenal dengan sintaks yang mirip dengan C dan *Perl*, PHP mendukung berbagai *platform* sistem operasi seperti *Windows*, *Linux*, dan *macOS*[23]. Dengan kemampuannya yang luas dan sintaks yang familiar, PHP menjadi pilihan utama

untuk mengembangkan *website* yang dinamis dan responsif, memungkinkan para pengembang untuk menciptakan pengalaman pengguna yang interaktif.

PHP memiliki berbagai *framework* populer yang dapat mempercepat dan menyederhanakan proses pengembangan aplikasi web. Beberapa *framework* PHP terkemuka meliputi *Laravel*, *Symfony*, *CodeIgniter*, dan *Yii*. *Laravel* dikenal dengan kemudahan penggunaan dan fitur-fitur canggihnya[24], sedangkan *CodeIgniter* menonjolkan kecepatan dan kebersihannya[25]. *Symfony* memberikan fleksibilitas tinggi dan fokus pada komponen yang dapat digunakan ulang[26], sementara *Yii* menawarkan performa tinggi dan dukungan bawaan untuk fitur-fitur seperti *caching* dan *RESTful API*[27]. Keberagaman pilihan ini memungkinkan pengembang untuk memilih *framework* yang paling sesuai dengan tujuan dan preferensi pengembangan mereka.

2.2.2. Hyper Text Markup Language (HTML)

HyperText Markup Language merupakan salah satu dari berbagai bahasa pemrograman yang digunakan dalam pengembangan situs. Penggunaan tanda tag < > oleh HTML secara dominan digunakan untuk menandai kode-kode yang akan diinterpretasikan oleh browser, memungkinkan halaman ditampilkan dan disusun sesuai dengan tata letak yang telah diatur[28]. Bahasa HTML ini secara utama digunakan untuk membantu dalam merancang struktur dasar halaman situs web. Dalam analogi, HTML dapat diibaratkan sebagai fondasi awal yang membantu menyusun kerangka dasar halaman situs web dengan lebih terstruktur, sebelum memasuki tahap desain dan penambahan fungsionalitas.

HTML terus berkembang dengan penambahan elemen-elemen baru dan fitur-fitur yang mendukung tampilan yang semakin dinamis dan interaktif. Dalam pengembangan situs modern, HTML juga sering bekerja beriringan dengan CSS (*Cascading Style Sheets*) untuk memberikan desain yang estetis dan *Bootstrap* untuk *responsivitas* yang lebih baik[29]. Keseluruhan, pemahaman mendalam terhadap HTML menjadi pondasi penting bagi pengembang web untuk menciptakan pengalaman pengguna yang memikat dan fungsional.

2.2.3. Website

Website adalah suatu aplikasi yang menyajikan dokumen multimedia, termasuk teks, gambar, suara, animasi, dan video, dengan menggunakan protokol *Hyper Text Transfer Protocol*(HTTP). Akses ke *website* dilakukan melalui perangkat lunak yang disebut *browser*, seperti *Internet Explorer*, *Mozilla Firefox*, *Opera*, dan *Safari*. *Browser* atau peramban, berfungsi menerjemahkan dokumen web dengan bantuan komponen internal yang disebut *web engine*[30]. Proses ini memungkinkan tampilan semua dokumen web setelah diterjemahkan.

Sebuah *website* bersifat statis jika informasinya tetap dan jarang berubah, dengan arah informasi yang hanya dari pemilik website ke pengguna. Sebaliknya, website bersifat dinamis jika informasinya selalu berubah dan interaktif dua arah antara pemilik dan pengguna[31]. Dengan demikian, *website* tidak hanya sebagai sarana penyajian informasi, tetapi juga sebagai alat interaktif yang memfasilitasi keterlibatan pengguna, menciptakan pengalaman yang lebih mendalam dan relevan dalam dunia maya.

2.2.4. Apache Web Server

Apache Web Server, sering disebut sebagai *Apache*, merupakan sebuah *server* web sumber terbuka yang secara luas diterima dan digunakan di seluruh dunia[32]. Dikenal karena keandalan dan kinerjanya yang unggul, *Apache* mendukung berbagai fitur dan modul, sehingga memberikan fleksibilitas kepada administrator server untuk mengonfigurasi *server* sesuai dengan kebutuhan spesifik, termasuk dukungan untuk *server-side scripting* menggunakan bahasa seperti PHP dan *Perl*[33]. Dengan dukungan untuk *server-side scripting*, seperti PHP dan *Perl*, *Apache* memberikan keleluasaan dalam mengonfigurasi *server* sesuai dengan kebutuhan spesifik pengelola.

Keunggulan utama *Apache* adalah kemampuannya dalam menangani lalu lintas web dengan efisien, serta kemampuan untuk diintegrasikan dengan mudah dengan berbagai teknologi dan *platform*[34]. *Apache* juga dikenal memiliki tingkat keamanan yang tinggi, menjadikannya pilihan utama untuk organisasi yang

mengutamakan keamanan dalam pengelolaan *server* web mereka[35]. Dengan dukungan yang terus berkembang dari komunitas *open-source*, *Apache* terus menjadi salah satu *server* web terdepan dalam industri.

2.2.5. *My Structured Query Language (MySQL)*

My Structured Query Language (MySQL) sebuah sistem manajemen basis data relasional (RDBMS) yang bersifat *open-source*, cepat, dan andal. Dikembangkan oleh *Oracle Corporation*, MySQL menjadi salah satu RDBMS yang paling populer di dunia, digunakan oleh berbagai aplikasi dan situs web, mulai dari skala kecil hingga besar[36]. Secara menyeluruh, MySQL menyediakan solusi yang handal dan efisien dalam manajemen basis data relasional. Dikembangkan oleh *Oracle Corporation*, MySQL telah meraih popularitas di berbagai aplikasi dan situs web, mampu mengatasi berbagai skala penggunaan dari tingkat kecil hingga besar.

MySQL menggunakan bahasa *query Structured Query Language (SQL)* untuk mengelola dan memanipulasi data dalam basis data. Kelebihan MySQL termasuk kinerja tinggi, skalabilitas, dan kemudahan penggunaan[37]. Dengan dukungan transaksi, indeks, dan fitur-fitur lainnya, MySQL dapat memenuhi kebutuhan berbagai jenis aplikasi, baik yang bersifat sederhana maupun kompleks. Kesuksesan dan penerimaan luas MySQL dalam industri membuatnya menjadi pilihan yang dapat diandalkan untuk organisasi dan pengembang perangkat lunak.

2.2.6. *System Development Life Cycle (SDLC)*

System Development Life Cycle (SDLC) merujuk pada serangkaian tahapan pekerjaan yang dilakukan oleh analis sistem dan programmer dalam rangka membangun sistem informasi. Metodologi SDLC sendiri adalah suatu proses pembuatan dan perubahan pada sistem, terutama pada Sistem Komputer atau Sistem Informasi. Proses ini terdiri dari tahapan-tahapan terstruktur yang melibatkan perencanaan, analisis, desain, implementasi, pengujian, dan pemeliharaan[38]. *System Development Life Cycle* bukan hanya sekadar

sekumpulan langkah teknis, tetapi juga sebuah pendekatan yang menyeluruh untuk mengelola dan mengembangkan sistem informasi

Tahap perencanaan melibatkan perumusan strategi dan tujuan proyek, sementara analisis fokus pada pemahaman mendalam terhadap kebutuhan sistem. Desain menghasilkan kerangka kerja visual sistem, diikuti oleh implementasi dan pengujian untuk memastikan keberhasilan dan keandalan sistem. Pemeliharaan kemudian terjadi setelah sistem berjalan, memastikan bahwa sistem tetap optimal dan dapat disesuaikan dengan perubahan kebutuhan yang mungkin timbul[39]. Dengan demikian, SDLC menjadi landasan yang kokoh dalam memastikan pengembangan sistem informasi berjalan secara terstruktur dan efisien.

2.2.7. Rapid Application Development (RAD)

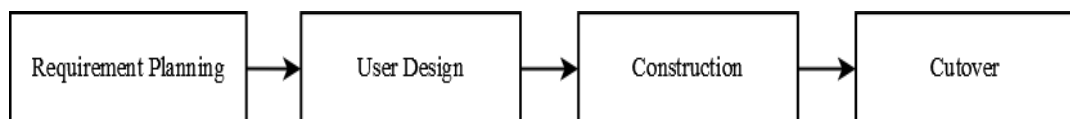
James Martin merupakan seorang ahli dalam industri IT, dikenal sebagai penemu *Rapid Application Development*(RAD). Konsep RAD pertama kali diperkenalkan oleh James Martin pada tahun 1980-an dan sejak itu, metodologi ini telah menjadi pilihan populer dalam pengembangan perangkat lunak, terutama untuk proyek-proyek dengan jadwal yang ketat[40][41]. Kontribusi James Martin sebagai penemu *Rapid Application Development* (RAD) telah memberikan dampak signifikan dalam pengembangan perangkat lunak. Metodologi RAD telah menjadi metode yang populer terutama dalam proyek-proyek dengan tenggat waktu yang ketat.

Rapid Application Development (RAD) merupakan suatu pendekatan berorientasi objek dalam pengembangan sistem informasi yang melibatkan metode pengembangan dan perangkat lunak. Metode ini memfokuskan pada pembuatan sistem dengan waktu singkat. Menggunakan pendekatan iteratif, di mana model kerja sistem dibangun di awal tahap pengembangan untuk menetapkan kebutuhan pengguna. Keunggulan RAD juga terletak pada keterlibatan pengguna dalam proses pengembangan, memastikan kebutuhan sistem terpenuhi sesuai dengan keinginan pengguna[42]. Dengan demikian, RAD bukan hanya memungkinkan pengembangan sistem dalam waktu singkat melalui pendekatan iteratif, tetapi juga

menonjolkan peran aktif pengguna dalam memastikan kebutuhan sistem sesuai dengan keinginan mereka.

Kelebihan yang dimiliki RAD adalah kemampuan untuk menggunakan kembali komponen yang ada, integrasi proses yang lebih cepat, kemudahan penyesuaian kebutuhan pengguna, dan mengurangi kemungkinan kesalahan. Namun, kekurangannya mencakup kebutuhan akan kolaborasi tim yang kuat, komitmen yang kuat antara pengembang dan stakeholder, cocok hanya untuk proyek kecil dengan waktu pengerjaan singkat, dan lebih sesuai untuk aplikasi modular terpisah dengan fokus pada suatu fitur[43]. Secara menyeluruh, kelebihan RAD, seperti pemanfaatan kembali komponen, integrasi yang efisien, dan penyesuaian mudah terhadap kebutuhan pengguna harus diimbangi dengan pertimbangan matang terkait kolaborasi tim yang kokoh dan keterbatasannya pada proyek kecil.

Metode RAD juga memiliki kekurangan, termasuk kebutuhan akan kolaborasi tim yang kuat dan komitmen yang tinggi antara pengembang dan *stakeholder*. Metode ini lebih cocok untuk proyek kecil dengan waktu pengerjaan singkat dan tidak sesuai untuk proyek skala besar atau kompleks yang memerlukan analisis mendalam[44]. Maka dari itu, didalam memilih metode pengembangan seperti RAD, perlu dilakukan evaluasi cermat terhadap karakteristik proyek yang bersangkutan, sehingga kelebihan dan kekurangannya dapat dioptimalkan sesuai dengan kebutuhan dan skala proyek yang dihadapi.



Gambar 2. 1 Alur Metode *Rapid Application Development*[40]

Tahapan *Rapid Application Development* adalah sebagai berikut :

1) *Requirement Planning*

Dalam fase ini, data dikumpulkan dan kebutuhan pengguna dianalisis. Data yang terkumpul melibatkan informasi mengenai proses bisnis, sistem

yang sedang berjalan, dan kebutuhan pengguna. Analisis kebutuhan pengguna bertujuan untuk mendapatkan pemahaman yang mendalam mengenai kebutuhan pengguna serta menentukan fitur-fitur yang akan diimplementasikan dalam sistem yang akan dikembangkan[40].

2) *User Design*

Dalam tahap ini, sistem direncanakan dan dirancang menggunakan berbagai alat bantu seperti diagram *use case*, *class diagram*, *activity diagram*, dan *sequence diagram*[40].

3) *Construction*

Dalam tahap ini, sistem dikonstruksikan sesuai dengan desain yang telah disiapkan. Proses konstruksi dapat menggunakan berbagai bahasa pemrograman, seperti PHP, Java, atau Python. Untuk penelitian yang akan dilakukan akan menggunakan bahasa pemrograman PHP dengan *framework* Laravel[40].

4) *Cutover*

Dalam tahap *Cutover*, dilakukan pengujian dan peluncuran aplikasi ke dalam lingkungan organisasi. Pada fase ini, para pengembang perangkat lunak memverifikasi kinerja aplikasi dan memastikan bahwa aplikasi tersebut memenuhi kebutuhan pengguna. Mereka juga memiliki kesempatan untuk memberikan masukan dan menyetujui sistem[40].

2.2.8. Laravel framework

Laravel, sebagai *framework* bahasa pemrograman *Hypertext Preprocessor* (PHP) yang dikembangkan oleh Taylor Otwell pada tahun 2011, mengintegrasikan berbagai fitur terbaik dari kerangka kerja seperti CodeIgniter, Yii, ASP.NET MVC, Ruby on Rails, Sinatra, dan lainnya. *Framework* ini menawarkan *bundle*, migrasi, dan *Command Line Interface* (CLI) Artisan yang menyediakan seperangkat alat dan arsitektur aplikasi[45]. Dengan menyajikan *bundle* fitur terbaik dan alat

pengembangan yang canggih, Laravel membuktikan diri sebagai pilihan utama dalam pengembangan aplikasi web berbasis PHP.

Fitur-fitur modern seperti *Eloquent ORM*, *Blade*, *Migration*, *Query Builder*, *Middleware Resource Controller*, dan *Automatic Pagination*, Laravel memberikan kemudahan dalam pengembangan aplikasi web. Keunggulan lainnya termasuk kecepatan pengembangan yang ditingkatkan dan penggunaan *package manager* PHP *Composer*, menjadikan Laravel sebagai pilihan yang sangat relevan dan bermanfaat dalam dunia pengembangan web *modern*[46]. Secara keseluruhan, Laravel tidak hanya memberikan kemudahan teknis, tetapi juga mendorong praktik pengembangan yang baik dan efisien.

2.2.9. Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah sebuah bahasa standar yang digunakan dalam pengembangan perangkat lunak untuk menggambarkan, merancang, dan mendokumentasikan sistem secara visual[47]. Dengan menggunakan UML, pengembang dapat menciptakan model visual yang membantu dalam analisis, perancangan, dan komunikasi ide dan konsep terkait perangkat lunak.



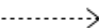
Unified Modeling Language (UML) menyediakan sejumlah diagram grafis, seperti *use case diagram* untuk menggambarkan interaksi antara sistem dan pengguna, *activity diagram* untuk menggambarkan aktivitas atau alur kerja dalam suatu sistem atau proses bisnis, *sequence diagram* untuk memvisualisasikan urutan interaksi atau komunikasi yang terjadi antar objek, serta *class diagram* untuk merepresentasikan struktur kelas dalam sistem[48]. Dengan demikian, UML menjadi alat yang sangat efektif dalam memfasilitasi pemahaman yang konsisten dan bersama-sama antara para pemangku kepentingan dalam pengembangan perangkat lunak.


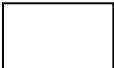
2.2.10. Use Case Diagram

Diagram *Use Case* adalah representasi pemodelan yang penting untuk menggambarkan perilaku sistem informasi yang sedang dikembangkan. Penggunaan Diagram *Use Case* bertujuan memberikan gambaran yang rinci tentang cara berinteraksi antara satu atau lebih aktor dengan sistem yang sedang dikembangkan. Dalam diagram ini, aktor-aktor seperti pengguna, sistem eksternal, atau elemen lainnya digunakan untuk mengidentifikasi fungsi-fungsi khusus yang dapat diakses oleh aktor-aktor tersebut[49]. Diagram *Use Case* sangat bermanfaat dalam memahami perilaku sistem dan menjelaskan secara detail interaksi antara entitas-entitas yang terlibat.

Diagram *Use Case* memberikan gambaran menyeluruh tentang cara berbagai fungsi dalam sistem informasi dapat dimanfaatkan oleh pihak-pihak yang terlibat. Lebih jauh, Diagram *Use Case* membantu pemahaman tentang interaksi sistem dengan elemen-elemen eksternal dan bagaimana aktor-aktor yang terlibat dalam sistem dapat memanfaatkan fungsionalitas sesuai dengan kebutuhan[50]. Dengan demikian, Diagram *Use Case* menjadi alat yang esensial dalam merinci dan memahami interaksi sistem informasi dengan aktor-aktor yang terlibat. *Node-node* yang terdapat pada *Use Case Diagram* dapat diidentifikasi melalui Tabel 2.2 yang menyajikan informasi mengenai nama, gambar simbol, dan keterangan masing-masing elemennya :

Tabel 2. 2 *Node Use Case Diagram*

No	Simbol	Keterangan
1	<p><i>Use Case</i></p> 	Simbol yang menggambarkan interaksi-interaksi antar aktor (pengguna) dan tujuannya untuk menjelaskan fungsionalitas dan perilaku sistem yang akan dikembangkan.
2	<p><i>Actor</i></p> 	Menggambarkan peran khusus yang dimainkan oleh pengguna saat terlibat dalam interaksi dengan <i>use case</i> .
3	<p><i>Dependency</i></p> 	Hubungan ini mengacu pada suatu perubahan dimana suatu unsur yang mandiri akan

No	Simbol	Keterangan
		berpengaruh terhadap unsur yang tidak memiliki otonomi.
4	<p><i>Include</i></p> <p>-----></p> <p><<include>></p>	Simbol ini mewakili hubungan di mana <i>use case</i> tambahan terdapat di dalam <i>use case</i> lain, dimana <i>use case</i> tersebut ditambahkan dengan <i>use case</i> utama yang digunakan untuk melakukan beberapa fungsionalitas.
5	<p><i>Generalization</i></p> <p>←</p>	Hubungan <i>generalisasi da spesialisasi</i> (umum-khusus) antara dua <i>use case</i> menggambarkan bahwa satu fungsi merupakan fungsi yang lebih umum dibandingkan dengan yang lainnya.
6	<p><i>Extend</i></p> <p>←-----</p> <p><<extend>></p>	Menggambarkan <i>use case</i> target dari <i>use case</i> sumber dapat bekerja lebih baik pada titik yang telah ditentukan sebelumnya.
7	<p><i>Collaboration</i></p> <p>○</p>	Menggambarkan interaksi antara aturan-aturan dan elemen-elemen lain dalam sebuah <i>use case</i> yang saling bekerja bersama untuk menciptakan tindakan yang lebih besar daripada jumlah dan kontribusi individu yang disebut sinergi.
8	<p><i>Note</i></p> <p></p>	Mengacu entitas fisik yang ada saat aplikasi sedang berjalan, penggunaan tersebut mewakili sumber daya komputasi.
9	<p><i>System</i></p> <p></p>	Menggambarkan sebuah paket yang memberikan gambaran sistem dengan cakupan yang terbatas.
10	<p><i>Association</i></p> <p>—</p>	Menggambarkan koneksi antara satu objek dengan objek lainnya.







2.2.11. Activity Diagram

Diagram aktivitas, yang juga dikenal sebagai diagram aliran kerja, merupakan representasi visual yang mengilustrasikan urutan langkah-langkah atau aktivitas dalam suatu sistem, proses bisnis, atau menu perangkat lunak. Keunggulan diagram aktivitas terletak pada kemampuannya untuk menggambarkan bagaimana sistem secara internal beroperasi dan menjalankan berbagai tindakan atau proses yang terkait[49]. Perlu dicatat bahwa fokus diagram aktivitas bukanlah pada aktivitas

yang dilakukan oleh aktor, melainkan lebih berorientasi pada aktivitas sistem itu sendiri.

Diagram aktivitas menyajikan representasi visual yang menggambarkan alur kerja internal suatu entitas, seperti sistem perangkat lunak, proses bisnis, atau sistem lainnya. Diagram ini mampu mencakup langkah-langkah khusus, keputusan, garis waktu, dan kondisi-kondisi khusus yang terjadi selama operasi sistem[51]. Dengan adanya diagram aktivitas, pemangku kepentingan dapat dengan mudah memahami serta menganalisis urutan aktivitas sistem, termasuk hubungan antar aktivitas yang memengaruhi jalannya proses secara menyeluruh. *Node-node* yang terdapat pada *Activity Diagram* dapat dilihat pada tabel 2.3 yang berisi nama dan gambar simbol beserta keterangannya :

Tabel 2. 3 *Node Activity Diagram*

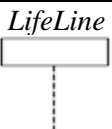


No	Simbol	Keterangan
1	<i>Initial Node</i> 	Bagaimana objek suatu aktivitas pada sistem diawali.
2	<i>Activity Final Node</i> 	Menggambarkan suatu aktivitas pada sistem dibentuk atau diakhiri.
3	<i>Activity</i> 	Menggambarkan interaksi antarmuka kelas yang saling terhubung.
4	<i>Action</i> 	Menunjukkan kondisi sistem yang mencerminkan implementasi suatu tindakan.
5	<i>Fork Node</i> 	Perubahan suatu aliran menjadi beberapa aliran pada tahapan tertentu.
6	<i>Decision</i> 	Merupakan suatu percabangan asosiasi, di mana terdapat penggabungan lebih dari satu aktivitas menjadi satu kesatu.

2.2.12. Sequence Diagram

Sequence Diagram adalah jenis diagram UML yang menunjukkan urutan interaksi antara objek-objek dalam sistem. Diagram ini memvisualisasikan bagaimana objek-objek berkomunikasi satu sama lain dan bagaimana pesan-pesan dikirim di antara mereka selama berjalannya suatu proses atau fungsi sistem. Setiap objek direpresentasikan oleh sebuah kotak vertikal, dan panah-panah mengindikasikan alur pesan atau panggilan metode antar objek[52]. *Sequence Diagram* memberikan gambaran visual yang jelas tentang bagaimana objek-objek berinteraksi dan bertukar informasi dalam sistem.

Dalam *Sequence Diagram*, urutan aktivitas antar objek disusun dari atas ke bawah, menciptakan representasi temporal dari proses yang dijelaskan. Selain itu, diagram ini memuat elemen-elemen seperti *lifeline* (garis vertikal yang mencirikan masa hidup objek), pesan (garis panah yang menunjukkan pesan yang dikirim), serta *fragment* (blok pemisah untuk mengelompokkan pesan-pesan terkait)[53]. Dengan representasi temporal yang disajikan secara berurutan, *Sequence Diagram* membantu pengembang perangkat lunak memahami alur eksekusi suatu proses atau fungsi dalam sistem secara rinci. *Node-node* yang terdapat pada *Sequence Diagram* beserta penjelasannya dapat dilihat pada tabel 2.4 :

Tabel 2. 4 *Node Sequence Diagram*

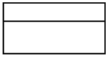

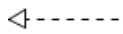


No	Simbol	Keterangan
1		Menggambarkan suatu entitas objek saling berhubungan satu sama lain.
2		Menjelaskan interaksi antar objek meliputi informasi mengenai kegiatan yang sedang terjadi.
3		Menjelaskan rincian mengenai komunikasi antar objek-objek, termasuk informasi terkait kegiatan yang terjadi.

2.2.13. Class Diagram

Class Diagram merupakan komponen kunci dalam UML yang memvisualisasikan struktur statis suatu sistem. Diagram ini menunjukkan kelas-kelas yang terlibat dalam sistem, bersama dengan atribut-atribut dan metode-metode yang dimiliki oleh masing-masing kelas[54]. Dengan menggunakan notasi berupa persegi panjang untuk merepresentasikan kelas-kelas, *Class Diagram* membantu pengembang untuk memahami hubungan dan hierarki antar objek dalam sistem.

Class Diagram juga dapat menampilkan asosiasi antar kelas, menggambarkan bagaimana satu kelas berhubungan dengan kelas lainnya. Selain itu, *multiplicity* pada *Class Diagram* dapat menunjukkan jumlah objek yang dapat berpartisipasi dalam asosiasi tertentu[55]. Dengan demikian, *Class Diagram* memberikan pandangan komprehensif terhadap struktur objek dan hubungannya dalam suatu sistem berbasis objek. *Node-node* yang muncul dalam *Class Diagram* dapat ditemukan melalui Tabel 2.5 yang memuat detail seperti nama, gambar simbol, dan keterangan untuk setiap elemen :

Tabel 2. 5 *Node Class Diagram*

No	Simbol	Keterangan
1	<p><i>Class</i></p> 	Mengacu pada sekumpulan objek dengan karakteristik dan fungsi yang sama.
2	<p><i>Collaboration</i></p> 	Menggambarkan rangkaian tindakan yang dilaksanakan oleh sistem sehingga hasil dapat diukur oleh aktor yang berkaitan.
3	<p><i>Realization</i></p> 	Menggambarkan tindakan atau operasi yang dilakukan oleh objek.
4	<p><i>Nary Association</i></p> 	Merujuk pada usaha untuk menghindari hubungan asosiasi yang melibatkan lebih dari dua objek.
5	<p><i>Association</i></p> 	Mengilustrasikan keterkaitan antara satu objek dengan objek-objek yang lainnya.

No	Simbol	Keterangan
6	<i>Dependency</i> ----->	Hubungan ini mengacu pada suatu perubahan dimana suatu unsur yang mandiri akan berpengaruh terhadap unsur yang tidak memiliki otonomi.
7	<i>Generalization</i> —————>	Hubungan <i>generalisasi da spesialisasi</i> (umum-khusus) antara dua <i>use case</i> menggambarkan bahwa satu fungsi merupakan fungsi yang lebih umum dibandingkan dengan yang lainnya.

2.2.14. *Blackbox Testing*

Black-box testing adalah metode pengujian yang mengevaluasi fungsionalitas suatu sistem tanpa memeriksa detail kode program dan desain internal. Dalam pengujian ini, fokus utama adalah pada evaluasi eksternal sistem, tanpa memerlukan pengetahuan mendalam tentang rincian internal program atau struktur desainnya[11]. Pendekatan *black-box testing* memperlakukan sistem sebagai kotak hitam, di mana perhatian utama diberikan pada *input* dan *output* yang dihasilkan oleh sistem.

Black-box testing dilakukan dari perspektif pengguna untuk mengidentifikasi ambiguitas atau inkonsistensi dalam spesifikasi persyaratan sistem. Dengan kata lain, pendekatan *black-box testing* berfokus pada evaluasi sistem dari luar, meniru cara pengguna akhir berinteraksi dengan aplikasi, dengan tujuan memastikan bahwa fungsionalitasnya sesuai dengan harapan[56]. Pengujian melalui metode *black-box* dilakukan tanpa memerlukan pengetahuan tentang implementasi atau struktur internal sistem, lebih menekankan pada pemahaman terhadap respons sistem terhadap *input* tertentu dan sejauh mana *output* yang dihasilkan sesuai dengan spesifikasi yang telah ditetapkan.

Berikut adalah tabel rincian dari metode *blackbox testing* :

Tabel 2. 6 Blackbox Testing

No	Pengujian	<i>Test Case</i>	Hasil yang Diharapkan	Hasil Pengujian	Kesimpulan
1.	Deskripsi singkat tentang pengujian yang dilakukan.	Skenario atau langkah-langkah uji yang dijalankan pada sistem.	Hasil yang seharusnya muncul jika pengujian berhasil.	Realisasi hasil yang diperoleh dari pengujian.	Penilaian keseluruhan terhadap hasil pengujian.
2.					
Dst.					

Dalam menghadapi keputusan penggunaan metode pengujian, perbandingan antara *Blackbox testing*, *Whitebox testing*, dan *Graybox testing* menjadi krusial. Pertimbangan kriteria seperti Pengetahuan Internal Kode, Tujuan Pengujian, Level Pengujian, Kecepatan Implementasi, dan Kemudahan Pelaksanaan menjadi panduan utama. Di bawah ini merupakan tabel perbandingan antara ketiga metode tersebut:

Tabel 2. 7 Perbandingan Blackbox, Whitebox, dan Graybox[57][58][59]

Kriteria Perbandingan	<i>Blackbox Testing</i>	<i>Whitebox Testing</i>	<i>Graybox Testing</i>
Pengetahuan Internal Kode	Tidak memerlukan pengetahuan internal kode.	Memerlukan pengetahuan internal kode.	Bergantung pada tingkat pengetahuan internal kode yang diperlukan.
Tujuan Pengujian	Fokus pada fungsi dan <i>output</i> sistem tanpa memerhatikan logika internal.	Memeriksa struktur internal dan logika dari suatu komponen atau sistem.	Kombinasi antara fokus pada fungsionalitas dan pengetahuan

			internal tertentu.
Level Pengujian	Diterapkan pada <i>level</i> pengujian sistem atau integrasi.	Diterapkan pada <i>level</i> pengujian unit, integrasi, dan sistem.	Dapat diterapkan pada <i>level</i> pengujian unit, integrasi, dan sistem.
Kecepatan Implementasi	Implementasi lebih cepat karena tidak memerlukan pengetahuan mendalam tentang kode.	Memerlukan waktu implementasi yang lebih lama karena memerlukan analisis kode yang mendalam.	Implementasi dapat lebih cepat daripada <i>Whitebox</i> testing, tetapi lebih lambat dibandingkan dengan <i>Blackbox</i> testing.
Kemudahan Pelaksanaan	Mudah dilaksanakan dan memerlukan sedikit koordinasi dengan tim pengembang.	Memerlukan koordinasi yang lebih erat dengan tim pengembang dan mungkin membutuhkan sumber daya tambahan.	Memerlukan koordinasi, tetapi lebih fleksibel dan memerlukan sumber daya yang lebih sedikit dibandingkan <i>Whitebox</i> testing.

Dipilihnya metode *Blackbox testing* didasarkan pada beberapa pertimbangan kriteria perbandingan. Metode ini optimal karena tidak memerlukan pengetahuan internal kode, sesuai dengan tujuan pengujian untuk mengidentifikasi bug dari perspektif pengguna. *Blackbox testing* dapat diterapkan pada berbagai tingkat pengujian, cepat diimplementasikan tanpa memerlukan analisis kode internal, dan mudah dilaksanakan oleh penguji dengan berbagai tingkat keahlian. Keputusan ini diambil untuk mencapai efisiensi dan efektivitas dalam pengujian proyek.

2.2.15. System Usability Scale

System Usability Scale (SUS) adalah metode evaluasi penggunaan sistem yang dikembangkan oleh John Brooke pada tahun 1986[60]. Metode ini digunakan untuk mengukur tingkat kepuasan pengguna terhadap suatu sistem atau produk. *System Usability Scale* terdiri dari 10 pertanyaan dengan skala likert yang berkisar antara 1 hingga 5, di mana 1 menunjukkan tingkat ketidaksetujuan penuh dan 5 menunjukkan tingkat setuju penuh. Setelah mengumpulkan jawaban dari pengguna, skor individu diubah menjadi skor standar dan dihitung untuk mendapatkan skor SUS akhir[12]. Skor SUS kemudian dapat diinterpretasikan untuk menentukan sejauh mana sistem atau produk dianggap mudah digunakan oleh pengguna. Metode ini sering digunakan dalam pengembangan perangkat lunak untuk mendapatkan umpan balik yang berguna tentang usability produk.

Pengujian *Usability Testing* sangat penting dilakukan untuk bahan referensi kelayakan aplikasi. Cara mengukur *usability* adalah dengan perhitungan kelayakan yang diperoleh dari skor jawaban dibandingkan terhadap skor yang diharapkan[61].

Berikut adalah tabel tabel 2.8 yang berisi rincian pertanyaan yang akan diajukan didalam metode *System Usability Scale*(SUS) :

Tabel 2. 8 Instrumen Pertanyaan SUS[62]

No	Pertanyaan	Skor
1.	Saya berpikir akan menggunakan sistem ini lagi	1 - 5
2.	Saya merasa sistem ini rumit untuk digunakan	1 - 5
3.	Saya merasa sistem ini mudah digunakan	1 - 5
4.	Saya membutuhkan bantuan dari orang lain atau teknisi dalam menggunakan sistem ini	1 - 5
5.	Saya merasa fitur-fitur sistem ini berjalan dengan semestinya	1 - 5
6.	Saya merasa ada banyak hal yang tidak konsisten (tidak serasi pada sistem ini).	1 - 5
7.	Saya merasa orang lain akan memahami cara menggunakan sistem ini dengan cepat.	1 - 5
8.	Saya merasa sistem ini membingungkan.	1 - 5
9.	Saya merasa tidak ada hambatan dalam menggunakan sistem ini	1 - 5
10.	Saya perlu banyak belajar sebelum menggunakan aplikasi ini.	1 - 5

Berikut adalah rincian dari kuantitas reaksi responden pada tabel 2.9 :

Tabel 2. 9 Rincian Kualitas Responden

Jawaban	Skor
Sangat tidak setuju	1
Tidak setuju	2
Ragu-ragu	3
Setuju	4
Sangat setuju	5

Contoh menghitung *score* menggunakan metode SUS pada Rumus 2.1 :

$$x = \frac{\sum x}{n} \quad (2.1)$$

X = rata-rata score

$\sum x$ = jumlah keseluruhan nilai *SUS*

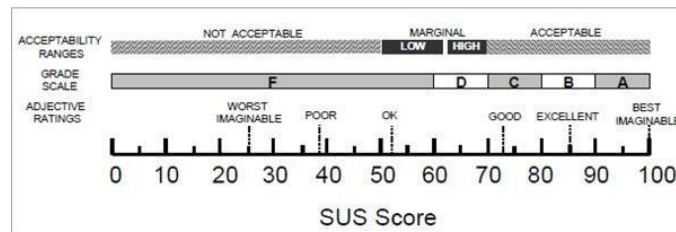
n = banyaknya responden

Terdapat dua metode untuk menilai nilai, yang pertama adalah mengklasifikasikan nilai penerimaan pengguna menjadi *acceptable*, *not acceptable*, dan *marginal* serta membagi skala menjadi enam, yaitu A, B, C, D, E, dan F. Sementara metode kedua melibatkan skor *percentile* (SUS) yang terdiri dari skala A, B, C, D, E. Skor adjektif juga dibagi menjadi *best imaginable*, *excellent*, *good*, *ok*, *poor*, dan *worst imaginable*[63]. Tabel 2.10 dibawah ini mengandung informasi mengenai Score Percentile Rank dari metode SUS dan untuk parameter SUS dapat dilihat pada gambar 2.2.

Tabel 2. 10 Score Precentile Rank

Grade	Keterangan
A	Skor $\geq 80,3$
B	Skor ≥ 74 dan $< 80,3$

C	Skor ≥ 68 dan < 74
D	Skor ≥ 51 dan < 68
E	Skor < 51



Gambar 2. 2 Parameter SUS

Pendekatan *System Usability Scale* (SUS) seringkali menjadi pilihan utama dalam mengukur *usability* suatu sistem dengan memanfaatkan kuesioner. Metode ini memberikan sejumlah keunggulan yang mencakup kesederhanaan implementasi, pengukuran subjektif berbasis persepsi pengguna, serta skalabilitas yang memungkinkan aplikasinya pada berbagai jenis sistem. Meski demikian, pemilihan metode pengujian *usability* tetaplah bergantung pada tujuan proyek dan kebutuhan spesifik dari konteks pengembangan yang bersangkutan. Keuntungan-keuntungan tersebut meliputi :

- 1) *System Usability Scale* dapat dengan mudah diimplementasikan dan diterapkan dalam berbagai konteks tanpa memerlukan pengetahuan teknis yang mendalam[64]. Keunggulan ini memungkinkan evaluasi *usability* yang efisien tanpa memakan banyak waktu atau sumber daya.
- 2) Metode *System Usability Scale* memfokuskan pada persepsi pengguna terhadap *usability* suatu sistem, yang memberikan gambaran *subjektif* tentang pengalaman pengguna[65]. Pendekatan ini mampu memberikan wawasan yang lebih menyeluruh terkait kualitas penggunaan sistem.
- 3) Metode *System Usability Scale* dapat digunakan dalam berbagai jenis aplikasi atau sistem, termasuk situs web, aplikasi mobile, atau perangkat

lunak desktop[66]. Karena itu, *System Usability Scale* bersifat skalabel dan dapat diterapkan pada berbagai konteks pengembangan.

- 4) *System Usability Scale* telah digunakan secara luas dan diakui secara internasional, sehingga hasilnya dapat dibandingkan dengan berbagai studi usability lainnya[67]. Oleh karena itu, *System Usability Scale* memberikan standar yang dapat diukur untuk mengukur *usability*.
- 5) Hasil dari kuesioner *System Usability Scale* dapat dengan cepat diolah dan memberikan skor numerik yang dapat diinterpretasikan[68]. Oleh karena itu, memudahkan penarikan kesimpulan tentang tingkat *usability* sistem.

Dengan keunggulan-keunggulan seperti kesederhanaan implementasi, pengukuran subjektif berbasis persepsi pengguna, skalabilitas, dan kemampuan analisis yang cepat, metode *System Usability Scale (SUS)* telah terbukti menjadi pilihan yang efektif dalam mengukur usability suatu sistem.

2.2.16. Populasi dan Sampel

Populasi dalam penelitian merujuk pada keseluruhan elemen atau individu yang memenuhi karakteristik tertentu dan menjadi fokus penelitian. Dalam konteks penelitian, populasi dapat mencakup semua orang, objek, atau peristiwa yang relevan dengan pertanyaan penelitian yang diajukan[69]. Sebagai contoh, dalam penelitian mengenai preferensi konsumen terhadap produk tertentu, populasi mencakup seluruh konsumen potensial dari produk tersebut. Pemahaman yang mendalam tentang populasi sangat penting untuk memberikan dasar yang kuat dalam merancang dan menginterpretasi hasil penelitian.

Sampel, di sisi lain, adalah representasi terpilih dari populasi yang diambil untuk mewakili keseluruhan. Pemilihan sampel dilakukan untuk memungkinkan peneliti mengumpulkan data dengan lebih efisien dan ekonomis tanpa harus menyelidiki seluruh populasi[70]. Pengambilan sampel yang dilakukan secara cermat dapat memastikan bahwa data yang dikumpulkan benar-benar mencerminkan seluruh populasi. Dengan begitu, hasil generalisasi dari sampel

dapat diandalkan untuk memberikan gambaran yang akurat terkait keseluruhan populasi. Dengan demikian, pemilihan sampel yang cermat menjadi kunci untuk memastikan keakuratan dan keandalan temuan penelitian.

2.2.17. Rumus Slovin

Rumus *Slovin* adalah formula pengambilan sampel yang dapat mewakili atau disebut representatif dari populasi tertentu dengan tingkat kepercayaan dan *margin of error* yang telah ditentukan[71]. Rumus ini sangat berguna dalam penelitian survei untuk memastikan bahwa sampel yang diambil dapat merepresentasikan populasi dengan akurasi yang memadai.

Rumus *Slovin* dinyatakan pada Rumus 2.2 dibawah ini :

$$n = \frac{N}{1 + (N \times e^2)} \quad (2.2)$$

n = ukuran sample

N = ukuran populasi

e = *margin of error*

Dalam penelitian ini, telah didapat 50 orang dari data pendaftar CPMI pada PT.Karyatama Mitra Sejati Cabang Banyumas bulan Januari sampai Februari tahun 2024. Dalam penerapan rumus *Slovin*, pemilihan *margin of error* (e) sangat penting karena berpengaruh langsung pada ukuran sampel yang diperoleh. *Margin of error* yang lebih kecil akan menghasilkan ukuran sampel yang lebih besar, yang berarti hasil survei akan lebih akurat, namun membutuhkan lebih banyak sumber daya[72]. Sebaliknya, *margin of error* yang lebih besar akan menghasilkan ukuran sampel yang lebih kecil, yang berarti survei dapat dilakukan dengan lebih cepat dan murah, tetapi dengan tingkat presisi yang lebih rendah[70]. Dalam penelitian ini, batas toleransi kesalahan ditetapkan sebesar 10% [73]. Dengan demikian, penggunaan rumus *Slovin* membantu peneliti menyeimbangkan antara akurasi dan efisiensi dalam pengambilan sampel.