

## **BAB II**

### **TINJAUAN PUSTAKA DAN LANDASAN TEORI**

#### **2.1. Penelitian Sebelumnya**

Penelitian yang dilakukan sebelumnya untuk mengatasi masalah penyalahgunaan narkoba telah banyak dilakukan. Dan banyak diantara penelitian tersebut yang memiliki dampak yang relatif berguna dalam mencegah penyalahgunaan narkoba di Indonesia. Berikut penelitian terdahulu yang menurut penulis mempunyai keterkaitan dengan penelitian yang akan dilakukan.

Pertama, penelitian berjudul “Media Informasi Berbasis Android Tentang Jenis Jenis Narkoba di Badan Narkotika Nasional Provinsi Kalimantan Timur” yang dilakukan oleh Muhammad Ridha Febrian, Nataniel Dengen, dan Bambang Cahyono pada tahun 2019. Penelitian ini bertujuan untuk membuat rancang bangun media informasi berbasis multiplatform yang tujuannya untuk menyebarkan informasi tentang narkoba kepada masyarakat. Dalam proses rancang bangun media informasi ini, digunakan metode pengembangan *waterfall* yang melibatkan tahapan perencanaan, analisis kebutuhan, desain, pengkodean, *testing*, dan implementasi. Penelitian ini menghasilkan bahwa semakin mudahnya *user* memperoleh informasi untuk mencegah penyalahgunaan narkoba di kalangan masyarakat [6].

Kedua, penelitian berjudul “Strategi Badan Narkotika Nasional Dalam Penanggulangan Narkotika di Masa Pandemi Covid-19 di Indonesia” yang dilakukan oleh Ernita Dewi, Khalida Ulfa, dan Safirussalim pada tahun 2022. Penelitian ini dilakukan untuk menganalisis tentang strategi apa yang diterapkan oleh BNN dalam pencegahan dan peredaran narkoba selama era pandemi. Dengan menggunakan metode kualitatif dan kuantitatif dari sumber data sosial media seperti jurnal, twitter, dan *website*. Hasil penelitian ini menunjukkan bahwa startegi BNN RI dilakukan dengan 3 langkah, yaitu *Soft Power Approach* dengan kegiatan penyuluhan narkoba, rehabilitasi, dan pemberdayaan masyarakat, *Hard Power Approach* dengan fokus pada

penegakan hukum dalam kasus penyalahgunaan narkoba, dan Pendekatan cerdas energi yang mengintegrasikan teknologi informasi dalam era *digital* untuk memberantas peredaran narkotika. [7].

Ketiga, penelitian berjudul “Penerapan React JS Pada Pengembangan *Front end* Aplikasi Startup Ubaform” yang dilakukan oleh Nasution dan Lizda Iswari pada tahun 2021. Penelitian ini bertujuan untuk meningkatkan performa sistem pada startup yang bergerak pada bidang pengembangan *website*. Pengembangan *website* menggunakan library Javascript yaitu ReactJS yang bersifat SPA (*Single Page Application*). Hasil yang didapatkan adalah mode SPA yang telah terinclude didalam reactJS dapat meningkatkan kinerja sistem pada startup serta mengurangi waktu dan biaya infrastruktur pengembangan [8].

Keempat, penelitian berjudul “Penerapan *Mobile-First Design* pada Antarmuka *Website* Profil Sekolah Menggunakan Metode *Human-Centred Design*” yang dilakukan oleh Adnan Zulkarnain pada tahun 2019. Penelitian ini dilakukan perbaikan pada *website* profil sekolah dengan mengimplementasikan teknik *Mobile-First-Design* menggunakan metode pengembangan *interface* yang disebut *Human-Centered-Design*. Hasil yang didapatkan adalah pengujian *usability testing* menunjukkan bahwa desain *prototype interface* memiliki tingkat kebergunaan sebesar 86,66%. Hal ini menunjukkan bahwa perubahan yang dilakukan pada tampilan *website* profil sekolah terbukti dapat meningkatkan *user interface* dan *user experience website* tersebut [9].

Kelima, penelitian berjudul “Analisis Metode *Agile* pada Pengembangan Sistem Informasi Berbasis *Website: Systematic Literature Review*” yang dilakukan oleh Sausan Hidayah Nova, Aris Puji Widodo, dan Budi Warsito pada tahun 2022. Penelitian ini dibuat dengan tujuan untuk melakukan analisis model *agile* yang diterapkan untuk membuat *website* sistem informasi dengan cara membandingkan dan menganalisis jurnal-jurnal yang telah terbit. Penelitian ini menggunakan metode SLC atau yang familiar disebut *System Literature Review*. SLC digunakan untuk mengevaluasi jurnal dengan

sistematis dan terstruktur, juga Langkah-langkahnya berpatokan pada prosedur yang telah dibuat. Hasilnya adalah model *agile* yang mayoritas digunakan oleh para developer adalah model *agile* scrum. Aspek bisnis menjadi fokus utama dalam implementasi metode *agile* dalam pengembangan sebuah sistem informasi [5].

Keenam, penelitian berjudul “Analisis Metode Pengujian Perangkat Lunak *Blackbox Testing* dan Pemodelan *Diagram Uml* Pada Aplikasi *Veterinary Services* yang Dikembangkan Dengan Model *Waterfall*” yang dilakukan oleh Muhamad Syarif dan Eri Bayu Pratama pada tahun 2021. Dalam penelitian ini, metode pengujian yang digunakan adalah metode *black box testing* untuk menguji fungsionalitas aplikasi *Veterinary Services* dan penelitian ini mendapatkan hasil bahwa Menerapkan pemodelan *diagram UML* dalam mengembangkan aplikasi dengan metode *agile* *waterfall* bisa mengurangi potensi kesalahan atau *error* yang kemungkinan dapat terjadi pada fase implementasi atau pengujian unit. Hal ini disebabkan oleh pemahaman stakeholder terhadap cakupan sistem yang disajikan secara visual pada *diagram use case*. Selain itu, kesalahan fungsional yang dapat dilakukan oleh programmer dapat diminimalkan karena bantuan dari *diagram class* dan *sequence* yang dijabarkan secara rinci, serta uji dengan metode *blackbox* memberikan hasil yang cukup memuaskan karena dari segi fungsionalitas, sistem dapat dijalankan dengan baik. Ini juga menandakan bahwa kepastian akan kebutuhan stakeholder terpenuhi secara sistem.

Ketujuh, penelitian berjudul “Pengujian *Usability Website* Pondok Pesantren Qodratullah Menggunakan *System Usability Scale*” yang dilakukan oleh Fitri Purwaningtias dan Usman Ependi pada tahun 2020. Penelitian ini dilakukan evaluasi untuk melihat perspektif pengguna terhadap *website*, dalam penelitian ini digunakan metode pengujian *System Usability Scale (SUS)* dan Evaluasi menyatakan bahwa *website* Pondok Pesantren Qodratullah meraih nilai akhir sebesar 88. Dengan nilai tersebut, *website* tersebut diberikan penilaian kata sifat "*excellence*," masuk ke dalam kelompok B dalam skala

nilai, dan dinyatakan dapat diterima (*acceptable*) dalam tingkat penerimaan. [10].

Kedelapan, penelitian berjudul “Pengembangan Produk Media Komik *Digital* Untuk Pencegahan Penyalahgunaan Narkoba Bagi Siswa SMA Di Kota Salatiga” yang dilakukan oleh David Wicaksono, Yari Dwikurnaningsih, dan Setyorini pada tahun 2022. Penelitian ini dilakukan dengan maksud untuk sebagai rancang bangun produk komik *digital* untuk Langkah pencegahan penyalahgunaan narkotika dan obat-obatan pada strata remaja di kota Salatiga. Metode yang digunakan dalam penelitian ini adalah kualitatif serta kuantitatif yang didapatkan dari saran, komentar, dan kritikan oleh ahli uji. Hasil penilaian dari ahli materi BK menunjukkan bahwa indikator kegunaan memiliki nilai 85,7%, indikator kelayakan memiliki nilai 65%, dan indikator ketepatan memiliki nilai 87,5%. Sementara itu, hasil penilaian dari ahli media menunjukkan bahwa indikator bentuk fisik memiliki nilai 75%, indikator penokohan memiliki nilai 50%, indikator kegrafisan memiliki nilai 68,75%, dan indikator warna memiliki nilai 65%. Lalu, hasil uji dari ahli yang dilakukan di lokasi objek penelitian, yaitu SMA Kristen Satya Wacana serta SMA Negeri 1 Salatiga menunjukkan bahwasannya nilai dari indikator dari segi kegunaan atau fungsionalitas memiliki nilai 94%. indikator kelayakan memiliki nilai 83,75%, dan indikator ketepatan memiliki nilai 68,75%. Berdasarkan hasil yang didapat, penulis dapat menyimpulkan bahwa produk berbentuk komik *digital* dengan judul “Jauhi Narkoba, Remaja Hebat, Muda Sehat” telah sesuai dengan kriteria yang telah di rencanakan serta layak dioperasikan sebagai media berbasis *digital* dalam layanan konseling. [11].

**Tabel 2. 1** Ringkasan Penelitian Sebelumnya

No	Judul, Penulis dan Tahun Terbit	Metode / Algoritma	Hasil	Perbedaan dengan penelitian yang dilakukan
1	Media Informasi Berbasis Android Tentang Jenis Jenis Narkoba Di Badan Narkotika Nasional Provinsi Kalimantan Timur Muhammad Ridha Febrian, Nataniel Dengen, dan Bambang Cahyono 2019 .[6]	Metode Pengembangan <i>n Waterfall</i>	Semakin mudahnya user dalam memperoleh informasi dalam rangka pencegahan penyalahgunaan narkoba di kalangan masyarakat	Penelitian sebelumnya menggunakan metode waterfall untuk membangun <i>website</i> dari awal, sementara penelitian ini fokus merancang <i>front end</i> dengan metode <i>Mobile First Design</i> menggunakan ReactJS.

2	<p>Penerapan React JS Pada Pengembangan Front end Aplikasi Startup Ubaform Nasution dan Lizda Iswari 2021. [8]</p>	<p>Pengembangan <i>n website front end</i> dengan pendekatan <i>Single Page Application (SPA)</i></p>	<p>Hasil yang didapatkan adalah mode SPA yang telah terinclude didalam reactJS dapat meningkatkan kinerja sistem pada startup serta mengurangi waktu dan biaya infrastruktur pengembangan.</p>	<p>Penelitian ini memanfaatkan ReactJS untuk mengembangkan <i>website</i> secara massif guna mendukung Pencegahan dan Pemberantasan Penyalahgunaan dan Peredaran Narkoba (P4GN) di desa.</p>
3.	<p>Strategi Badan Narkotika Nasional Dalam Penanggulangan Narkotika di Masa Pandemi Covid-19 di Indonesia Ernita Dewi, Khalida Ulfa, dan Safirussalim</p>	<p>Kualitatif dan Kuantitatif</p>	<p>Penelitian ini menunjukkan strategi BNN RI dalam tiga langkah: Soft Power Approach dengan kegiatan penyuluhan, rehabilitasi, dan pemberdayaan masyarakat; Hard Power Approach dengan penegakan hukum terhadap penyalahgunaan narkoba; dan Smart Power Approach dengan pemanfaatan</p>	<p>Penelitian sebelumnya menganalisis strategi BNN RI dalam menangani narkotika selama pandemi COVID-19 dengan pendekatan Soft Power, Hard Power, dan Smart Power.</p>

	2022. [7]		teknologi informasi untuk memberantas narkoba.	
4.	Pengembangan Produk Media Komik <i>Digital</i> Untuk Pencegahan Penyalahgunaan Narkoba Bagi Siswa SMA Di Kota Salatiga David Wicaksono, Yari Dwikurnaningsih, dan Setyorini 2022[11]	Kualitatif dan Kuantitatif	Hasil penilaian dari ahli materi BK menunjukkan bahwa indikator kegunaan memiliki nilai 85,7%, indikator kelayakan memiliki nilai 65%, dan indikator ketepatan memiliki nilai 87,5%. Sementara itu, hasil penilaian dari ahli media menunjukkan bahwa indikator bentuk fisik memiliki nilai 75%, indikator penokohan memiliki nilai 50%, indikator kegrafisan memiliki nilai 68,75%, dan indikator warna memiliki nilai 65%.	Penelitian sebelumnya memanfaatkan Komik <i>Digital</i> untuk mencegah penyalahgunaan Narkoba,

No	Judul, Penulis dan Tahun Terbit	Metode / Algoritma	Hasil	Perbedaan dengan penelitian yang dilakukan
5.	Analisis Metode <i>Agile</i> pada Pengembangan Sistem Informasi Berbasis Website: <i>Systematic Literature Review</i> Sausan Hidayah Nova, Aris Puji Widodo, Budi Warsito 2022. [5]	<i>Systematic Literature Review (SLR)</i>	SLR 2021: Model paling umum dalam metode <i>agile</i> untuk pengembangan <i>website</i> adalah scrum, dengan fokus pada bidang bisnis. Pemodelan UML pada model waterfall dapat mengurangi kesalahan pada implementasi dan pengujian unit. Penggunaan <i>diagram use case</i> , class, dan <i>sequence</i> membantu meminimalkan kesalahan fungsional. Uji <i>blackbox</i> menunjukkan hasil positif, memastikan pemenuhan kebutuhan stakeholder oleh sistem.	Penelitian sebelumnya menganalisis metode <i>agile</i> berdasarkan literatur jurnal,
6.	Analisis Metode Pengujian	Metode Pengujian	Pemodelan UML pada pengembangan perangkat lunak	Penelitian sebelumnya menganalisis penerapan metode SDLC Waterfall,



	Perangkat Lunak <i>Blackbox Testing</i> dan Pemodelan <i>Diagram Uml</i> Pada <i>Aplikasi Veterinary Services</i> yang Dikembangkan Dengan <i>Model Waterfall</i> Muhamad Syarif, Eri Bayu Pratama 2021. [12]	<i>Blackbox Testing</i>	dengan model waterfall mengurangi potensi kesalahan pada implementasi dan pengujian unit. Visualisasi cakupan sistem melalui <i>diagram use case</i> membantu pemahaman stakeholder, sedangkan <i>diagram class</i> dan <i>sequence</i> membantu mengurangi kesalahan fungsional. Hasil uji <i>blackbox testing</i> positif, menjamin pemenuhan kebutuhan stakeholder oleh sistem.	sementara penelitian ini mengadopsi metode SDLC <i>Agile</i> dalam perancangan sistem informasi untuk Badan Narkotika Nasional Kabupaten Banyumas.
7.	Pengujian <i>Usability Website</i> Pondok Pesantren Qodratullah Menggunakan <i>System Usability Scale</i>	Metode Pengujian <i>Usability</i> dengan <i>System Usability Scale</i>	<i>Website</i> Pondok Pesantren Qodratullah memiliki tingkat <i>usability</i> yang tinggi, dinilai 88 dari skala 1 hingga 100 menurut penilaian responden. <i>Website</i> ini juga masuk dalam kategori dapat diterima, seiring dengan tingkat	Penelitian sebelumnya fokus pada objek <i>website</i> Pondok Pesantren Qodratullah,

	Fitri Purwaningtias, Usman Ependi 2020. [10]		penerimaan yang termasuk dalam kategori acceptable. Berdasarkan nilai adjective rating, <i>website</i> ini dapat dikategorikan sebagai <i>excellence</i> bagi pengunjungnya.	
8.	Penerapan Mobile-First <i>Design</i> pada Antarmuka <i>Website</i> Profil Sekolah Menggunakan Metode Human-Centred <i>Design</i> Adnan Zulkarnain 2019. [9]	Mobile-First- <i>Design</i>	Hasil <i>usability testing</i> menunjukkan tingkat kebergunaan desain prototipe interface mencapai 86,66%, menunjukkan peningkatan pada user interface dan pengalaman pengguna <i>website</i> profil sekolah.	Penelitian sebelumnya menerapkan metode Mobile-First- <i>Design</i> pada <i>website</i> profil sekolah di SMPN 21 Malang.

Tinjauan kedelapan penelitian sebelumnya sangat relevan dengan penelitian ini, terutama dalam hal topik pembahasan yang serupa. Namun, salah satu perbedaan utama antara penelitian ini dan penelitian sebelumnya adalah penggunaan studi kasus yang berbeda. Dengan mempertimbangkan temuan-temuan penelitian sebelumnya, penelitian ini memiliki potensi untuk memberikan wawasan baru dan kontribusi yang berbeda terhadap pemahaman dan penanggulangan penyalahgunaan narkoba di Indonesia khususnya Kabupaten Banyumas.

## 2.2.Landasan Teori

### 2.2.1. *Desa Bebas Narkoba*

Desa Bebas Narkoba atau yang familiar disebut dengan desa bersinar juga biasa disebut perangkat wilayah setingkat Kelurahan/Desa yang memenuhi kriteria tertentu, di mana program Pencegahan dan Pemberantasan Penyalahgunaan dan Peredaran Gelap Narkoba (P4GN) diimplementasikan secara besar-besaran Petunjuk Teknis Pelaksanaan Desa Bebas Narkoba Desa Bersinar ini disusun dan diimplementasikan oleh serta untuk melibatkan warga sipil, pemerintah, dan non-pemerintah, dan sektor swasta. Mereka berperan dalam memberikan fasilitasi, pendampingan, dan pembinaan sebagai bagian dari pelaksanaan program ini. [13].

### 2.2.2. *Metode Agile*

Metode *agile* biasa didefinisikan sebagai suatu metode dalam Siklus Hidup Pengembangan Perangkat Lunak (SDLC) skema incremental serta iterative sudah menjadi hal yang mendasar. *Agile* adalah metode SDLC yang mengimplementasi sprint kecil yang berkelanjutan. Ini membuat pendekatan *agile* terbilang fleksibel dengan sebuah perubahan ketika sedang mendevlop sebuah software/perangkat lunak. [5]Metode *Agile* juga sering di deffinisikan sebagai sebuah pendekatan pengembangan perangkat lunak yang berfokus pada kerja kolaboratif, fleksibilitas, dan adaptabilitas. Dalam metode ini, tim pengembang bekerja secara iteratif dan inkremental, memecah proyek menjadi serangkaian siklus pengembangan yang disebut "*sprint*". Setiap sprint memiliki tujuan yang jelas dan waktu yang terbatas untuk menyelesaikan sejumlah pekerjaan yang telah ditentukan sebelumnya. Pendekatan ini dapat mengakselerasi tim dalam sebuah proyek agar dapat beradaptasi dan memberikan evaluasi sebuah project dengan lebih cepat, hal ini dikarenakan pendekatan ini dapat me-manage sebuah prioritas sesuai. Komunikasi yang terbuka dan intensif antara anggota tim juga menjadi kunci dalam metode *Agile*, dengan mengadopsi praktik-praktik seperti *daily stand-up meetings* dan *retrospective meetings* untuk memastikan kelancaran proyek. Metode *Agile*

juga mendorong penggunaan prototipe yang dapat diuji oleh pengguna sejak awal, sehingga memungkinkan iterasi dan perbaikan berkelanjutan. Dengan pendekatan ini, metode *Agile* memberikan fleksibilitas dan transparansi dalam pengembangan perangkat lunak, memungkinkan tim untuk menghadapi perubahan yang tidak terduga dengan lebih efektif dan dapat menyediakan hasil yang sesuai dengan keinginan pengguna dengan lebih baik. Metode *agile* memiliki beberapa tahapan yaitu *requirement*, *design*, *develop*, *test*, dan *deploy* [5].

#### 2.2.2.1. Requirement

Pada fase awal ini, manajer proyek akan mengevaluasi persyaratan yang diperlukan untuk proyek tersebut. Penting untuk dicatat bahwa dalam tahap perencanaan, harus ada klarifikasi mengenai cakupan dan tujuan pembuatan aplikasi perangkat lunak. Tujuannya adalah agar tim dapat mengembangkan perangkat lunak dengan efektif tanpa deviasi dari tujuan awal. Pada tahap ini, tim pengembang juga terlibat dalam pengumpulan dan analisis kebutuhan pengguna. *Requirement* yang telah dirinci akan diatur dan didokumentasikan dalam sebuah dokumen SRS (*Software Requirement Specification*) atau BRD (*Business Requirement Document*).

**Tabel 2. 2** Tabel Model Requirement Pada Dalam Metode Agile

Kebutuhan	Keterangan
Registrasi dan <i>Login</i>	<ul style="list-style-type: none"> <li>• <i>Admin</i> dapat membuat akun user</li> <li>• User dapat membuat akun sendiri</li> </ul>
<i>CRUD</i> User	<ul style="list-style-type: none"> <li>• .....</li> <li>• .....</li> </ul>

#### 2.2.2.2. Design

Pada tahap ini, *designer* akan mempersiapkan sejumlah dokumen desain sistem dan *software* sesuai dengan dokumen spesifikasi kebutuhan di tahap sebelumnya. Tahapan ini akan menghasilkan

prototype dan beberapa output lain meliputi dokumen berisi desain, pola, dan komponen yang diperlukan untuk mewujudkan proyek tersebut sehingga dapat membantu untuk menentukan arsitektur sistem secara keseluruhan. Pembuatan *design* dapat berupa UML (Undefined Language Model), Wireframe, Flowchart, dll.

#### 2.2.2.3. *Develop*

Tahap ini merupakan tahap penulisan program. Biasanya, jika proyeknya kecil, program dapat ditulis oleh satu developer. Namun, jika proyek besar dapat dipecah dan diselesaikan oleh banyak tim. Pada tahap ini, Access Control atau Source Code Management juga dapat digunakan untuk membantu developer melacak perubahan kode dan memastikan kompatibilitas antara proyek tim yang berbeda untuk mencapai tujuan mereka.

Proses pengkodean tidak terbatas pada kompilasi kode agar aplikasi dapat berjalan, tetapi juga mencakup banyak tugas lainnya, seperti menemukan dan memperbaiki *error* atau *glitches* yang ditemukan, dan menulis dokumentasi, seperti panduan pengguna atau bantuan FAQ untuk pertanyaan seputar teknis. Fase ini merupakan salah satu fase dalam siklus hidup pengembangan perangkat lunak yang membutuhkan waktu lebih lama.

#### 2.2.2.4. *Test*

Setelah selesai proses pengembangan perangkat lunak, tim pengujian harus melakukan uji untuk menilai kinerja optimal produk. Fase pengujian perangkat lunak bertujuan untuk mengevaluasi apakah produk yang telah dikembangkan beroperasi dengan baik sesuai kebutuhan pengguna. Aktivitas pengujian ini melibatkan Quality Assurance (QA), tim pengembang, dan mungkin juga melibatkan pengguna akhir yang akan menggunakan produk tersebut. Test dapat berupa fungsional test atau *usability test*.

#### 2.2.2.5. *Deploy*

Setelah menangani permasalahan seperti bug dan defect yang muncul sebelumnya, perangkat lunak telah mencapai tingkat kesiapan yang memungkinkannya dirilis ke pasar dan digunakan oleh pengguna. Proses ini mungkin menantang dan kompleks, terutama ketika beberapa perusahaan perlu melakukan pembaruan database untuk menyelaraskan dengan aplikasi yang baru dikembangkan, mengakibatkan penggunaan waktu dan upaya yang lebih besar.

#### 2.2.3. *Website*

*Website* adalah koleksi *page* yang dapat memperlihatkan informasi teks, animas, gambar, video, dan suara. *Page-page* ini saling terhubung melalui *hyperlink* yang dapat diakses melalui perangkat lunak bernama browser. Browser merupakan aplikasi yang mampu menginterpretasikan dokumen-dokumen web sehingga pengguna dapat menjelajahi dan mengakses konten-konten tersebut [14].

#### 2.2.4. *Javascript*

JavaScript dapat didefinisikan sebagai bahasa pemrograman atau skrip kode yang dapat disematkan bersama dengan HTML atau dapat berdiri sendiri. Fungsinya adalah untuk memberikan interaktivitas dan membuat tampilan *website* menjadi lebih dinamis dengan menanggapi aksi pengguna dan memanipulasi elemen HTML pada *page* web. Bahasa pemrograman JavaScript diperkenalkan dan dikembangkan oleh Brendan Eich, seorang anggota tim di Netscape, pada bulan September 1995. Javascript awalnya dinamakan Moccha, lalu berganti menjadi Mona, lalu menjadi Livescript, setelah itu berganti nama lagi menjadi Javascript yang kita kenal sampai sekarang pada tahun 2006. Bahasa pemrograman ini (javascript) sudah mengalami perubahan yang begitu signifikan. Pada awalnya javascript begitu terbatas, lalu menjadi Bahasa yang sangat populer di kalangan web developer. Javascript adalah Bahasa

pemrograman yang dipakai sebagai tool untuk membuat sebuah *website* yang interaktif serta dinamis pada pengembangan web. [15].

#### 2.2.5. *ReactJS*

ReactJS merupakan perpustakaan (*library*) front-end yang dikembangkan oleh Facebook, yang menonjol dengan keunggulan-keunggulan seperti meningkatkan kecepatan, sederhana dalam penggunaan (*simplicity*), dan skalabilitas. ReactJS menyediakan peluang kepada para developer untuk membuat komponen interaktif, ReactJS juga mempunyai fitur yang dinamakan state yang dapat digunakan kembali [16]. ReactJS adalah sebuah pustaka JavaScript yang populer untuk pengembangan antarmuka pengguna (UI) yang interaktif dan dapat disesuaikan. Pendekatan utama dalam ReactJS adalah dengan membagi antarmuka pengguna menjadi komponen-*reusable* yang mandiri, yang dapat dikelola secara efisien. ReactJS menggunakan model pemrograman yang disebut "Virtual DOM" yang memungkinkan manipulasi cepat dan efisien terhadap perubahan pada *UI*. ReactJS juga memanfaatkan JSX, yaitu gabungan antara JavaScript dan HTML, untuk membangun komponen *UI* dengan sintaks yang terlihat mirip dengan HTML. Dengan menggunakan konsep ini, ReactJS menyediakan pengembangan yang modular, mudah dipahami, dan memungkinkan pengembang untuk membangun aplikasi web yang responsif dan efisien.

#### 2.2.6. *Tailwind-CSS*

Tailwind CSS merupakan suatu *framework* CSS yang memperkenalkan pendekatan baru dengan konsep "*utility first*" di mana kita tidak perlu lagi khawatir tentang penggunaan nama kelas yang panjang untuk elemen HTML. Tailwind CSS membawa pendekatan yang berbeda dibandingkan dengan *framework* CSS tradisional seperti Bootstrap, Foundation, atau Bulma. Jika *framework* lainnya menyediakan komponen yang utuh dan tinggal kita gunakan, beda dengan Tailwind CSS yang memberikan class class kecil yang bisa kita gunakan dengan sangat fleksibel untuk membangun antarmuka yang



sesuai keinginan tanpa harus *override styles* seperti di *framework* CSS lainnya [17].

#### 2.2.7. *Mobile First Design*

*Mobile-first design* adalah metode desain UI sebuah situs web di mana perancangan dimulai dengan fokus pada tampilan *mobile*. Setelah itu, dilakukan penyesuaian untuk tampilan *desktop*. [9].

#### 2.2.8. *Black Box Testing*

*Black-box testing* merupakan Teknik pengujian software yang berfokus pada fungsional dari software tersebut. Cara kerja *black box testing* ini adalah dengan mengabaikan struktur *internal* dari perangkat lunak, sehingga fokus utamanya adalah pada pemahaman dan pengujian fitur fungsional atau informasi domain yang dapat diakses dari luar. *Black-box testing* memungkinkan pengembang perangkat lunak untuk membuat kumpulan kondisi input yang akan menguji semua syarat fungsional suatu program tanpa memerhatikan implementasi *internal* atau struktur kode [18].

#### 2.2.9. *Usability*

*Usability* merupakan teknik evaluasi sebuah *website* yang dilakukan dengan pertimbangan beberapa aspek seperti "*learnability*" (kemampuan pengguna untuk mempelajari dan menggunakan sistem), "*efficiency*" (efisiensi dalam penggunaan), "*memorability*" (kemampuan untuk diingat dan digunakan kembali), "*errors*" (tingkat kesalahan yang terjadi), dan "*satisfaction*" (kepuasan pengguna). *Learnability* adalah evaluasi seberapa mudah pengguna dapat menguasai penggunaan *website*, *efficiency* mengukur seberapa cepat *website* memberikan informasi, *memorability* menilai sejauh mana pengguna dapat mengingat dan menggunakan kembali *website*, *errors* mengevaluasi cara *website* menangani kesalahan pengguna, dan *satisfaction* melibatkan penilaian terhadap tingkat kepuasan pengguna saat menggunakan *website*. [10].

### 2.2.9.1. *System Usability Scale (SUS)*

Metode SUS begitu populer dalam pengujian *usability*. Sistem *Usability (SUS)* diteliti dan dipertajam oleh John Brooke di sekitar tahun 1986. SUS (*Sistem Usability*) adalah alat pengukuran skala *usability* yang terpercaya, populer, efektif, dan terjangkau. Uji ketergantungan (*usability testing*) menggunakan SUS Dalam rangka mengukur efisiensi suatu sistem, sekaligus menilai sejauh mana kemudahan proses pembelajaran bagi pengguna, serta kemampuan mereka untuk menyimpan dan mengingat cara berinteraksi tanpa menghadapi kesulitan atau membuat kesalahan. Identifikasi masalah desain sistem dapat dilakukan dengan cukup mengandalkan lima orang pengguna dalam pengujian tersebut. Pengujian dengan lima orang dapat memberikan kemungkinan untuk menemukan lebih dari 80% masalah *usability* yang ada. Jika menggunakan jumlah peserta tes yang lebih banyak, peningkatan signifikan yang diperoleh mungkin hanya sekitar 20%.[19]. Bentuk pertanyaan dari pengujian SUS adalah sebagai berikut:

**Tabel 2. 3** Pertanyaan Kuesioner SUS

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

9.	Saya merasa tidak ada hambatan dalam menggunakan sistem ini.
10.	Saya perlu membiasakan diri terlebih dahulu sebelum menggunakan sistem ini.

Menggunakan metode SUS, data yang digunakan terdiri dari 10 pernyataan SUS yang dinilai menggunakan skala likert. Berikut adalah tabel dari skala likert yang digunakan:

**Tabel 2. 4** Skala Likert Penilaian SUS

Skala Likert	Keterangan
1	Sangat Tidak Setuju
2	Tidak Setuju
3	Ragu-Ragu
4	Setuju
5	Sangat Setuju

#### 2.2.9.2. *Usability Metrics for User Experience - Lite (UMUX-Lite)*

*Usability Metrics for User Experience- Lite (UMUX-Lite)* adalah versi singkat dari UMUX yang hanya terdiri dari dua pertanyaan [20]. *Usability Metric for User Experience Lite (UMUX-Lite)* adalah alat pengukuran kepuasan pengguna yang dirancang untuk mengevaluasi sejauh mana pengguna merasa puas dengan pengalaman penggunaan suatu sistem atau produk. *UMUX-Lite* terdiri dari pertanyaan yang fokus pada kemudahan penggunaan (*ease of use*) dan nilai penggunaan (*usefulness*). Skala penilaian yang digunakan berkisar pada tingkat setuju atau tidak setuju, membantu mendapatkan gambaran keseluruhan kepuasan pengguna secara singkat dan efisien. *UMUX-Lite* telah terbukti efektif dalam mengukur pengalaman pengguna dengan mempertimbangkan aspek keseluruhan, menjadikannya alat yang bermanfaat untuk evaluasi cepat

dalam konteks penelitian UX (User Experience) dan desain produk. Bentuk pertanyaan dari pengujian UMUX-*Lite* adalah sebagai berikut [20].

**Tabel 2. 5** Pertanyaan Kuesioner UMUX-*Lite*

No	Pertanyaan
1.	This <i>website/ product/ tool/ software/ prototype</i> capabilities meet my <i>requirements</i> .
2.	This <i>website/ product/ tool/ software/ prototype</i> is easy to use

Menggunakan metode UMUX-*Lite*, data yang digunakan terdiri dari 2 pernyataan dinilai menggunakan skala likert. Berikut adalah tabel dari skala likert yang digunakan:

**Tabel 2. 6** Skala Likert Penilaian UMUX-*Lite*

Skala Likert	Keterangan
1	Sangat Tidak Setuju
2	Tidak Setuju
3	Cukup Tidak Setuju
4	Netral
5	Cukup Setuju
6	Setuju
7	Sangat Setuju

#### 2.2.10. UML

UML (Unified Modeling Language) adalah tools yang sangat bermanfaat sebagai alat bantu dalam konteks *development* sistem. UML sudah menjadi patokan untuk perancangan dan visualisasi serta dokumentasi untuk *software development* [12]. UML juga disebut bahasa standar yang digunakan untuk menggambarkan, merancang, dan mendokumentasikan *software system*. Dengan UML, developer dapat dengan gampang melakukan komunikasi, menggambarkan sistem secara visual, memfasilitasi analisis yang mendalam, perancangan yang sistematis, dan dokumentasi yang konsisten dari sistem

perangkat lunak yang kompleks. UML bermacam-macam jenisnya, seperti *Activity Diagram*, *Use-Case Diagram*, *Sequence Diagram*, *Communication Diagram*, *State Diagram*, *Timing Diagram*, dan *Interaction Overview Diagram*.

### 1. *Activity diagram*

*Diagram* aktivitas (*activity diagram*) adalah representasi grafis yang digunakan dalam pemodelan sistem untuk menggambarkan aliran kontrol, aksi, dan aktivitas terstruktur secara visual [21]. *Diagram* ini memberikan pandangan yang jelas tentang bagaimana suatu proses atau fungsi dalam sistem berinteraksi, termasuk urutan langkah-langkah, keputusan, dan kondisi yang mungkin terjadi. Dengan menggunakan simbol-simbol seperti aktivitas, garis penghubung, dan garis pengendali, *diagram* aktivitas membantu pengembang dan analis sistem untuk merancang, memahami, dan mengkomunikasikan proses secara efektif, sehingga memfasilitasi pengembangan sistem yang terorganisir dan efisien.




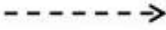
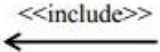
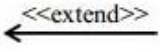
### 2. *Sequence Diagram*

*Diagram* urutan (*sequence diagram*) adalah jenis *diagram* interaksi dalam pemodelan sistem yang menggambarkan kolaborasi antar objek atau elemen dalam suatu kelas [21]. *Diagram* ini menunjukkan urutan pesan atau panggilan metode yang dikirim antar objek-objek tersebut selama suatu interaksi atau proses tertentu. Dengan menggunakan baris waktu vertikal untuk mewakili objek dan panah untuk merepresentasikan pesan atau panggilan metode, *diagram* urutan memberikan pemahaman yang jelas tentang bagaimana objek-objek saling berinteraksi dan berkomunikasi selama eksekusi suatu proses atau fungsionalitas dalam sistem. Hal ini sangat berguna dalam merancang dan memahami alur kerja yang terjadi di dalam sistem berbasis objek.

### 3. *Use case Diagram*

*Use case* merupakan tahap awal dalam memodelkan sistem, digunakan untuk merepresentasikan kebutuhan fungsional. Masing-masing *use case*



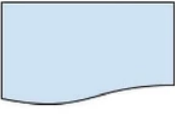
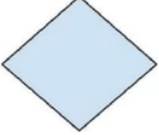

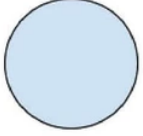


adalah gambaran singkat dari situasi di mana aktor terlibat dalam batas sistem yang telah ditetapkan. Hubungan antar *use case* diindikasikan dengan garis notasi. [22]. *Use case diagram* membantu dalam pemodelan fungsionalitas sistem dengan mengidentifikasi aktor yang terlibat, tindakan atau proses yang dijalankan oleh *role* yang disebut aktor, serta korelasi antara *role* aktor dan *use case* yang menjelaskan interaksi sistem dengan pengguna. *Use case diagram* memberikan gambaran yang jelas tentang kebutuhan pengguna, fungsi yang diharapkan, dan batasan sistem yang sedang dikembangkan, sehingga mempermudah pemahaman dan komunikasi antara pemangku kepentingan dalam pengembangan perangkat lunak.

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

**Gambar 2. 1** Simbol-simbol pada *UseCase Diagram* [23]

### 2.2.11. *Flowchart*

*Flowchart* adalah proses penyusunan langkah-langkah pemecahan masalah yang dapat diungkapkan melalui tulisan atau simbol-simbol khusus [24]. *Flowchart* adalah representasi visual dari urutan langkah-langkah atau alur logika dalam suatu proses. *Flowchart* digunakan untuk menggambarkan pemrosesan informasi, pengambilan keputusan, atau algoritma dengan menggunakan simbol-simbol grafis seperti kotak, panah, dan berlian. Setiap simbol dalam *flowchart* mewakili tindakan atau keputusan tertentu, dan panah menghubungkan simbol-simbol tersebut untuk menunjukkan aliran urutan yang tepat. *Flowchart* membantu dalam pemahaman dan komunikasi yang jelas mengenai langkah-langkah suatu proses, memperjelas alur logika, mengidentifikasi masalah, serta membantu dalam perancangan, analisis, dan dokumentasi sistem perangkat lunak atau proses bisnis.

No.	Simbol Flowchart	Nama	Arti Simbol Flowchart
1		<i>Terminator</i>	Awal atau akhir konsep (prosedur)
2		<i>Process</i>	Proses operasional
3		<i>Document</i>	Dokumen atau laporan berupa <i>print out</i>
4		<i>Decision</i>	Keputusan atau sub-point. Garis yang terhubung dengan bentuk <i>decision</i> merujuk pada situasi-situasi yang berbeda sesuai dengan keputusan yang digambarkan
5		Data	Input dan Output (Contohnya, Input: feedback dari pelanggan. Output: desain produk baru)
6		<i>On-Page Reference/Connector</i>	Penghubung alur dalam halaman yang sama
7		<i>Off-Page Reference/Off-Page Connector</i>	Penghubung alur dalam halaman yang berbeda
8		<i>Flow</i>	Arah alur dalam konsep (prosedur)

**Gambar 2. 2** Simbol-simbol pada Flowchart [25]