

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Tinjauan Pustaka**

Pada penelitian ini telah dilakukan studi literatur terhadap beberapa jurnal penelitian sebelumnya. Jurnal tersebut dipilih berdasarkan topik yang berkaitan dengan tema penelitian yaitu mengenai *front-end website* PT. Alfa Omega Network dengan metode perancangan *design thinking*. Berikut penjelasan lebih lanjut.

Penelitian menurut Destiara Kirana Safitri dan Andrianingsih pada tahun 2022 yang berjudul “*Analisis UI/UX untuk Perancangan Ulang Front-End Web Smart-SITA dengan Metode UCD dan UEQ*” membahas tentang Smart-SITA yang merupakan sebuah aplikasi berbasis web untuk membantu mahasiswa FTKI UNAS melakukan proses pengerjaan Tugas Akhir secara online. Penelitian ini memiliki tujuan, yaitu, menganalisis UI/UX dari Smart-SITA dan membuat desain solusi UI/UX nya untuk diterapkan dalam perancangan ulang *Front-End Web Smart-SITA* dengan menggunakan metode *User Centered Design (UCD)*. Penelitian ini juga menggunakan UEQ untuk evaluasi sistemnya, metode tersebut dapat menghasilkan penilaian dari user meliputi kesan pengalaman user yang menyeluruh. Untuk testing kualitas halaman *Front-End Smart-SITA* nilai rata-rata *testing* berdasarkan *Performances* mendapat kategori sangat baik, *Accessibility* rata-rata, dan *Best Practice* sangat baik [9].

Pada penelitian menurut Hananda Ilham, Bangun Wijayanto, Swahesti Puspita Rahayu pada tahun 2020 yang berjudul “*Analysis And Design Of User Interface/User Experience With The Design Thinking Method In The Academic Information System Of Jenderal Soedirman University*” membahas tentang desain Sistem Informasi Akademik (SIA) yang dirasa kurang ramah pengguna, akan berdampak kepada pengguna itu sendiri dan juga sistemnya. Tujuan dilakukannya analisis dan perancangan UI/UX pada SIA Universitas Jenderal Soedirman adalah untuk mengatasi

permasalahan yang dialami pengguna saat ini. Perancangan yang dibuat menggunakan metode *Design Thinking* dan pengujian menggunakan *Usability Testing* [10].

Pada penelitian menurut Nurul Citra Chergia Sidik pada tahun 2020 yang berjudul “*Pengembangan Front-End Website User HiDepok pada Proyek 1000 Aplikasi Smart City Kota Depok di Diskominfo Kota Depok*” membahas tentang pengembangan aplikasi Hi-Depok sebagai sarana informasi publik yang mendukung Depok *Smart City*. Pengembangan aplikasi ini hanya difokuskan pada bagian *front-end website user* yang menggunakan Laravel untuk mempermudah penghubungan antara *front-end* dan *back-end* dalam satu aplikasi dengan memanfaatkan pola *Model-View-Controller* (MVC). Dalam pengembangan front-end website user, diterapkan metodologi *User Centered Design* (UCD) yang memiliki beberapa tahapan guna memastikan tingkat *usability* produk yang dihasilkan tinggi. Oleh karena itu, penelitian ini dapat memberikan hasil yang baik bagi pengguna aplikasi hingga tahap pengujian. Tampilan *website user* Hi-Depok memiliki kriteria *user friendly* sehingga memudahkan pengguna untuk mengakses aplikasi. Saat ini aplikasi Hi-Depok dapat diakses melalui <http://hi.depok.go.id/> secara umum [11].

Pada penelitian menurut Muchamad Aqiel Hilman Maulandany, Dana Sulisty Kusumo, Jati Hiliamsyah Husen pada tahun 2021 yang berjudul “*Integrasi Scrum Dengan Design Thinking Dalam Pengembangan Perangkat Lunak Pada Startup Awal (studi Kasus Kioos)*” dalam penelitian ini telah dilakukan penggabungan metode *Design Thinking* dan *Scrum*. *Design Thinking* digunakan untuk menciptakan solusi produk yang diinginkan oleh pengguna dan *Scrum* digunakan untuk proses pembuatan perangkat lunak untuk produk tersebut. Hasil pengujian *prototype* menghasilkan nilai SUS rata-rata sebesar 70.625 yang menandakan *prototype* dapat diterima oleh market. Hasil pengujian produk dengan *Framework Pirate Funnel* selama 28 hari memperlihatkan jika produk yang dihasilkan belum dapat dikatakan *product-market fit* dikarenakan

persentase *Retention* dari *Acquisition* hanya sebesar 19.4%. Namun, produk tersebut memiliki potensi untuk mencapai *product- market fit* dikarenakan terdapat peningkatan pada *activation* dan *retention*. Sehingga penelitian ini membuktikan bahwa integrasi *Design Thinking* dan *Scrum* dapat dilakukan pada *Early-Stage Startup* untuk menciptakan produk yang dibutuhkan oleh market namun proses ini memerlukan penyesuaian pada tim [12].

Pada penelitian menurut Gusti Karnawan, Septi Andryana, Ratih Titi Komalasari pada tahun 2021 yang berjudul “*Implementasi User Experience Menggunakan Metode Design Thinking Pada Prototype Aplikasi Cleanstic*” dalam penelitian ini telah dibuat *prototype* berbasis android untuk mengedukasi masyarakat tentang pengelolaan sampah plastik ataupun menyumbangkan sampah plastiknya kepada yang membutuhkan. Metode yang digunakan dalam penelitian ini adalah *design thinking* mulai dari awal perancangan sampai akhir, sehingga menghasilkan tingkat *user experience* yang baik. Dari hasil evaluasi yang didapat sangat baik dari pengujian desain solusi *cleanstic* yang telah diujicobakan, semua aspek mulai dari tingkat *Attractiveness*, *Perspiciuity*, *Efficiency*, *Dependability*, *Stimulation* dan *Novelty* mendapatkan nilai yang sama, yaitu *excellent* [13].

Dari penjelasan diatas, ringkasan penelitian yang relevan ditunjukkan pada Tabel 2.1 dibawah ini:

Tabel 2. 1 Penelitian Terdahulu

No.	Peneliti	Judul	Metode	Hasil	Revelansi	Kebaruan/Inovasi	Simpulan
1.	Destiara Kirana Safitri dan Andrianingsih [9]	Analisis UI/UX untuk Perancangan Ulang <i>Front-End</i> <i>Web Smart-SITA</i> dengan Metode UCD dan UEQ	<i>User</i> <i>Centered</i> <i>Design</i> (UCD)	Perancangan <i>Front-End</i> <i>Web Smart-SITA</i> berhasil di implementasikan dari sisi <i>user</i> mahasiswa. Serta mendapat kategori sangat baik, <i>Accessibility</i> rata-rata, dan <i>Best</i> <i>Practice</i> sangat baik.	Topik pada penelitian yang digunakan memiliki relevansi tentang <i>front-end</i> <i>website</i> .	Pada penelitian yang dilakukan peneliti mengukur tingkat kepuasan pengguna dan hasil rekomendasi <i>user</i> <i>experience</i> pada <i>website</i>	Penelitian digunakan untuk perancangan <i>front- end website</i> proses bimbingan sedangkan pada penelitian ini digunakan untuk perancangan <i>website</i> layanan internet
2.	Hananda Ilham, Bangun Wijayanto, Swahesti	<i>Analysis And</i> <i>Design Of User</i> <i>Interface/User</i> <i>Experience With</i> <i>The Design</i>	<i>Design</i> <i>thinking</i>	Perancangan desain UI/UX pada SIA Unsoed dirasa mampu membantu dalam menghasilkan	Topik pada penelitian yang digunakan memiliki	Pada penelitian yang dilakukan peneliti mengukur tingkat kepuasan pengguna dan hasil	Penelitian ini menggunakan metode yang sama yaitu <i>design</i> <i>thinking</i> namun

	Puspita Rahayu [10]	<i>Thinking Method In The Academic Information System Of Jenderal Soedirman University</i>		desain <i>prototype</i> produk yang sesuai dengan kebutuhan mahasiswa.	relevansi tentang <i>design thinking</i>	rekomendasi <i>user experience</i> pada <i>website</i>	menggunakan teknik pengujian yang berbeda yaitu <i>system usability scale</i> (SUS)
3.	Nurul Citra Chergia Sidik [11]	Pengembangan <i>Front-End Website User</i> HiDepok pada Proyek 1000 Aplikasi <i>Smart City</i> Kota Depok di Diskominfo Kota Depok	<i>User Centered Design</i> (UCD)	Telah dikembangkan <i>website user</i> Hi-Depok meskipun belum sampai tahap final	Topik pada penelitian yang digunakan memiliki relevansi tentang <i>front-end website</i> .	Pada penelitian yang dilakukan peneliti mengukur tingkat kepuasan pengguna dan hasil rekomendasi <i>user experience</i> pada <i>website</i>	Penelitian ini menggunakan metode yang sama yaitu <i>design thinking</i> namun menggunakan teknik pengujian yang berbeda yaitu <i>system usability scale</i> (SUS)
4.	Muchamad Aqiel Hilman	Integrasi Scrum Dengan <i>Design</i>	<i>Design thinking</i>	Telah menghasilkan produk yang	Topik pada penelitian yang	Pada penelitian yang dilakukan	Penelitian ini menggunakan

	Maulandany, Dana Sulistyio Kusumo, Jati Hiliasmasyah Husen [12]	<i>Thinking</i> Dalam Pengembangan Perangkat Lunak Pada Startup Awal (studi Kasus Kioos)		dibutuhkan oleh <i>market</i>	digunakan memiliki relevansi tentang <i>design thinking</i>	peneliti mengukur tingkat kepuasan pengguna dan hasil rekomendasi <i>user experience</i> pada <i>website</i>	metode yang sama yaitu <i>design thinking</i> namun menggunakan teknik pengujian yang berbeda yaitu <i>system usability scale</i> (SUS)
5.	Gusti Karnawan, Septi Andryana, Ratih Titi Komalasari [13]	Implementasi <i>User Experience</i> Menggunakan Metode <i>Design Thinking</i> Pada <i>Prototype</i> Aplikasi Cleanstic	<i>Design thinking</i>	Hasil evaluasi yang didapat sangat baik dari pengujian desain solusi cleanstic yang telah diujicobakan	Topik pada penelitian yang digunakan memiliki relevansi tentang <i>design thinking</i>	Pada penelitian yang dilakukan peneliti mengukur tingkat kepuasan pengguna dan hasil rekomendasi <i>user experience</i> pada <i>website</i>	Penelitian ini menggunakan metode yang sama yaitu <i>design thinking</i> namun menggunakan teknik pengujian yang berbeda yaitu <i>system usability scale</i> (SUS)

## 1.2 Dasar Teori

Berikut adalah kajian mengenai beberapa teori yang digunakan pada penelitian ini:

### 1.2.1 *Front-end*

*Front-end* adalah sebuah bagian dari sistem yang menyediakan tampilan kepada pengguna. Bertugas untuk mengembangkan komponen-komponen visual pada suatu sistem dan bertanggung jawab pada tampilan antarmuka [16].

*Front-end website* adalah gabungan dari tiga bahasa pemrograman, yaitu *Hypertext Markup Language* (HTML), *Cascading Style Sheets* (CSS), dan *JavaScript.*, yang digunakan untuk menciptakan halaman *website* yang dapat diakses dan berinteraksi langsung oleh pengguna. Tujuan dari *front-end* adalah memudahkan pengguna dalam memahami informasi yang disajikan [17].

*Front-end* sendiri adalah sebuah bagian dari sistem yang menyediakan tampilan kepada pengguna. Bertugas untuk mengembangkan komponen-komponen visual pada suatu sistem dan bertanggung jawab pada tampilan antarmuka [18].

### 1.2.2 *Website*

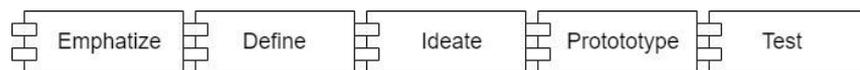
*Website* adalah sekumpulan halaman web yang terhubung satu sama lain dan saling berkaitan melalui file-file yang terkait. Web terdiri dari halaman-halaman, yang tergabung dalam *homepage* dan disusun berurutan dari posisi teratas hingga ke bawah. *Homepage* merupakan halaman utama dan posisinya paling atas dalam susunan, sementara halaman-halaman terkait ditempatkan di bawahnya. Setiap halaman di bawah *homepage* biasanya disebut sebagai *child page*, yang berisi tautan atau *hyperlink* ke halaman-halaman lain dalam web [15].

### 1.2.3 PT. Alfa Omega Network

Perseroan Terbatas (PT) Alfa Omega Network merupakan salah satu perusahaan yang bergerak dibidang jasa provider internet yang menyediakan akses internet dan perangkat jaringan internetnya kepada pengguna atau *client* yang berpusat di Cibitung, Jawa Barat(Lampiran 1).

### 1.2.4 *Design Thinking*

*Design Thinking* adalah metodologi yang bertujuan untuk memecahkan masalah kompleks dengan menerapkan pengetahuan Desain. Pada prosesnya *Design Thinking* memiliki lima langkah tahapan yang berbeda-beda dalam mencapai tujuannya, yaitu [19]:



Gambar 2. 1 Tahapan Design Thinking

#### 1. *Emphatize* (Empati)

*Emphatize* berfokus pada pemahaman tentang pikiran, keinginan, dan kebutuhan pengguna. Untuk mengetahui *user* atau pengguna yang dituju dilakukan sebuah observasi pengguna, waawancara, kuisisioner untuk memahami pengguna. Langkah ini diperlukan karena masalah yang diselesaikan adalah masalah pengguna, sehingga penting untuk memahaminya dari sudut pandang pengguna.

#### 2. *Define* (Penetapan)

Pada tahap *Define* permasalahan yang disebut *point of view* mengarahkan kepada sebuah *insight* terhadap kebutuhan user yang didapatkan dari proses menganalisa informasi [20].

##### a. *User Persona*

Dalam *user persona* memudahkan untuk melakukan analisis target *user* untuk mengetahui *user goals*, perilaku *user*, dan *pain point* dari pengguna.

b. *How Might We*

*How Might We* (HWM) merupakan salah satu cara yang dilakukan untuk mengubah permasalahan yang dihadapi menjadi tantangan untuk diselesaikan, *How* dan *might* memiliki penjelasan yaitu *how* adalah sebuah pertanyaan yang dibuat dari suatu permasalahan, dan *might* adalah jawaban daari pertanyaannya.

c. *Impact Effort*

Setelah membuat HMW, proses selanjutnya yang dilakukan adalah membuat *impact effort*. *Impaact Effort* berfungsi untuk mengukur permasalahan yang memiliki *impact* yang paling besar bagi *user* dan *effort* yang kemungkinan dapat dapat dikerjakan pada penelitian ini.

3. *Ideate* (Ide)

Tahap *Ideate* berisi ide-ide untuk menemukan solusi dari permasalahan yang telah didapatkan.

a. *User Flow*

*User flow* sendiri merupakan representasi visual dari langkah-langkah atau urutan tindakan yang diambil oleh pengguna saat berinteraksi dengan sistem.

4. *Prototype* (Prototipe)

*Prototype* adalah tahap dalam proses di mana rancangan tampilan website yang direncanakan dibuat dan diimplementasikan untuk menghasilkan suatu prototype atau produk yang siap untuk diuji [21].

a. *Wireframe*

*Wireframe* merupakan representasi visual untuk penataan elemen item-item yang terdapat pada sebuah produk baik *website* maupun tampilan mobile. *Wireframe* sendiri terbagi dalam dua macam, yaitu :

1. *Low-Fidelity Wireframe*

Merupakan desain yang tingkat persepsi nya masih rendah dan tujuannya hanya untuk menunjukkan tata letak.

2. *High-Fidelity Wireframe*

Merupakan desain yang tingkat persesinya tinggi, sudah mencakup warna, ukuran, jarak dan bentuk elemen lainnya dengan kata lain lebih mendetail.

5. *Test*

Dilakukan pengujian dan evaluasi terhadap pengguna atau *user*. Tahap ini penting dilakukan karena akan didapatkan pengalaman dan masukan dari user untuk memperbaiki produk atau aplikasi yang sudah diimplementasikan.

## 2.2.5 Pengujian *Front-End Website*

Pengujian *Front-end website* penting dilakukan dalam perancangan sebuah tampilan website untuk dapat memastikan kualitasnya. Untuk pengujian yang akan dilakukan dalam penelitian ini adalah dengan menggunakan *metode System Usability Scale (SUS)* [22].

### 2.2.5.1 *System Usability Scale (SUS)*

*System Usability Scale (SUS)* adalah sebuah metode pengujian yang digunakan untuk mengukur tingkat *usability* sebuah sistem. Metode ini diperkenalkan oleh John Brooke pada tahun 1986, yang dapat digunakan untuk melakukan berbagai

jenis produk termasuk didalamnya perangkat website dan aplikasi [23].

Penggunaan SUS dalam melakukan pengujian lebih menekankan perspektif pengguna akhir sehingga hasil evaluasi akan lebih sesuai dengan keadaan nyata. Pengujian SUS memiliki 10 pernyataan sebagai alat pengujian. *System Usability Scale* memiliki daftar pertanyaan pada gambar berikut [24].

Tabel 2. 2 Daftar pertanyaan SUS

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

SUS berupa kuesioner yang terdiri dari 10 item pertanyaan seperti ditunjukkan pada Tabel 2.1 Kuesioner SUS menggunakan 5 poin skala Likert. Responden diminta untuk memberikan penilaian “Sangat Tidak Setuju (STS)”, “Tidak Setuju (TS)”, “Ragu-ragu (RG)”, “Setuju (S)”, dan “Sangat

setuju (SS)' atas 10 item pernyataan SUS sesuai dengan penilaian subyektifnya [25]. Untuk lebih jelasnya skala penelitian dapat dilihat pada Tabel 2.2.

Tabel 2. 3 Skala penelitian

No.	Jawaban	Skor
1.	Sangat Tidak Setuju (STS)	1
2.	Tidak Setuju (TS)	2
3.	Ragu-ragu (RG)	3
4.	Setuju (S)	4
5.	Sangat Setuju (SS)	5

Setelah melakukan pengisian kuesioner selanjutnya data tersebut dihitung sesuai aturan perhitungan skor dalam SUS. Ada beberapa aturan dalam perhitungan skor kuesioner yang berlaku untuk masing-masing responden, yaitu:

1. Setiap pertanyaan bernomor ganjil (1,3,5,7,9), skor setiap pertanyaan yang didapat dari skor pengguna akan dikurangi 1.
2. Setiap pertanyaan bernomor genap (2,4,6,8,10), skor akhir didapat dari nilai 5 dikurangi skor pertanyaan yang didapat dari pengguna.
3. Skor SUS didapat dari hasil penjumlahan skor setiap pertanyaan yang kemudian dikali 2,5.
4. Perhitungan rata-rata jawaban pertanyaan

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

Keterangan:

$\tilde{x}$  = Skor Rata-rata

$\sum x$  = Jumlah skor SUS

n = Jumlah Responden

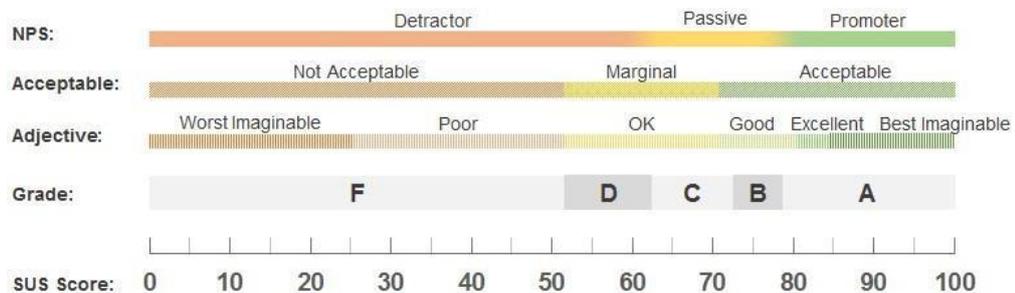
Penilaian SUS dapat memperjelas dalam proses perhitungannya, dapat dilihat pada persamaan sebagai berikut:

$$\begin{aligned} \text{Skor SUS} = & ((P1 - 1) + (5 - P2) + (P3 - 1) + (5 - P4) \\ & + (P5 - 1) + (5 - P6) + (P7 - 1) + (5 - P8) \\ & + (P9 - 1) + (5 - P10)) \times 2.5 \end{aligned} \quad (2,2)$$

Keterangan :

P = Pertanyaan SUS

Dalam pengukuran *System Usability Scale* (SUS) adapun kriteria adalah sebagai berikut:



Gambar 2. 2 Kriteria Penilaian SUS

Berdasarkan hasil perhitungan persamaan skor *System Usability Scale* (SUS) dapat ditarik kesimpulan bahwa ada 5 tingkatan pada hasil akhir menurut Jhon Brooke yang tertera pada Tabel 2.3.

Tabel 2. 4 Kriteria penelitian SUS

SUS	Tingkatan	Kriteria
Lebih dari 81	A	Excellent
74 -81	B	Good
68-73	C	Ok atau fair
51-67	D	Poor
Kurang dari 51	F	Worst

Pada Gambar 2.2 dan Tabel 2.4 mengenai kriteria penilaian *System Usability Scale* (SUS) dijelaskan bahwa tingkatan nilai

>80,3 mendapatkan kriteria sangat bagus, sedangkan tingkatan nilai terendah < 51 mendapatkan kriteria sangat kurang [26].

### 2.2.6 *ReactJs*

*ReactJs* adalah *library JavaScript* yang digunakan untuk mengembangkan komponen antarmuka pengguna (UI) yang dapat digunakan kembali. *React* pada dasarnya memungkinkan pengembang aplikasi berbasis web besar dan kompleks yang dapat mengubah datanya tanpa *refresh* halaman berikutnya. *React* mengabstraksi *Document Object Model (DOM)*, sehingga menawarkan pengalaman pengembangan aplikasi yang sederhana, berkinerja, kuat dan efisien [27].

*React* sendiri merupakan salah satu pustaka *JavaScript* paling populer yang mengalahkan pustaka *JavaScript* lain seperti *Angular* dan *Vue*. *React* juga termasuk pustaka *JavaScript* yang paling mudah untuk dipelajari. Secara kinerja, *React* juga mendapatkan hasil pengujian kinerja yang bagus dari pengujian yang telah dilakukan antara tiga pustaka, yaitu *React*, *Angular*, dan *Vue*. Peringkat pertama didapatkan oleh *Vue*, kemudian disusul oleh *React*, dan yang terakhir adalah *Angular* [28].

*React* memungkinkan pengembangan antarmuka pengguna yang kompleks dengan kode yang kecil dan terisolasi yang disebut "komponen". Penggunaan *React* mempermudah proses pembuatan antarmuka pengguna yang interaktif. Sintaksis *React* yang deklaratif mempermudah pemahaman dan memudahkan proses debug. *React* berbasis komponen, yang berarti dapat membuat beberapa komponen terenkapsulasi yang mengatur state mereka sendiri, kemudian menggabungkan komponen-komponen ini untuk membentuk antarmuka pengguna yang lebih kompleks. Pendekatan ini memudahkan pemeliharaan, karena jika terjadi masalah, pengembang dapat langsung fokus pada komponen yang bermasalah tanpa mengganggu komponen lainnya, yang menghemat waktu dan mempercepat proses

pemeliharaan. Selain itu, komponen-komponen yang telah ditulis dapat digunakan kembali, menghemat waktu dan membuat kode lebih efisien, karena tidak perlu menulis ulang kode yang serupa secara berulang-ulang [29].

Di dalam *React* juga terdapat JSX yang merupakan sintak seperti XML atau HTML yang digunakan oleh *React* untuk memperluas ECMAScript, sehingga teks seperti XML atau HTML dapat digunakan berdampingan dengan kode *JavaScript* atau *React*. *React* memiliki logika rendering secara inheren yang digabungkan dengan logika UI lainnya, seperti bagaimana event ditangani, bagaimana status dapat berubah dari waktu ke waktu, dan bagaimana data dapat disiapkan untuk ditampilkan. JSX direkomendasikan untuk digunakan bersama dengan *React* dengan tujuan untuk menjelaskan bagaimana tampilan UI, sehingga memungkinkan *React* untuk menampilkan pesan dan peringatan yang lebih berguna bagi pengembang. JSX memudahkan pengembang karena kode antara markup (HTML) dapat digabungkan langsung dengan logika (*JavaScript*), sehingga penulisan sintaksisnya lebih sederhana dan mudah daripada penulisan keduanya secara terpisah [29].