

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1. Tinjauan Pustaka

Penulis mencoba menghubungkan penelitian ini dengan beberapa penelitian terdahulu atau literatur-literatur ilmiah yang sudah ada, sehingga dapat mengetahui dan memahami setiap teori yang berkaitan dengan studi kasus maupun metode-metode pada penelitian karya ilmiah ini.

Sebuah penelitian yang menggunakan metode *User-Centered Design* pernah dilakukan untuk merancang desain pada aplikasi pengelolaan *laundry* di Purwokerto yang mengalami kendala dalam mengelola data transaksi *laundry* yang masih menggunakan media fisik. Peneliti ini menghasilkan skor *usability testing* sebesar 79,8% dan pada pengujian *blackbox* menghasilkan skenario uji yang secara keseluruhan dapat berjalan dengan baik. Penelitian ini menjelaskan bahwa tahapan yang ada pada metode *User-Centered Design* membantu peneliti dalam membuat sistem yang baik, terutama pada tahap proses dan hasil analisis kebutuhan *user* sangat penting karena dapat mempengaruhi hasil desain yang layak atau tidaknya pada sistem tersebut [6].

Selain itu penelitian lain yang juga menggunakan metode *User-Centered Design*, namun memiliki studi kasus yang berbeda, yaitu dengan melakukan evaluasi dan perbaikan *user interface* pada *website* Awake Project untuk memperbaiki desain *website* yang masih kurang memuaskan dan penambahan fitur *menu profile*, pengujian pada penelitian ini menghasilkan jumlah total skor SUS sebesar 1672.5 dari yang sebelum dilakukan evaluasi pada desain hanya mendapatkan skor sebesar 820. Jumlah tersebut jika rata-rata orang mendapatkan nilai skor SUS sebesar 83.62 atau jika dilihat dari skor *Percentile Rank* mendapatkan *Grade A* berdasarkan pengujian pada 20 orang responden [7].

Lain halnya dengan penelitian yang dilakukan menggunakan metode *heuristic evaluation* pada proses pengujiannya. Penelitian ini membahas tentang rancang bangun *prototype* pada sistem informasi program studi teknik informatika universitas muhammadiyah malang untuk menampilkan kegiatan yang berhubungan dengan akademik. Hasil pengujian yang didapatkan memiliki nilai rata-rata sebesar 1,34 atau masuk dalam skala 1, kategori *cosmetic problem*, yaitu masalah tidak perlu diperbaiki kecuali ada waktu tersisa dalam pengerjaan proyek [8]. Masalah *usability* paling besar terdapat pada aspek *help and documentation* yang mendapatkan nilai *severity rating* 2,40, sedangkan nilai terendah terdapat pada aspek *visibility of system status* dengan nilai *severity rating* 1,01. Sementara itu, terdapat penelitian mengenai pengembangan *e-commerce* pada toko Falona Fashion untuk membantu mengelola toko dan melakukan proses transaksi pembelian secara online antara pengelola toko, pelanggan dan reseller. Desain yang telah dibuat menggunakan metode *User-Centered Design* dilakukan proses pengujian menggunakan metode *black box* dengan total terdapat 12 task yang diujikan berhasil dieksekusi 100% dengan baik, sehingga desain sistem dapat dikatakan siap untuk diimplementasikan ke tahap lebih lanjut [9].

Terdapat peneliti yang melakukan pengujian menggunakan 3 metode yang berbeda, tentang pengembangan desain *frontend* pada sistem informasi perpustakaan Politeknik Negeri Jakarta yang berbasis *website* menggunakan metode *User-Centered Design* untuk memudahkan *civitas* akademika maupun kalangan umum dalam mendapatkan informasi dan meningkatkan pelayanan kepada pengunjung perpustakaan [10]. Hasil pengujian dengan menggunakan metode *system usability scale* didapatkan skor sebesar 83, skor tersebut masuk dalam kategori *acceptable*, dengan *grade scale* masuk dalam kategori B dan *adjective ratings* masuk dalam kategori *excellent*. Pengujian yang menggunakan metode *user experience questionnaire* mendapatkan penilaian yang *positif*, yaitu *attractiveness* (daya tarik) mendapatkan nilai 1,86, *pragmatic quality* (kualitas pragmatis) sebesar 1,81, dan 1,47 untuk

hedonic quality (kualitas hedonis). Sementara itu pengujian dengan menggunakan metode *black box* berjalan sesuai dengan skenario yang ada.

Metode *User-Centered Design* bukan satu-satunya metode untuk mendesain sebuah sistem perangkat lunak, terdapat beberapa metode lain yang digunakan dalam mendesain, seperti penelitian yang menggunakan metode *human centered design* dalam perancangan *website laundry* untuk memudahkan pendataan data transaksi dalam membuat laporan, mencetak struk transaksi dan menyimpan data ke dalam *database* sehingga dapat meningkatkan pelayanan pada bisnis. Penelitian ini menghasilkan rata-rata nilai skala 6 yaitu sebesar 1,401 pada 40 responden menggunakan metode pengujian *user experience questionnaire*. Dengan hasil tersebut maka desain *website* siap untuk dikembangkan ketahap lebih lanjut [11]. Sementara itu, ada penelitian yang menggunakan metode *goal-directed design* untuk melakukan evaluasi dan perbaikan desain pada *website* Jawa Timur Park Group yang memiliki kekurangan seperti *user interface* yang kurang memuaskan, desain terkesan ketinggalan zaman, beberapa informasi sulit ditemukan, tidak adanya menu navigasi, pemilihan *font* dan grafik yang kurang tepat, penempatan *layout* yang terkesan kurang menarik dan desain yang tidak *user friendly*. Metode ini menghasilkan perbaikan desain dengan nilai skor SUS sebesar 82,75 dari yang sebelum dilakukan perbaikan hanya mendapatkan skor SUS sebesar 46,5, yang berarti perbaikan ini menghasilkan *percentile rank* dengan *grade A* [12]. Selain itu terdapat metode *design sprint* yang dipakai pada penelitian tentang perancangan *user interface* berupa *prototype* aplikasi EzyPay untuk melakukan *update* pada desain yang masih kurang memuaskan dan menambahkan fitur saldo EzyPay agar dapat menunjang berbagai proses bisnis yang baru. Pengujian yang menggunakan metode *system usability scale* menghasilkan skor SUS paling besar yaitu 92,5 dan *time-based* yang dibutuhkan pada setiap *task* yaitu sebesar 0,038 atau 2/53 [13].

Sementara itu, terdapat penelitian yang melakukan pendekatan menggunakan metode *design thinking* dengan menghimpun karakteristik

penggunanya menggunakan *tool user persona* pada perancangan desain aplikasi penjualan sesuai kebutuhan bisnis. Pengujian menggunakan *usability testing* pada 35 halaman *prototype* dengan total 4 *user scenario* pengujian menghasilkan presentase keberhasilan hingga mencapai 100%. Penelitian ini menghasilkan *prototype* aplikasi penjualan berbasis *mobile* sesuai dengan target pengguna berdasarkan *user persona* dan kebutuhan bisnis [14]. Selain itu, penelitian serupa juga pernah dilakukan menggunakan metode *design thinking* [15]. Penelitian ini mengimplementasi *user experience* pada *prototype* aplikasi *cleanstic* untuk memberi pembelajaran kepada masyarakat tentang cara mengolah sampah plastik dan memudahkan bagi masyarakat dalam menjual ataupun memberikan sampah plastiknya kepada orang atau pihak yang sedang membutuhkan. Pengujian ini menghasilkan skor SUS sebesar 75.8 pada pengujian pertama dan pengujian kedua mendapatkan nilai skor SUS sebesar 78.8. Serta pengujian dengan metode *user experience questionnaire* menghasilkan skala rata-rata nilai yang dihasilkan yaitu *attractiveness* sebesar 2.197, *perspicuity* sebesar 2.275, *efficiency* sebesar 2.150, *dependability* sebesar 2.050, *stimulation* sebesar 2.383 dan *Novelty* sebesar 2.089.

Berdasarkan beberapa penelitian diatas dapat disimpulkan bahwa terdapat berbagai metode dalam mendesain sebuah sistem, seperti *design thinking* yang merupakan metode pemikiran yang digunakan seorang desainer dalam melakukan desain dengan pendekatan yang berpusat atau berpacu pada objek responden untuk pemecahan masalah [14], *human centered design* yang berorientasi pada pengguna sehingga perancang dapat mengetahui apa saja yang menjadi kebutuhan pengguna atau user [11], *goal-directed design* menggunakan pendekatan yang berpusat pada pengguna untuk mempersatukan tujuan yang berbeda-beda [12], dan *design sprint* yang merupakan metode untuk menyelesaikan masalah dengan melibatkan pengguna melalui perancangan, pembuatan *prototype*, dan pengujian ide dengan cepat [13].

Semua metode diatas memiliki fungsi dan kelebihan masing-masing, namun pada penelitian ini penulis akan menggunakan metode *user-centered design* untuk mendesain sebuah *website* berdasarkan pendekatan pengguna sebagai pusat dari proses pengembangan sebuah sistem [7], agar *website* yang dibuat dapat digunakan sesuai dengan kebutuhan masing-masing penggunanya.

Tabel 2.1 Daftar Penelitian Terdahulu

No	Peneliti	Deskripsi	Metode	Hasil
1	Pambudi dkk (2021)	Perancangan aplikasi pengelolaan <i>laundry</i> yang mengalami kendala dalam mengelola data transaksi <i>laundry</i> yang masih menggunakan media fisik.	<i>User-Centered Design</i>	1) Pengujian menggunakan metode <i>usability testing</i> diperoleh kelayakan sebesar 79,8%. 2) Pengujian <i>blackbox testing</i> secara keseluruhan menghasilkan skenario uji yang dapat berjalan dengan baik.
2	Gunawan dkk (2019)	Evaluasi dan perbaikan <i>user interface</i> pada <i>website Awake Project</i> yang digunakan untuk memberikan informasi harga produk dan cara pembelian produk kepada pelanggan. Perbaikan yang dilakukan bertujuan untuk memperbaiki desain <i>user interface website</i> yang masih kurang memuaskan dan penambahan fitur <i>menu Profile</i> . Proses dimulai dari tahap evaluasi awal, menentukan konteks pengguna, menentukan kebutuhan pengguna, menyempurnakan desain <i>user interface</i> hingga tahap evaluasi akhir.	– <i>User-Centered Design</i> – <i>Card Sorting</i>	Pada tahap pengujian awal sebelum dilakukan evaluasi dan perbaikan desain didapatkan jumlah skor sebesar 820 dari 20 orang responden dan rata-rata persatu orang mendapatkan skor SUS sebesar 41. Setelah dilakukan evaluasi dan perbaikan tampilan <i>website</i> menggunakan metode <i>User-Centered Design</i> didapatkan jumlah skor sebesar 1672.5 dari 20 orang responden yang sama dan rata-rata orang mendapatkan nilai skor SUS sebesar 83.62. Dapat disimpulkan bahwa <i>website Awake Project</i> mengalami peningkatan nilai rata-rata berdasarkan hasil pengujian SUS setelah dilakukan evaluasi dan perbaikan.
3	Rahman dkk (2020)	Rancang bangun <i>prototype</i> sistem informasi program studi teknik informatika universitas muhammadiyah malang untuk menampilkan kegiatan yang berhubungan dengan	<i>User-Centered Design</i>	Pengujian menggunakan metode <i>heuristic evaluation</i> secara keseluruhan dari semua aspek <i>usability</i> yang ada didapatkan nilai rata-rata sebesar 1,34 atau masuk dalam skala 1 yang berarti

No	Peneliti	Deskripsi	Metode	Hasil
		<p>akademik. Sistem ini dibuat untuk menggantikan <i>website</i> Teknik Informatika UMM yang jarang diakses oleh para mahasiswanya karena informasi yang kurang lengkap, tampilan yang monoton serta belum terintegrasi dengan situs utama Universitas Muhammadiyah Malang, Laboratorium Teknik Informatika dan <i>e-learning</i>.</p>		<p>termasuk kategori <i>cosmetic problem</i>, yaitu masalah tidak perlu diperbaiki kecuali ada waktu tersisa dalam pengerjaan proyek. Sementara itu terdapat masalah <i>usability</i> yang paling besar pada sub aspek <i>help and documentation</i> dengan nilai <i>severity rating</i> 2,40, sedangkan nilai terendah terdapat pada aspek <i>visibility of system status</i> dengan nilai <i>severity rating</i> 1,01.</p>
4	Widiati (2019)	<p>Pengembangan <i>e-commerce</i> pada salah satu toko <i>fashion</i> di Klaten bernama Falona <i>Fashion</i> untuk membantu mengelola toko dan melakukan proses transaksi pembelian secara <i>online</i> antara pengelola toko, pelanggan dan <i>reseller</i>.</p>	<i>User-Centered Design</i>	<p>Pengujian yang dilakukan dengan menggunakan <i>black box testing</i> dengan total terdapat 12 <i>task</i> yang diujikan, hasilnya semua <i>task</i> berhasil dieksekusi dengan baik dan siap untuk diimplementasikan.</p>
5	Prawastiyo dan Hermawan (2020)	<p>Pengembangan desain <i>frontend</i> pada sistem informasi perpustakaan Politeknik Negeri Jakarta berbasis <i>website</i> untuk memudahkan civitas akademika maupun kalangan umum dalam mendapatkan informasi dan meningkatkan pelayanan kepada pengunjung perpustakaan.</p>	<i>User-Centered Design</i>	<ol style="list-style-type: none"> 1) Hasil pengujian menggunakan metode SUS didapatkan skor sebesar 83, skor tersebut masuk dalam kategori <i>acceptable</i>, dengan <i>grade scale</i> masuk dalam kategori B dan <i>adjective ratings</i> masuk dalam kategori <i>excellent</i>. 2) Hasil pengujian <i>user experience questionnaire</i> didapatkan penilaian yang positif yaitu <i>attractiveness</i> (daya tarik) mendapatkan nilai 1,86, <i>pragmatic quality</i> (kualitas pragmatis) sebesar 1,81, dan 1,47 untuk <i>hedonic quality</i> (kualitas hedonis). 3) Sementara itu hasil pengujian <i>black box</i> berjalan sesuai dengan skenario.
6	Salsabilah dkk (2022)	<p>Analisa UI/UX perancangan <i>website laundry</i> untuk memudahkan pendataan</p>	<i>Human Centered Design</i>	<p>Pengujian menggunakan metode <i>user experience questionnaire</i> kepada 40 responden rata-rata</p>

No	Peneliti	Deskripsi	Metode	Hasil
		data transaksi dalam membuat laporan, mencetak struk transaksi dan menyimpan data ke dalam <i>database</i> sehingga dapat meningkatkan pelayanan pada bisnis.		mendapatkan nilai skala 6 yaitu sebesar 1,401 yang artinya <i>website</i> siap untuk dikembangkan ketahap lebih lanjut.
7	Maulana dkk (2019)	Evaluasi dan perbaikan desain pada <i>website</i> Jawa Timur Park Group yang memiliki kekurangan seperti <i>user interface</i> yang kurang memuaskan, desain terkesan ketinggalan zaman, beberapa informasi sulit ditemukan, tidak adanya menu navigasi, pemilihan <i>font</i> dan grafik yang kurang tepat, penempatan <i>layout</i> yang terkesan kurang menarik dan desain terkesan lebih tidak <i>user friendly</i> . <i>Website</i> ini berfungsi agar masyarakat dapat melakukan pembelian tiket dan mengetahui informasi tentang berbagai wahana yang ada di Jawa Timur Park secara <i>online</i> .	<i>Goal-Directed Design</i>	Tahap pengujian awal sebelum dilakukan evaluasi dan perbaikan desain didapatkan jumlah skor SUS sebesar 46,5 yang berarti masuk dalam <i>percentile rank grade</i> F, berdasarkan dari 3 orang <i>stakeholder</i> dan 3 orang pengunjung. Sementara itu pada pengujian kedua setelah dilakukan perbaikan desain didapatkan skor SUS sebesar 82,75 yang berarti masuk ke dalam <i>percentile rank grade</i> A.
8	Ramadan dkk (2019)	Perancangan <i>user interface</i> berupa <i>prototype</i> aplikasi EzyPay untuk melakukan <i>update</i> pada desain <i>user interface</i> yang dirasa masih kurang memuaskan agar desain dapat lebih <i>clean</i> dan <i>modern</i> , serta menambahkan fitur baru yaitu saldo EzyPay agar dapat menunjang berbagai proses bisnis yang baru.	<i>Design Sprint</i>	<i>Time-based</i> yang dibutuhkan setiap <i>task</i> yaitu sebesar 0,038 atau 2/53. Hasil pengujian <i>System Usability Scale</i> (SUS) 1) $R1 = 32 \times 2,5 = 80$ 2) $R2 = 36 \times 2,5 = 90$ 3) $R3 = 36 \times 2,5 = 90$ 4) $R4 = 37 \times 2,5 = 92,5$
9	Angelina dkk (2022)	Perancangan desain aplikasi penjualan yang sesuai kebutuhan bisnis menggunakan pendekatan <i>design thinking</i> dengan menghimpun karakteristik pengguna memanfaatkan tool <i>user persona</i> agar dapat membantu bisnis dalam melakukan pendekatan kepada pelanggannya, memberikan	<i>Design Thinking</i>	Hasil pengujian menggunakan <i>usability testing</i> pada 35 halaman <i>prototype</i> dengan total 4 <i>user scenario</i> pengujian. Seluruh responden yang melakukan pengujian berhasil melaksanakan dan menyelesaikan ke-4 <i>user scenarios</i> yang telah disediakan dengan presentase keberhasilan

No	Peneliti	Deskripsi	Metode	Hasil
		kesan positif pertama kali menggunakan, dan memberikan pengalaman dan kepuasan dalam berinteraksi agar dapat meningkatkan jumlah konsumen.		mencapai 100% pada iterasi pertama.
10	Karnawan dkk (2021)	Implementasi <i>user experience</i> menggunakan metode <i>design thinking</i> pada <i>prototype</i> aplikasi <i>cleanstic</i> untuk memberi pembelajaran kepada masyarakat tentang cara mengolah sampah plastik dan memudahkan bagi masyarakat dalam menjual ataupun memberikan sampah plastiknya kepada orang atau pihak yang sedang membutuhkan.	<i>Design Thinking</i>	<p>1) Pengujian menggunakan metode SUS pada dua kali pengujian, pengujian pertama menghasilkan skor SUS sebesar 75.8 dan pengujian kedua didapatkan nilai skor SUS sebesar 78.8 dari kedua pengujian tersebut diketahui hasil pengujian yang telah dilakukan sama-sama sudah baik dan karena masuk dalam kategori <i>good</i>.</p> <p>2) Pengujian menggunakan metode <i>user experience questionnaire</i>, didapatkan skala rata-rata nilai yang dihasilkan adalah <i>attractiveness</i> sebesar 2.197, <i>perspicuity</i> sebesar 2.275, <i>efficiency</i> sebesar 2.150, <i>dependability</i> sebesar 2.050, <i>stimulation</i> sebesar 2.383 dan <i>novelty</i> sebesar 2.089.</p>

2.2 Landasan Teori

2.2.1 Website

Situs *web* atau *website* adalah kumpulan halaman *web* yang saling terhubung melalui *file digital* yang menampilkan informasi seperti *text*, gambar, video menggunakan *browser* [16]. Agar situs *web* bisa diakses oleh seluruh orang di dunia, situs *web* harus disimpan atau di hosting di komputer yang terhubung ke *internet* sepanjang waktu. Komputer tersebut dikenal dengan istilah *Web Server*. Komponen untuk membuat situs *web* terdiri dari hosting sebagai lokasi tempat situs *web* berada secara fisik, *address* untuk

menamai alamat atau URL dari *website*, halaman *web* untuk melihat informasi pada *website* tersebut, *design web* yaitu tampilan grafik pada *website*, konten sebagai isi dari halaman *website*, dan struktur navigasi yang berfungsi untuk melihat struktur *menu* atau halaman pada suatu *website*. *Website* terbagi menjadi 2 jenis [17] antara lain yaitu:

1. *Website* statis, adalah jenis *website* di mana kontennya tidak dapat berubah dengan sendirinya. Untuk mengubah konten pada *website* statis, perlu dilakukan secara manual. Istilah "statis" mengacu pada kondisi di mana tidak ada pergerakan atau perubahan yang terjadi. Dalam konteks *website* statis, hal ini berarti kontennya tetap dan tidak aktif. Dengan kata lain, setelah *website* statis dibuat, kontennya cenderung tidak mengalami perubahan kecuali jika ada intervensi langsung. Untuk melakukan perubahan konten pada *website* statis, perlu menggunakan bahasa pemrograman yang digunakan dalam aplikasi tersebut. Melalui bahasa pemrograman tersebut, pengguna dapat mengedit atau memodifikasi konten yang ada dalam *website* statis.
2. *Website* dinamis, adalah jenis *website* yang memungkinkan pengguna untuk berinteraksi secara langsung dengan melakukan aktivitas seperti menambah, mengedit, atau menghapus data dan konten dalam *website* tersebut. Keuntungan dari *website* dinamis adalah kemudahan penggunaan, efisiensi, dan kemampuan untuk melakukan pembaruan data dengan cepat. *Website* dinamis juga memiliki tampilan *layout* yang dilengkapi dengan berbagai informasi yang dapat mengalami perubahan secara dinamis. Kelemahan dari *website* dinamis adalah kompleksitasnya yang dapat menyulitkan proses pencarian di mesin pencari. Secara keseluruhan, *website* dinamis memiliki lebih banyak keunggulan daripada kekurangannya.

2.2.2 Frontend

Dalam konteks pengembangan *website*, *frontend* merujuk pada tampilan antarmuka pengguna yang berfungsi untuk menyajikan fitur dan kegunaan dari sebuah *website* ketika pengguna berinteraksi langsung dengan

website tersebut [18]. *Frontend* juga berhubungan erat dengan *user experience* saat berinteraksi dengan *website* tersebut. Aspek *visual website* yang dapat dilihat dan dialami oleh pengguna adalah *frontend*. Biasanya *frontend* dibangun dengan menggunakan komponen-komponen utama seperti HTML, CSS, dan JavaScript. Menggunakan kombinasi dari ketiga bahasa pemrograman ini, *frontend* dapat menciptakan sebuah antarmuka yang menarik, responsif, dan intuitif bagi pengguna. Dengan demikian, *frontend* memainkan peran penting dalam menciptakan pengalaman pengguna yang memuaskan dalam mengakses dan berinteraksi dengan sebuah *website*.

2.2.3 UI/UX

User Interface (UI) adalah satu-satunya jembatan yang menghubungkan pengguna ke lingkungan *web* tertentu atau aplikasi dan oleh karena itu merupakan *platform* interaksi antara pengguna dan mesin. Desain yang baik dapat mengoptimalkan kesuksesan *website* dan meningkatkan kepercayaan, peningkatan kinerja, dan mendatangkan kembali pengguna lama [19].

User Experience (UX) adalah salah satu aspek penting dalam interaksi manusia dan komputer. Tujuan dari UX adalah memberikan pemahaman kepada pengembang tentang pentingnya sistem dalam kehidupan pengguna, termasuk bagaimana pengguna melihat desain pada produk tersebut. *User experience* dapat memberikan dampak emosional dan kesan bagi pengguna saat menggunakan sistem informasi, layanan, atau aplikasi [20].

2.2.4 Prototype

Prototype adalah suatu bentuk model atau representasi dari produk yang akan dikembangkan, yang memungkinkan pengguna untuk berinteraksi dengan elemen-elemen desainnya dan melakukan evaluasi terhadap kesesuaiannya [21]. *Prototype* digunakan untuk mengeksplorasi atau mendemonstrasikan beberapa aspek yang dapat dipakai dimasa mendatang, biasanya menggunakan alat berupa kertas atau media digital untuk membuat tampilan dari desain *low-fidelity* hingga *high-fidelity* [22]. Dengan kata lain,

prototype merupakan simulasi desain yang dapat interaksi menggunakan alat atau perantara media oleh calon penggunanya.

Dalam pengembangan perangkat lunak, *prototype* merupakan versi awal yang dibuat sebagai model dasar dari produk atau sistem informasi yang sedang dikembangkan. *Prototype* ini digunakan untuk tujuan demonstrasi atau sebagai bagian dari proses pengembangan sistem. Biasanya, pembuatan *prototype* dilakukan pada tahap awal pengembangan, karena hal ini memungkinkan kesalahan dan kekurangan dapat diidentifikasi dan diperbaiki sejak awal, sehingga dapat menghemat biaya pengembangan [21]. Selain itu, kesalahan yang terdapat pada *prototype* dapat lebih mudah dan murah untuk diperbaiki dibandingkan dengan produk yang telah selesai.

Prototype memiliki peran penting dalam evaluasi dan mendapatkan umpan balik awal dari pengguna, serta memfasilitasi interaksi antara pengguna dan pengembang dengan memungkinkan mereka melihat, merasakan, dan berinteraksi langsung dengan representasi produk [21]. Dengan kata lain, *prototype* digunakan untuk menguji ide, menggambarkan kebutuhan, dan merangsang tanggapan dari pengguna. Dengan demikian, keberadaan *prototype* sangat krusial dalam proses desain dan pengembangan produk.

Dalam bidang Interaksi Manusia dan Komputer, *prototype* digunakan sebagai salah satu alat pengembangan untuk produk tertentu. Contohnya termasuk sketsa layar, papan cerita, presentasi PowerPoint, simulasi video, dan model desain menggunakan bahan seperti kayu, plastik, atau kaca. Pembangunan *prototype* memiliki tujuan utama untuk menguji kelayakan desain (fisibilitas) [21]. Dengan demikian, tujuan dari pembuatan *prototype* adalah memastikan bahwa desain dapat dikembangkan dan produk berikutnya dapat diimplementasikan dengan sukses.

Seperti yang sudah dibahas, *prototype* memiliki bentuk yang berbeda satu sama lain tergantung pada fungsi dan tujuan itu dibangun. Tiga jenis *prototype* yang sering digunakan oleh banyak desainer atau pengembang [21] yaitu:

1) *Low-Fidelity*

Prototype jenis ini menggunakan bahan atau media yang berbeda dengan produk akhir, sehingga memudahkan perubahan sesuai kebutuhan. *Low-fidelity prototype* biasanya dibuat pada tahap awal pengembangan dan sering digunakan bersama dengan skenario yang menjelaskan desain tertentu. Dalam pengembangan perangkat lunak, *low-fidelity prototype* umumnya terdiri dari gambar sederhana, bentuk balok, serta elemen dasar seperti judul, *label*, dan lainnya.

2) *Mid-Fidelity*

Jenis *prototype* ini memiliki kelebihan dari biayanya yang lebih murah dari biaya *prototype high-fidelity*, namun dari segi tampilan dan fungsi tidak dapat memberikan hasil yang mendekati seperti produk akhir. Biasanya desainer perangkat lunak menambahkan beberapa element yang lebih detail dari *low-fidelity* dan menggunakan warna abu-abu untuk menonjolkan elemen individual di dalamnya.

3) *High-Fidelity*

Prototype jenis ini memiliki kemiripan yang tinggi dengan produk yang sebenarnya, sehingga pengguna mungkin menganggapnya sebagai produk yang sudah jadi. *Prototype* ini memberikan pengalaman kepada pengguna yang hampir serupa dengan menggunakan produk yang sebenarnya, karena penampilan dan sebagian besar fungsi telah lengkap menyerupai produk akhir yang diusulkan. Namun, jenis *prototype* ini memerlukan biaya yang tinggi dan waktu yang lebih lama dalam pembuatannya. Pada jenis *prototype* ini, desain perangkat lunak sudah mencakup gambar asli, warna, dan konten yang relevan.

Secara garis besar, dapat disimpulkan bahwa dalam proses desain, *prototype low-fidelity* digunakan pada tahap awal untuk eksplorasi dan

validasi konsep, kemudian *prototype mid-fidelity* digunakan untuk mengembangkan dan menguji desain yang lebih rinci. Terakhir, *prototype high-fidelity* digunakan untuk menciptakan representasi yang sangat mirip dengan produk akhir dan untuk pengujian akhir sebelum implementasi. Dengan demikian, penggunaan *prototype* dalam tiga tingkatan ini mengikuti kemajuan dan kedalaman pengembangan desain.

2.2.5 Usability

Usability merupakan komponen yang tak terpisahkan dari *User Experience (UX)*. *Usability* mencerminkan persepsi pengguna akhir terkait dengan tingkat keefektifan, keefisienan, dan kepuasan mereka dalam menyelesaikan tugas saat menggunakan aplikasi. Tidak ada peraturan atau konsensus yang baku mengenai karakteristik-karakteristik *usability*, sehingga peneliti dan pengembang perangkat lunak memiliki keleluasaan dalam menentukan parameter-parameter yang akan diuji dan dievaluasi [23].

Usability adalah ukuran sejauh mana suatu produk atau layanan dapat digunakan oleh pengguna tertentu untuk mencapai tujuan yang telah ditentukan, dengan memperhatikan efektivitas, efisiensi, dan tingkat kepuasan dalam konteks penggunaan yang spesifik [24]. *Usability* adalah properti untuk menilai kualitas penggunaan dalam suatu konteks pada keseluruhan sistem [25]. Pengukuran *usability* tergantung pada interaksi antara pengguna, produk, tugas, dan lingkungan [26]. Kualitas *usability* dapat diukur dari tiga aspek [25] yaitu, (efektivitas) sejauh mana tujuan penggunaan sistem secara keseluruhan tercapai, (efisiensi) sumber daya yang harus dikeluarkan untuk mencapai tujuan yang dimaksud dan (kepuasan) sejauh mana pengguna dapat menerima sistem.

2.2.6 User-Centered Design (UCD)

Pendekatan *User-Centered Design (UCD)* merupakan suatu metode dalam desain dan pengembangan antarmuka pengguna yang melibatkan partisipasi pengguna selama proses desain dan pengembangan [27]. Desain yang berorientasi pada pengguna tidak hanya memprioritaskan pemahaman tentang pengguna sistem komputer yang sedang dikembangkan, tetapi juga

memperhatikan pemahaman tentang tugas yang akan dilakukan oleh pengguna dengan sistem tersebut serta lingkungan di sekitar mereka, seperti konteks organisasi, sosial, dan fisik di mana sistem tersebut akan digunakan. Mengambil pendekatan *user-centered design* harus mengoptimalkan usability pada sistem komputer. Metode ini membantu membuat perangkat lunak yang dapat memenuhi kebutuhan penggunanya. Desain yang berpusat pada pengguna lebih dari sekadar mengumpulkan persyaratan pengguna dan mengubahnya menjadi persyaratan fungsional [28]. Filosofi di balik desain yang berpusat pada pengguna adalah: pengguna paling tahu [29].

Proses desain pada UCD membutuhkan seorang desainer yang dapat memahami elemen hasil observasi seperti survei dan wawancara kepada pengguna untuk bisa diintegrasikan hasil kebutuhan pengguna menjadi desain sebuah sistem, selain itu desainer harus mampu melakukan pendekatan dengan metode *brainstorming* (Curah Pendapat) untuk menemukan dan menentukan solusi dari kebutuhan pengguna.

User-centered design memiliki prinsip utama untuk mencapai *usability* yang baik pada sistem, keempat prinsip utama itu adalah [27]:

- 1) Keterlibatan aktif pengguna
- 2) Alokasi fungsi yang tepat antara pengguna dan sistem
- 3) Iterasi solusi desain
- 4) Tim desain multidisiplin

Empat aktivitas desain yang terdapat pada *user-centered design* [27] adalah:

- 1) *Understand Context of Use*

Dalam merancang sistem, penting bagi perancang untuk memahami konteks penggunaan, seperti pengguna yang akan menggunakan aplikasi, tujuan penggunaannya, serta situasi di mana aplikasi tersebut akan digunakan.

- 2) *Specify User Requirements*

Setelah perancang memperoleh pemahaman tentang konteks penggunaan aplikasi, langkah selanjutnya adalah menentukan kebutuhan pengguna

(*user requirements*). Dalam proses ini, perancang harus dapat mengidentifikasi kebutuhan pengguna yang relevan dengan tujuan bisnis yang ingin dicapai.

3) *Design Solutions*

Proses selanjutnya adalah mendesain solusi dari *user requirements* yang telah dijelaskan pada proses sebelumnya, proses desain ini akan melalui beberapa tahapan mulai dari perancangan design sistem dalam bentuk diagram, hingga menciptakan desain *prototype* yang dapat di coba pengguna.

4) *Evaluation Against Requirements*

Evaluasi akan dilakukan dengan melibatkan pengguna yang akan menggunakan sistem. Evaluasi dimulai dari satu proses dan dilanjutkan ke proses selanjutnya.

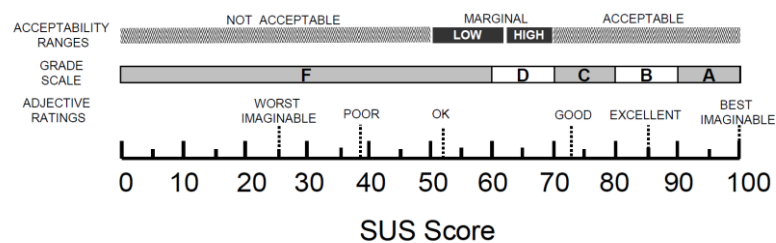
2.2.7 *System Usability Scale (SUS)*

System Usability Scale (SUS) adalah sebuah standar kuesioner yang sering digunakan dalam evaluasi kegunaan sistem [30]. SUS dikembangkan pertama kali oleh John Brooke sejak 1986. SUS merupakan pilihan yang ideal dalam mengukur kepuasan pengguna karena efisiensinya yang tinggi dalam pengelolaan yang cepat dan biaya yang terjangkau. Metode ini diakui sebagai salah satu cara yang paling efisien dalam mengumpulkan data yang valid secara statistik, dengan memberikan skor yang jelas dan akurat [23]. Karena alasan ini, penggunaan SUS sering disebut sebagai "*Quick and Dirty test*".

Biasanya, dalam kuesioner *System Usability Scale (SUS)*, digunakan skala *Likert* dengan rentang 1 hingga 5. Skor 1 menunjukkan bahwa responden "Sangat Tidak Setuju" dengan pernyataan yang diberikan. Skor 2 berarti responden "Tidak Setuju", skor 3 mewakili sikap "Netral", skor 4 mengindikasikan "Setuju", dan skor 5 menunjukkan bahwa responden "Sangat Setuju". Rentang skor kuesioner SUS adalah 0 hingga 100, dan situs web umumnya dianggap memuaskan jika skornya di atas 68 [23].

SUS terdiri dari 10 item pertanyaan. Item pertanyaan dengan nomor ganjil (1, 3, 5, 7, dan 9) memiliki arti positif, sedangkan item dengan nomor

genap (2, 4, 6, 8, dan 10) memiliki arti negatif. Untuk item pertanyaan bernomor ganjil, kontribusi skor diperoleh dengan mengurangi 1 dari posisi skala. Sedangkan untuk item pertanyaan bernomor genap, kontribusi skor diperoleh dengan mengurangi posisi skala dari 5. Setelah itu, total skor yang diperoleh dikalikan dengan 2,5 untuk menghasilkan nilai keseluruhan SUS. Sistem penilaian SUS mensyaratkan untuk 10 item pertanyaan terisi semua, jadi jika responden mengosongkan salah satu item, maka item tersebut harus diberi skor mentah yaitu posisi yang berada di tengah dari 5 poin jawaban [30].



Gambar 2.1 Skala Penilaian Skor SUS [31]

Berdasarkan skala penilaian System Usability Scale pada Gambar N diatas, interpretasi skor SUS dapat dibagi menjadi lima *grade*: *Grade A* (90-100) menunjukkan kepuasan sangat tinggi, *Grade B* (80-90) adalah kepuasan baik dengan ruang perbaikan kecil, *Grade C* (70-80) adalah kepuasan yang cukup dengan perbaikan yang diperlukan, *Grade D* (60-70) menunjukkan masalah signifikan yang perlu diperbaiki, dan *Grade F* (<60) adalah tingkat kepuasan yang sangat rendah, mungkin tidak dapat diterima. Ini membantu pemilik produk atau layanan memahami kekuatan dan kelemahan pengalaman pengguna. Detail interpretasi skor SUS berdasarkan *grade scale* dapat dilihat seperti pada Tabel 2.2.

Tabel 2.2 Interpretasi SUS Score Berdasarkan Grade Scale

<i>SUS Score</i>	<i>Grade Scale</i>
100 – 90	A
90 – 80	B
80 – 70	C
60 – 70	D
< 60	F

2.2.8 Node js

Node.js adalah *environment runtime* JavaScript yang mencakup semua yang diperlukan untuk menjalankan program yang ditulis dalam bahasa JavaScript [32]. JavaScript ditunjukkan untuk digunakan pada program di web *browser*. Node.js dapat membuat JavaScript memiliki banyak potensi untuk bisa dipakai di luar web *browser*, bukan hanya sekedar bisa membuat *website* yang interaktif. Node js menggunakan mesin *runtime* dari Google yang bernama V8, mesin V8 mendukung bahasa JavaScript untuk berjalan di browser, khususnya Google Chrome. V8 dan Node js sebagian besar dibuat dengan menggunakan bahasa pemrograman C dan C++, dengan fokus pada kinerja dan konsumsi memori yang rendah [33].

2.2.9 React js

React js adalah *library* JavaScript digunakan untuk mengembangkan tampilan *user interface* pada *website* khusus *Single Page Application* (SPA) [32]. SPA adalah jenis aplikasi *web* yang terdiri dari satu halaman *web* yang menangani tindakan dinamis tanpa perlu *me-refresh* halaman [34]. Dengan menggunakan teknologi SPA, interaksi antara pengguna dan sistem dapat ditangani tanpa harus sering menggunakan *server*, meningkatkan kinerja dengan waktu muat yang lebih cepat, dan kemudahan navigasi antarhalaman. Implementasi SPA bertujuan untuk mengurangi beban kerja *server* saat permintaan data dari *client*, serta mengoptimalkan penggunaan sumber daya oleh *server*. Dengan menggunakan satu halaman *web* sebagai tampilan, SPA memberikan pengalaman pengguna yang mirip dengan aplikasi *desktop*.

React dikembangkan oleh Facebook, pada tahun 2013 *library* ini dirilis ke publik. React pada dasarnya adalah kerangka kerja web yang dirancang untuk mengatasi masalah kinerja dalam aplikasi web. React js digunakan untuk menangani banyak lapisan tampilan aplikasi web yang membutuhkan banyak komponen berbeda untuk dapat digunakan kembali. React js bisa populer di kalangan pengembang web dan aplikasi adalah karena aplikasi web yang menggunakan React js dapat melakukan perubahan data tanpa harus menyegarkan ulang halaman (*refresh page*).

Reactjs berbasis komponen, artinya ketika sebuah halaman dimuat, semua elemen yang ditampilkan di layar dibagi menjadi beberapa komponen. Dengan menggunakan React js setiap komponen yang melakukan perubahan, maka komponen lain tidak akan terpengaruh dan tidak akan perlu melakukan *refresh* pada halaman. Hal ini membuat *website* yang dibuat menggunakan react js tampak lebih sedikit dari halaman web biasa dan membuat website tampak seperti aplikasi yang berjalan dengan cepat. Pengembang dapat menggunakan satu komponen untuk banyak halaman tanpa perlu menuliskan kembali code untuk masing-masing halaman. React menggunakan DOM *virtual* yang memutuskan apakah komponen harus dimuat ulang atau tidak berdasarkan status komponen saat itu dan pada saat perubahan yang telah terjadi, hal tersebut mencegah agar aplikasi tidak merender ulang.

2.2.10 Blackbox Testing

Pengujian *blackbox* didasarkan pada pengetahuan yang dikumpulkan dari persyaratan dan dokumentasi fungsionalitas pada perangkat lunak yang dibuat [35]. Oleh karena itu mereka juga disebut pengujian teknik perilaku atau fungsional. Pengujian *blackbox* mengabaikan struktur internal dari objek yang diuji [36]. Dengan demikian, *blackbox* tidak mempertimbangkan bagaimana cara pengimplementasian dari perangkat lunak yang diuji, akan tetapi berdasarkan pada hasil uji pakai apakah sesuai dengan persyaratan sistem yang dibuat. Dengan kata lain pengujian ini tidak perlu memeriksa sumber kode pada proses pengujian, penguji tidak perlu memiliki pengetahuan tentang bahasa pemrograman tertentu dan pengetahuan tentang pembuatan perangkat lunak. *Blackbox* dapat membantu untuk memvalidasi fungsionalitas dari suatu sistem secara keseluruhan. Pengujian ini dapat mengidentifikasi input atau output yang *valid* dan tidak *valid* berdasarkan perspektif pengguna sesuai dengan spesifikasi awal sistem, sehingga spesifikasi yang tidak sesuai dapat dengan mudah diidentifikasi dan ditangani segera.