

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terkait

Sebelum penelitian ini dilakukan telah ada penelitian terkait yang membuat peneliti untuk melakukan penelitian yang berkaitan dengan design *user interface* dan *user experience* menggunakan metode *design thinking*.

Pada tahun 2023, Ratna Nur Fadilah dan Dhian Sweetania melakukan penelitian yang berjudul Perancangan Design Prototype UI/UX Aplikasi Reservasi Restoran Dengan Menggunakan Metode Design Thinking. Menurut peneliti, perkembangan usaha kuliner yang semakin pesat menuntut adanya kemudahan dalam proses reservasi restoran, terutama di kota-kota besar seperti Tangerang, Depok, dan Bogor. Kebutuhan akan platform yang memudahkan calon konsumen untuk melakukan reservasi restoran menjadi latar belakang utama penelitian ini. Tujuan penelitian adalah merancang sebuah aplikasi reservasi restoran dengan fokus pada desain antarmuka pengguna (UI) dan pengalaman pengguna (UX) yang mudah dipahami. Metode penelitian yang digunakan adalah Design Thinking, dengan alat bantu Figma untuk membuat prototipe desain UI/UX. Penelitian ini mencakup 5 restoran di daerah Tangerang, Depok, dan Bogor, dan hanya berfokus pada tahap pembuatan prototipe desain. Hasil penelitian menunjukkan bahwa pengujian usability testing terhadap prototipe aplikasi menggunakan Maze menghasilkan nilai sebesar 85, sedangkan pengolahan data aspek usability testing memperoleh nilai 81,6%. Kesimpulan dari penelitian ini adalah prototipe aplikasi reservasi restoran yang dirancang telah memenuhi kebutuhan pengguna dan

memenuhi nilai-nilai aspek usability testing, sehingga dapat dikatakan bahwa desain yang dihasilkan efektif dan mudah dipahami oleh pengguna[1].

Pada tahun yang sama di 2023 Istiqomah Br Karo Sekali dan rekan-rekan melakukan penelitian yang berjudul Perancangan UI/UX Aplikasi Mobile Produk Fashion Pria pada Toko Celcius di Kota Manado Menggunakan Design Thinking. Berdasarkan latar belakang penelitian, Celcius sebagai brand fashion pria di bawah PT. Aditya Mandiri Sejahtera menghadapi permasalahan dalam sistem pembelian online melalui Instagram, seperti ketiadaan informasi stok, pembaruan produk yang tidak berkala, dan proses pembelian yang harus dilakukan melalui dua aplikasi berbeda (Instagram dan WhatsApp). Hal ini menyebabkan kesulitan bagi pembeli dalam memperoleh informasi produk dan melakukan transaksi. Tujuan penelitian ini adalah merancang UI/UX aplikasi mobile untuk toko Celcius menggunakan metode Design Thinking guna memudahkan pembeli dalam melakukan pembelian dan memperoleh informasi produk. Metode penelitian yang digunakan meliputi perancangan desain dan pengujian kegunaan melalui usability testing dengan instrumen kuesioner Single Ease Question (SEQ) dan System Usability Scale (SUS). Hasil penelitian menunjukkan bahwa perancangan UI/UX dinilai berhasil dan efektif, dengan skor SEQ sebesar 6,85 yang mengindikasikan pengguna tidak mengalami kesulitan dalam menggunakan sistem. Selain itu, skor SUS sebesar 95 (dari 10 responden) menempatkan aplikasi dalam kategori “Best Imaginable” dengan grade A, menunjukkan kepuasan pengguna dan pengalaman yang sangat baik saat menggunakan aplikasi. Dengan demikian, dapat disimpulkan bahwa perancangan UI/UX aplikasi mobile untuk Celcius telah

berhasil memenuhi tujuan penelitian dan memberikan solusi atas permasalahan yang dihadapi dalam proses pembelian online[2].

Pada tahun yang sama di 2023, Jovianto Marcellino Dumalang dan rekan-rekan, melakukan penelitian yang berjudul Perancangan UI/UX Aplikasi Penjualan Makanan Berbasis Mobile pada UMKM di Kota Manado menggunakan metode Design Thinking, penelitian ini bertujuan untuk merancang User Interface (UI) dan User Experience (UX) guna memudahkan masyarakat dalam membeli makanan di UMKM Kuliner. Metode yang digunakan adalah Design Thinking, yang meliputi tahapan Empathize, Define, Ideate, Prototype, dan Testing. Metode ini dipilih untuk menciptakan solusi atas permasalahan yang dihadapi pengguna. Pengujian dilakukan melalui Usability Testing dengan melibatkan 10 responden. Hasil pengujian menggunakan Single Ease Question menunjukkan skor 6,7 dari 7, sedangkan pengujian dengan System Usability Scale menghasilkan skor 93 dari 100. Hasil ini mengindikasikan bahwa perancangan UI/UX aplikasi penjualan makanan berbasis mobile pada UMKM di kota Manado dapat digunakan dengan baik oleh responden dan mampu memberikan solusi atas permasalahan yang ada[3].

Pada tahun 2024, Muhammad Zidhan dan rekan-rekan melakukan penelitian yang berjudul Perancangan UI/UX Pada Aplikasi Sayur Mayur Online (Samaron) Menggunakan Metode Design Thinking. Berdasarkan latar belakang penelitian, masalah yang sering dihadapi oleh masyarakat, mahasiswa, dan anak kost adalah kesulitan dalam membeli sayuran secara langsung karena kesibukan mereka. Oleh karena itu, penelitian ini bertujuan untuk merancang aplikasi bernama Samaron

yang memudahkan pengguna dalam membeli sayuran secara online serta memberikan rekomendasi menu masakan beserta cara memasaknya. Metode penelitian yang digunakan adalah Design Thinking, yang meliputi tahapan Empathize, Define, Prototype, dan Testing. Hasil penelitian menunjukkan bahwa desain UI/UX aplikasi Samaron dinilai mudah digunakan, terutama bagi orang yang memiliki waktu terbatas. Analisis data survei menggunakan System Usability Scale (SUS) menghasilkan skor 76,25, yang termasuk dalam kategori Good pada adjective rating dan C pada grade scale. Dengan demikian, dapat disimpulkan bahwa aplikasi Samaron berhasil dirancang untuk memenuhi kebutuhan pengguna yang sibuk dan memberikan pengalaman penggunaan yang baik[4].

Kemudian di tahun yang sama pada 2024, Ahmad Rayhaan Yusri dan rekan-rekan melakukan penelitian dengan judul Perancangan Desain UI/UX Berbasis Scan Barcode Dengan Metode Design Thinking Untuk Pemesanan Makanan, penelitian tersebut berfokus pada pengembangan solusi pemesanan makanan digital berbasis scan barcode untuk meningkatkan efisiensi dan pengalaman pelanggan di Warkop Bu Haji sebagai lokasi penelitian. Latar belakang yang disajikan oleh peneliti dalam jurnalnya menunjukkan adanya kebutuhan untuk mengatasi ketidaknyamanan dan waktu tunggu yang lama dalam pemesanan manual, sementara tujuan penelitian adalah menciptakan sistem pemesanan yang lebih efisien dengan memanfaatkan teknologi terbaru. Metode penelitian yang digunakan adalah Design Thinking, yang mengoptimalkan desain antarmuka pengguna (UI) dan pengalaman pengguna (UX) berdasarkan kebutuhan

pengguna. Hasil evaluasi terhadap prototipe menunjukkan tingkat kepuasan yang tinggi, dengan nilai 86,6% dari pelanggan dan 90% dari mitra, serta kepuasan pelanggan mencapai 97,1% dan mitra 91,6%[5].

2.2 Landasan Teori

2.2.1 Design Thinking

Design thinking adalah salah satu pendekatan dalam proses desain yang berfokus pada pengguna atau *user* dalam menyelesaikan masalah. Metode ini menggabungkan kepekaan seorang desainer dengan teknik-teknik yang sesuai untuk memadukan kebutuhan pengguna dengan teknologi yang relevan serta strategi bisnis yang mampu menciptakan nilai bagi pelanggan dan peluang pasar.

Design thinking menggunakan tiga tahapan proses untuk menciptakan inovasi yang efektif. Ketiga tahapan tersebut meliputi inspirasi, ideasi, dan implementasi. Berbeda dengan metodologi konvensional yang biasanya bersifat sekuensial, ketiga tahapan dalam design thinking dapat saling berinteraksi dan tidak harus dilakukan secara berurutan selama proses pengerjaannya[6].

1) Empatize

Empathy map adalah sebuah dokumen yang digunakan untuk memvisualisasikan dan merangkum informasi yang telah diketahui mengenai tipe pengguna tertentu. Dokumen ini berfungsi untuk mengeksternalisasi pengetahuan yang diperoleh dari pengguna agar tercipta pemahaman bersama di antara tim pengembang, sekaligus menjadi dasar dalam pengambilan keputusan desain yang berorientasi pada kebutuhan pengguna.

Struktur empathy map terdiri dari empat kuadran utama yang merepresentasikan aspek-aspek kognitif dan perilaku pengguna yang diamati selama proses observasi atau wawancara pada tahap riset. Empat kuadran tersebut mencerminkan hal-hal yang diungkapkan pengguna (apa yang mereka katakan), tindakan yang dilakukan (apa yang mereka lakukan), serta aspek internal seperti pikiran (think) dan perasaan (feel) mereka.

Kelebihan dari penggunaan empathy map antara lain kemampuannya dalam menyajikan kebutuhan pengguna secara visual dengan cepat, memfasilitasi pemahaman terhadap sudut pandang pengguna secara langsung, serta mendorong diskusi kolaboratif antar pemangku kepentingan. Selain itu, empathy map bersifat fleksibel dan mudah diperbarui, sehingga dapat diadaptasi berdasarkan temuan data terbaru yang muncul selama proses iteratif desain..

2) Define

Tahap Define merupakan langkah lanjutan setelah memahami kebutuhan pengguna. Pada fase ini, tim pengembang mulai merumuskan dan menggambarkan ide-ide utama yang akan menjadi fondasi dalam proses perancangan produk atau aplikasi. Fokus utama dari tahap ini adalah menyaring temuan dari proses empathize menjadi pernyataan masalah yang jelas dan berorientasi pada pengguna.

a. User Persona

User persona adalah dokumen representatif yang menggambarkan karakteristik dari tipe pengguna utama. Dokumen ini dirancang untuk membantu proses analisis pengguna dengan tujuan mengidentifikasi kebutuhan, tujuan, serta

minat mereka. Melalui user persona, tim pengembang dapat memahami sasaran pengguna secara lebih mendalam, termasuk perilaku, motivasi, serta kendala atau pain point yang mereka alami. Penggunaan user persona memungkinkan pengambilan keputusan desain yang lebih tepat karena mempertimbangkan aspek psikologis, perilaku, dan demografis pengguna yang relevan

b. User Scenario

User scenarios merupakan narasi yang menjelaskan alur interaksi pengguna dengan sistem atau aplikasi secara rinci. Skenario ini menyajikan urutan tindakan yang dilakukan pengguna, mulai dari bagaimana mereka menemukan aplikasi, fitur apa yang mereka akses, area mana yang mereka klik, hingga durasi interaksi mereka di dalam aplikasi. Skenario yang efektif harus disusun secara ringkas namun jelas, sehingga dapat memberikan gambaran konkret tentang pengalaman pengguna dalam konteks nyata

3) Ideate

Tahap Ideate merupakan fase di mana tim mulai merumuskan berbagai alternatif solusi yang dapat menjawab kebutuhan pengguna, setelah kebutuhan tersebut dirumuskan secara jelas pada tahap Define. Pada tahap ini, eksplorasi ide-ide kreatif dilakukan secara terbuka untuk menghasilkan beragam kemungkinan pendekatan solusi yang inovatif dan relevan dengan masalah yang telah diidentifikasi.

Proses ideation biasanya melibatkan teknik seperti brainstorming, mind mapping, ataupun metode scamper guna menggali potensi solusi secara luas.

Seluruh ide yang muncul akan dievaluasi dan diseleksi berdasarkan kelayakan, kesesuaian dengan kebutuhan pengguna, serta potensi implementasinya..

4) Prototype

Tahap Prototype merupakan proses implementasi ide-ide yang telah dihasilkan pada fase sebelumnya ke dalam bentuk representasi nyata, baik itu berupa produk, antarmuka, maupun simulasi interaktif. Tujuan utama dari tahap ini adalah untuk mewujudkan skenario penggunaan yang merefleksikan kebutuhan dan ekspektasi pengguna secara lebih konkret.

Prototyping memungkinkan tim pengembang menguji bagaimana solusi yang dirancang akan digunakan dalam konteks sebenarnya, serta memperoleh umpan balik awal dari pengguna. Representasi ini tidak harus berupa produk akhir, melainkan cukup sebagai versi awal *low-fidelity* atau *high-fidelity* yang dapat membantu dalam mengevaluasi fungsionalitas, alur interaksi, dan efektivitas desain

5) Test

Tahap Test merupakan fase kritis dalam proses *Design Thinking* di mana produk atau prototipe yang telah dikembangkan diuji langsung oleh pengguna akhir. Pengujian ini bertujuan untuk mengevaluasi sejauh mana solusi yang diimplementasikan telah memenuhi kebutuhan, harapan, serta kenyamanan pengguna dalam konteks penggunaan nyata. Melalui tahap ini, diperoleh wawasan mendalam mengenai pengalaman pengguna, termasuk kelebihan, kekurangan, dan potensi perbaikan terhadap sistem.

Salah satu metode yang umum digunakan dalam tahap ini adalah *System Usability Scale* (SUS), yaitu instrumen evaluasi kuantitatif berbasis kuesioner yang terdiri dari 10 pernyataan. SUS memberikan skor usability yang menggambarkan tingkat kemudahan penggunaan sistem secara keseluruhan dari perspektif pengguna. Hasil dari pengujian ini menjadi landasan untuk melakukan penyempurnaan desain sebelum produk diluncurkan secara final.

2.2.2 User Interface

User Interface (UI) atau Antarmuka Pengguna adalah salah satu aspek kunci dalam pengembangan aplikasi atau sistem digital. UI mencakup semua elemen visual dan interaktif yang memungkinkan pengguna berkomunikasi dengan suatu produk. Ini termasuk tombol, ikon, warna, typography, gambar, dan tata letak yang dibuat untuk memandu pengguna dalam menyelesaikan tugas tertentu [7]. Tujuan utama UI adalah menciptakan antarmuka yang tidak hanya menarik secara estetika tetapi juga intuitif dan mudah digunakan. Dengan UI yang baik, pengguna dapat berinteraksi dengan sistem tanpa kebingungan atau kesulitan.

UI berfokus pada bagaimana suatu produk terlihat dan dirasakan secara visual. Desainer UI bertanggung jawab untuk memastikan bahwa setiap elemen antarmuka konsisten, selaras dengan identitas merek, dan sesuai dengan prinsip-prinsip desain yang baik. Misalnya, pemilihan warna yang tepat dapat memengaruhi emosi pengguna, sementara tata letak yang terorganisir dapat meningkatkan keterbacaan dan kemudahan navigasi. UI yang dirancang dengan baik tidak hanya membuat aplikasi terlihat profesional tetapi juga membangun kepercayaan pengguna.

Selain aspek visual, UI juga mencakup interaktivitas antara pengguna dan sistem. Ini melibatkan cara pengguna berinteraksi dengan elemen-elemen antarmuka, seperti mengklik tombol, menggeser layar, atau mengisi formulir. Desainer UI harus memastikan bahwa interaksi ini lancar, responsif, dan memberikan umpan balik yang jelas kepada pengguna. Misalnya, ketika pengguna menekan tombol, perubahan warna atau animasi kecil dapat memberi tahu pengguna bahwa tindakan mereka telah diterima. Interaktivitas yang baik meningkatkan kepuasan pengguna dan membuat pengalaman menggunakan aplikasi lebih menyenangkan[8].

2.2.3 User Experience

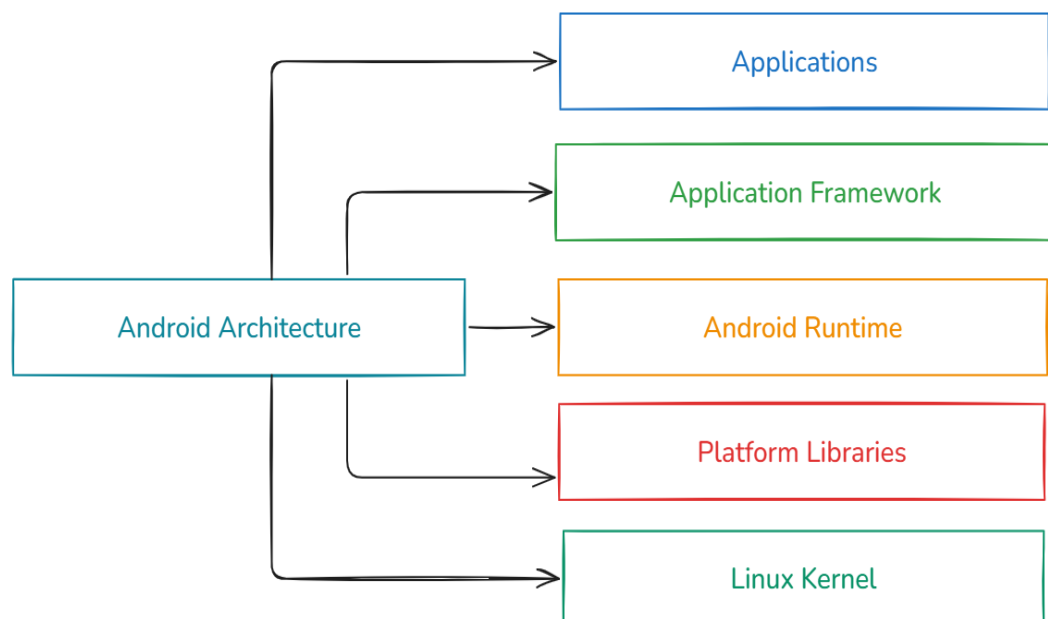
User Experience (UX) atau Pengalaman Pengguna adalah konsep yang mencakup seluruh interaksi dan perasaan pengguna saat menggunakan suatu produk, sistem, atau layanan. UX tidak hanya terbatas pada aspek visual atau fungsional, tetapi juga melibatkan emosi, persepsi, dan kepuasan pengguna secara keseluruhan. Tujuan utama UX adalah menciptakan pengalaman yang bermakna, menyenangkan, dan memenuhi kebutuhan pengguna. Dengan UX yang baik, pengguna tidak hanya dapat menyelesaikan tugas dengan mudah tetapi juga merasa puas dan terhubung dengan produk yang digunakan .

UX dimulai dengan penelitian pengguna untuk memahami kebutuhan, perilaku, dan tantangan yang dihadapi oleh target audiens. Proses ini melibatkan metode seperti wawancara, survei, observasi, atau analisis data untuk mengumpulkan wawasan mendalam tentang pengguna. Hasil penelitian ini kemudian digunakan untuk merancang solusi yang sesuai dengan ekspektasi dan

preferensi pengguna. Misalnya, jika pengguna menginginkan aplikasi yang cepat dan mudah digunakan, desainer UX akan fokus pada penyederhanaan alur pengguna dan mengurangi langkah-langkah yang tidak perlu.

2.2.4 Android

Android adalah sistem operasi berbasis kernel Linux yang dirancang terutama untuk perangkat mobile layar sentuh seperti smartphone dan tablet. Dikembangkan oleh Open Handset Alliance yang dipimpin oleh Google, Android pertama kali dirilis secara komersial pada tahun 2008. Sejak itu, Android telah menjadi sistem operasi mobile paling populer di dunia, mengalahkan pesaingnya seperti iOS dari Apple. Keberhasilan Android tidak lepas dari sifatnya yang open-source, memungkinkan produsen perangkat untuk memodifikasi dan menyesuaikan sistem operasi ini sesuai kebutuhan mereka.[9]



Gambar 2. 1 Arsitektur Android

Arsitektur Android terdiri dari beberapa lapisan, mulai dari kernel Linux di bagian paling dasar, diikuti oleh libraries, Android runtime, application framework, dan lapisan teratas berupa aplikasi itu sendiri. Kernel Linux bertanggung jawab atas manajemen memori, keamanan, dan driver perangkat keras. Lapisan libraries menyediakan berbagai fungsi inti seperti web browsing, database SQLite, dan pemutaran media. Android runtime, yang termasuk mesin virtual Dalvik, menjalankan aplikasi yang ditulis dalam bahasa Java. Android sebagai sistem operasi memiliki berbagai jenis atau varian yang dikembangkan untuk memenuhi kebutuhan spesifik perangkat atau pasar tertentu. Berikut adalah beberapa jenis Android yang paling umum

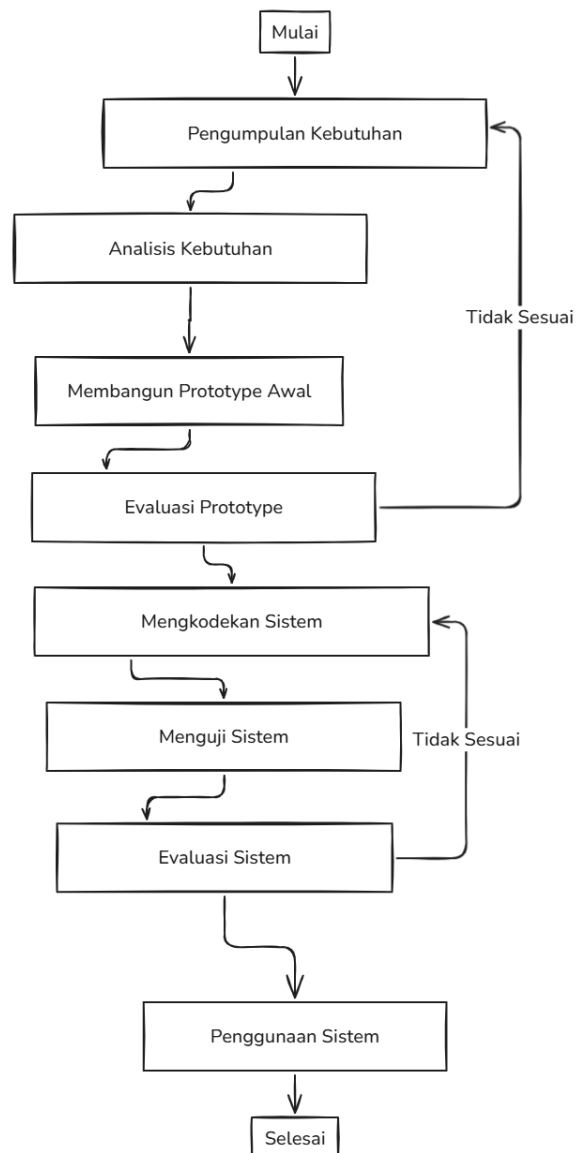
2.2.5 Prototype

Prototype merupakan salah satu pendekatan dalam rekayasa perangkat lunak yang memungkinkan pengembang untuk secara langsung menunjukkan cara kerja suatu perangkat lunak atau komponen-komponennya dalam lingkungan nyata sebelum tahap pembangunan sebenarnya dimulai. Metode Prototype adalah salah satu metode dalam pengembangan perangkat lunak yang dianggap sebagai paradigma baru, menandai evolusi dalam dunia pengembangan perangkat lunak. Metode ini merevolusi pendekatan tradisional seperti Metode Waterfall yang bersifat sekuensial.

Dalam metode prototype, perangkat lunak yang dikembangkan kemudian dipresentasikan kepada klien, yang kemudian diberi kesempatan untuk memberikan masukan dan kritik. Hal ini memastikan bahwa perangkat lunak yang dihasilkan benar-benar sesuai dengan kebutuhan dan harapan pelanggan. Proses

perubahan pada perangkat lunak dapat dilakukan berulang kali hingga tercapai kesepakatan antara pengembang dan klien mengenai bentuk akhir dari perangkat lunak yang akan dikembangkan[10].

Tahap-tahap dalam prototype dapat dikatakan merupakan tahap-tahap yang dipercepat. Strategi utama dalam prototype adalah kerjakan yang mudah terlebih dahulu dan sampaikan hasil kepada pengguna sesegera mungkin



Gambar 2. 2 Tahapan Prototype

1) Pengumpulan Kebutuhan

Klien dan pengembang bersama-sama mendefinisikan format software, mengidentifikasi kebutuhan dan sistem yang dibuat.

Pada tahap ini, dilakukan sesi wawancara, observasi, dan diskusi langsung antara pihak pengembang dan klien untuk memahami harapan serta permasalahan yang dihadapi pengguna akhir. Proses ini bertujuan untuk

menggali kebutuhan fungsional fitur yang harus dimiliki sistem dan non-fungsional seperti keamanan, kecepatan akses, dan kemudahan penggunaan.

2) Membangun Prototype/Prototype

Membangun prototype dengan membuat perancangan sementara yang berfokus penyajian kepada pelanggan (contoh membuat input dan format output).

Proses pembangunan prototype bertujuan untuk menghasilkan representasi awal dari sistem yang akan dikembangkan. Prototype ini tidak hanya menampilkan rancangan visual, tetapi juga menunjukkan alur interaksi pengguna dengan sistem.

Dalam konteks pengembangan berbasis kebutuhan pengguna, prototype berperan sebagai alat verifikasi awal yang memungkinkan klien mengevaluasi apakah rancangan tersebut sesuai dengan ekspektasi dan kebutuhan operasional mereka.

3) Evaluasi *Prototype*

Tahap ini dilakukan oleh klien untuk memastikan apakah prototype yang dibangun sudah sesuai dengan keinginan dan kebutuhan pelanggan atau belum. Jika tidak sesuai, prototype akan direvisi dengan mengulangi langkah-langkah sebelumnya. Tapi jika sudah sesuai, maka langkah selanjutnya akan dilaksanakan.

Evaluasi *prototype* merupakan proses sistematis yang melibatkan pengguna dalam menilai kesesuaian antara rancangan awal sistem dengan kebutuhan aktual mereka. Pada tahap ini, pengguna memberikan umpan balik berdasarkan interaksi langsung terhadap prototype yang telah dibangun, baik melalui observasi, wawancara, maupun pengisian kuesioner. Evaluasi ini penting untuk

mengidentifikasi potensi kekurangan dalam aspek fungsionalitas, kemudahan penggunaan, maupun efisiensi interaksi pengguna

4) Mengkodekan Sistem

Di tahap ini prototype yang sudah disepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.

Proses ini melibatkan transformasi kebutuhan fungsional dan desain antarmuka ke dalam kode yang dapat dijalankan oleh komputer. Pemilihan bahasa pemrograman, kerangka kerja (framework), serta arsitektur sistem yang tepat sangat penting untuk memastikan efisiensi, skalabilitas, dan kemudahan pemeliharaan sistem

5) Menguji Sistem

Setelah sistem sudah menjadi suatu software yang siap pakai, maka software harus di tes dahulu sebelum digunakan. Hal ini bertujuan untuk meminimalisir kesalahan software tersebut. Pengujian dilakukan dengan Black Box, White box, Pengujian arsitektur, Basis path dan lain-lain.

Pengujian bertujuan untuk mengidentifikasi kesalahan (bug), memastikan keandalan sistem, dan memverifikasi bahwa semua kebutuhan fungsional dan non-fungsional telah terpenuhi sebelum sistem digunakan secara luas.

6) Evaluasi Sistem

Di tahap ini klien mengevaluasi sistem yang sudah dibuat sudah sesuai yang diinginkan. Jika tidak, maka pengembang akan mengulangi langkah ke 4 dan 5. Tapi jika iya, maka langkah ke 7 akan dilakukan.

Jika sistem belum sepenuhnya sesuai, maka pengembang akan kembali ke tahap pengkodean dan pengujian untuk melakukan penyesuaian dan perbaikan. Namun, jika sistem telah dinyatakan sesuai dan layak oleh klien, maka pengembangan dilanjutkan ke tahap berikutnya.

7) Menggunakan Sistem

Software yang telah diuji dan diterima klien siap digunakan. Ini berarti sistem siap digunakan secara operasional oleh pengguna akhir. Selama fase ini, pengembang mungkin masih memberikan dukungan teknis untuk memastikan bahwa sistem berjalan lancar dan menangani masalah apa pun yang mungkin muncul setelah peluncuran.

Pada fase ini, pengembang juga dapat menyediakan layanan pelatihan pengguna serta dukungan teknis untuk menjamin kelancaran operasional dan membantu jika terjadi masalah pasca-implementasi.

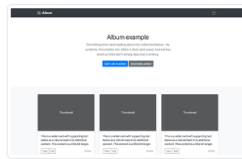
Tahap ini juga dapat mencakup proses monitoring awal untuk menilai performa sistem dalam situasi nyata dan memastikan bahwa sistem dapat memberikan manfaat sebagaimana yang telah direncanakan. Dengan keberhasilan tahap ini, sistem dianggap telah selesai dikembangkan dan siap mendukung aktivitas organisasi atau pengguna secara optimal.

2.2.6 Bootstrap

Bootstrap adalah framework front-end yang digunakan untuk membantu pengembangan tampilan website agar lebih cepat dan konsisten. Bootstrap menyediakan kumpulan komponen antarmuka pengguna (UI) seperti tombol, formulir, navbar, dan layout grid yang sudah didesain responsif dan siap pakai.

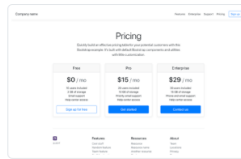
Custom Components

Brand-new components and templates to help folks quickly get started with Bootstrap and demonstrate best practices for adding onto the framework.



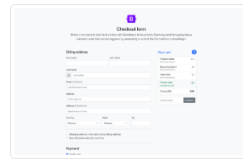
[Album](#)

Simple one-page template for photo galleries, portfolios, and more.



[Pricing](#)

Example pricing page built with Cards and featuring a custom header and footer.



[Checkout](#)

Custom checkout form showing our form components and their validation features.



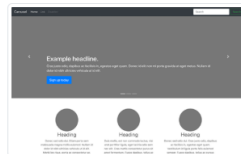
[Product](#)

Lean product-focused marketing page with extensive grid and image work.



[Cover](#)

A one-page template for building simple and beautiful home pages.



[Carousel](#)

Customize the navbar and carousel, then add some new components.



[Blog](#)

Magazine like blog template with header, navigation, featured content.



[Dashboard](#)

Basic admin dashboard shell with fixed sidebar and navbar.

Gambar 2. 3 Komponen Bootstrap

2.2.7 Wireframe

Wireframe adalah representasi visual sederhana dari sebuah antarmuka pengguna (UI) yang digunakan dalam proses desain produk digital, seperti website atau aplikasi mobile. Tujuan utama wireframe adalah untuk menguraikan struktur, tata letak, dan hierarki informasi tanpa fokus pada elemen visual seperti warna, tipografi, atau gambar. Wireframe biasanya dibuat dalam bentuk sketsa hitam putih atau abu-abu, menggunakan bentuk dan garis sederhana untuk mewakili komponen seperti tombol, teks, gambar, dan menu. Dengan demikian, wireframe berfungsi sebagai cetak biru atau kerangka dasar yang membantu desainer dan stakeholder memahami alur dan fungsionalitas produk sebelum masuk ke tahap desain yang lebih detail.

Wireframe juga berfungsi sebagai alat komunikasi yang efektif antara desainer, developer, dan stakeholder. Dengan menyajikan struktur dan alur produk secara visual, wireframe memudahkan semua pihak untuk memberikan

masuk dan mengidentifikasi area yang perlu diperbaiki sebelum proses pengembangan dimulai. Selain itu, wireframe dapat digunakan untuk melakukan pengujian usability awal, di mana pengguna potensial diminta untuk mengevaluasi alur navigasi dan fungsionalitas dasar. Umpan balik dari pengujian ini dapat digunakan untuk menyempurnakan desain sebelum melanjutkan ke tahap berikutnya. Ada dua jenis wireframe yang umum digunakan yaitu, *low-fidelity* dan *high-fidelity*. Wireframe *low-fidelity* biasanya dibuat dengan cepat menggunakan alat sederhana seperti kertas dan pensil atau aplikasi desain dasar. Jenis ini cocok untuk brainstorming awal dan diskusi cepat dengan tim. Sementara itu, wireframe *high-fidelity* lebih detail dan sering dibuat menggunakan alat desain digital seperti Figma[11].

2.2.8 Usability

Usability berasal dari kata *usable* yang berarti dapat digunakan dengan baik. Usability adalah tingkat kualitas dari sistem yang mudah dipelajari, mudah digunakan dan mendorong pengguna untuk menggunakan sistem sebagai alat bantu positif dalam menyelesaikan tugas. Usability merupakan kombinasi berbagai faktor antara lain: perancangan yang intuitif, kemudahan dipelajari, efisiensi penggunaan atau seberapa cepat pengguna dapat menyelesaikan tugasnya, kemudahan diingat, tingkat kesalahan pengguna, dan kepuasan pengguna secara subyektif. *International Standard Organization* (ISO) mendefinisikan usability sebagai sejauh mana suatu produk dapat digunakan oleh pengguna tertentu berdasarkan efektivitas, efisiensi dan kepuasan dalam konteks penggunaan guna mencapai tujuan yang ditentukan[12].

2.2.9 Usability Testing

Usability testing adalah metode evaluasi sistematis yang digunakan untuk mengukur sejauh mana suatu produk, sistem, atau antarmuka pengguna dapat digunakan secara efektif, efisien, dan memuaskan oleh pengguna akhir. Tujuan utama dari *usability testing* adalah mengidentifikasi masalah yang mungkin menghambat pengalaman pengguna atau *user experience* dan memberikan rekomendasi untuk perbaikan. Metode ini melibatkan pengamatan langsung terhadap interaksi pengguna dengan produk dalam skenario yang terkontrol atau alami, sehingga memungkinkan peneliti untuk mengumpulkan data kualitatif dan kuantitatif tentang kinerja produk. Dengan demikian, usability testing menjadi komponen kritis dalam proses desain yang berpusat pada pengguna[13].

Proses *usability testing* biasanya melibatkan beberapa tahapan, mulai dari perencanaan, rekrutmen peserta, pelaksanaan tes, hingga analisis hasil. Pada tahap perencanaan, peneliti menentukan tujuan tes, merancang skenario tugas yang relevan, dan memilih metrik evaluasi seperti tingkat keberhasilan tugas, waktu penyelesaian, dan tingkat kepuasan pengguna. Peserta tes direkrut berdasarkan karakteristik yang mewakili pengguna target, sehingga hasil tes dapat memberikan gambaran yang akurat tentang bagaimana produk akan digunakan dalam konteks nyata. Selama pelaksanaan tes, peserta diminta untuk menyelesaikan serangkaian tugas sementara peneliti mengamati dan mencatat perilaku, kesulitan, serta umpan balik verbal dari peserta[14].

Data yang dikumpulkan dari usability testing dianalisis untuk mengidentifikasi pola masalah yang muncul, seperti kesulitan navigasi,

kebingungan dalam memahami antarmuka, atau ketidakefisienan dalam menyelesaikan tugas. Hasil analisis ini kemudian digunakan untuk merumuskan rekomendasi perbaikan yang spesifik dan dapat ditindaklanjuti. Misalnya, jika peserta sering kali gagal menemukan tombol penting, peneliti dapat merekomendasikan perubahan tata letak atau penambahan petunjuk visual.

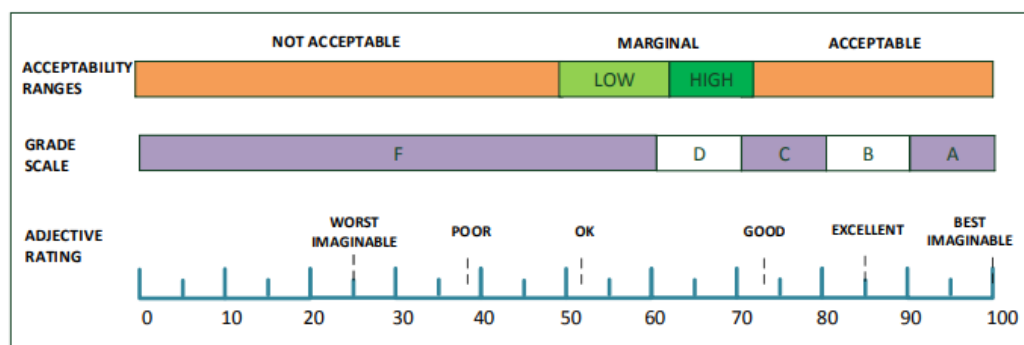
2.2.10 System Usability Scale

System Usability Scale (SUS) adalah instrumen pengukuran kuantitatif yang digunakan untuk mengevaluasi persepsi pengguna terhadap usability suatu sistem atau produk. Dikembangkan oleh John Brooke pada tahun 1986, SUS dirancang sebagai metode yang cepat, sederhana, namun andal untuk mengukur tingkat kegunaan suatu sistem berdasarkan pengalaman pengguna. Instrumen ini terdiri dari sepuluh pernyataan yang direspons oleh pengguna menggunakan skala Likert lima poin, mulai dari sangat tidak setuju hingga sangat setuju. Skor SUS yang dihasilkan kemudian dihitung dan diinterpretasikan untuk memberikan gambaran tentang seberapa mudah atau sulitnya sistem digunakan[15].

Proses penggunaan SUS dimulai dengan pengumpulan data melalui kuesioner yang diberikan kepada pengguna setelah mereka berinteraksi dengan sistem. Sepuluh pernyataan dalam SUS mencakup aspek-aspek usability seperti kemudahan penggunaan, konsistensi, kompleksitas, dan kepercayaan diri pengguna dalam menggunakan sistem. Setelah pengguna memberikan respons, skor dihitung dengan mengikuti formula khusus yang mengkonversi respons kualitatif menjadi nilai kuantitatif. Skor SUS berkisar antara 0 hingga 100, di mana skor yang lebih tinggi menunjukkan tingkat usability yang lebih baik. Skor

rata-rata SUS umumnya berada di sekitar 68, yang dianggap sebagai titik acuan untuk menilai apakah suatu sistem memiliki usability yang memadai.

Adapun cara dalam melakukan evaluasi *System Usability Scale* adalah dengan menggunakan konsep *Acceptability*, *Grade Scale*, *Adjective Rating*, dan *SUS Percentile*



Gambar 2. 4 Penilaian berdasarkan Acceptability, Grade Scale, Adjective Rating

Acceptability, Grade Scale, Adjective Rating, dan SUS Percentile Rank adalah konsep dan metrik yang sering digunakan dalam evaluasi usability untuk menginterpretasikan dan mengkomunikasikan hasil pengukuran kegunaan suatu sistem.

System Usability Scale (SUS) merupakan metode evaluasi usability yang efisien karena mempertimbangkan aspek waktu, biaya, dan jumlah sampel yang relatif kecil, namun tetap mampu menghasilkan data yang valid dan reliabel. Keunggulan utama SUS terletak pada kemampuannya memberikan hasil yang memadai dengan sumber daya terbatas.

Dalam implementasinya, responden memberikan tanggapan melalui skala *Likert* lima poin yang mencakup Sangat Setuju, Setuju, Netral, Tidak Setuju dan Sangat Tidak Setuju. Untuk menghitung skor SUS, digunakan rumus berikut:

$$\text{Skor SUS} = [(Q1-1) + (5-Q2) + (Q3-1) + (5-Q4) + (Q5-1) + (5-Q6) + (Q7-1) + (5-Q8) + (Q9-1) + (5-Q10)] \times 2.5$$

Dimana :

Q = Pertanyaan

1. Untuk pernyataan (Q) bernomor ganjil (positif), kurangi nilai respons dengan 1
2. Untuk pernyataan (Q) bernomor genap (negatif), kurangi 5 dari nilai respons
3. Jumlahkan seluruh nilai yang telah dikonversi
4. Kalikan total skor dengan 2,5 untuk mendapatkan nilai akhir SUS (skala 0-100)

Setelah menerapkan rumus *System Usability Scale* dalam perhitungan, langkah berikutnya adalah menentukan nilai rata-rata berdasarkan seluruh respons yang diperoleh dari kuesioner. Nilai tersebut kemudian diinterpretasikan sesuai dengan standar pengukuran *System Usability Scale*. Hasil perhitungan selanjutnya diklasifikasikan berdasarkan kriteria *Acceptability Ranges*.

Tabel 2. 2 Skor System Usability Scale

<i>SUS Score</i>	<i>Grade</i>	<i>Adjective Rating</i>
> 80.3	A	<i>Excellent</i>
68 – 80.3	B	<i>Good</i>
68	C	<i>OK</i>
51 – 67	D	<i>Poor</i>
< 51	F	<i>Awful</i>

Skor *System Usability Scale* atau *SUS Score* dapat dikategorikan ke dalam beberapa tingkatan berdasarkan rentang nilai tertentu pada tabel diatas. Skor di atas 80,3 memperoleh predikat *Excellent* dengan kualitas mutu tertinggi yang dilambangkan sebagai *Grade A*. Rentang skor antara 68 hingga 80,3 termasuk dalam kategori *Good* dan diberi *Grade B*. Skor tepat 68 diklasifikasikan sebagai *Okay* dengan *Grade C*. Apabila skor berada pada kisaran 51 hingga 68, tingkat *usabilitas* dinilai *Poor* dan termasuk *Grade D*. Sementara itu, skor di bawah 51 menandakan tingkat *usabilitas* yang sangat rendah atau *Awful*, sehingga diberi *Grade E*

1) Acceptability

Acceptability mengacu pada tingkat penerimaan pengguna terhadap suatu sistem berdasarkan pengalaman mereka. Konsep ini tidak hanya mencakup aspek *usability*, tetapi juga faktor-faktor seperti kepuasan, kegunaan, dan kesesuaian sistem dengan kebutuhan pengguna [16]. Acceptability sering diukur melalui kombinasi metrik kuantitatif, seperti skor *System Usability Scale* (SUS), dan umpan balik kualitatif dari pengguna.

2) Grade Scale

Grade Scale adalah metode interpretasi skor *usability* yang mengkategorikan hasil pengukuran ke dalam tingkatan atau grade, mirip dengan sistem penilaian akademis [17]. Misalnya, skor SUS dapat dikonversi ke dalam grade seperti A, B, C, D, atau F, di mana A menunjukkan *usability* yang sangat baik dan F

menunjukkan usability yang buruk. Grade Scale memudahkan stakeholder untuk memahami hasil evaluasi usability tanpa harus mendalami detail teknis [18].

3) Adjective Rating

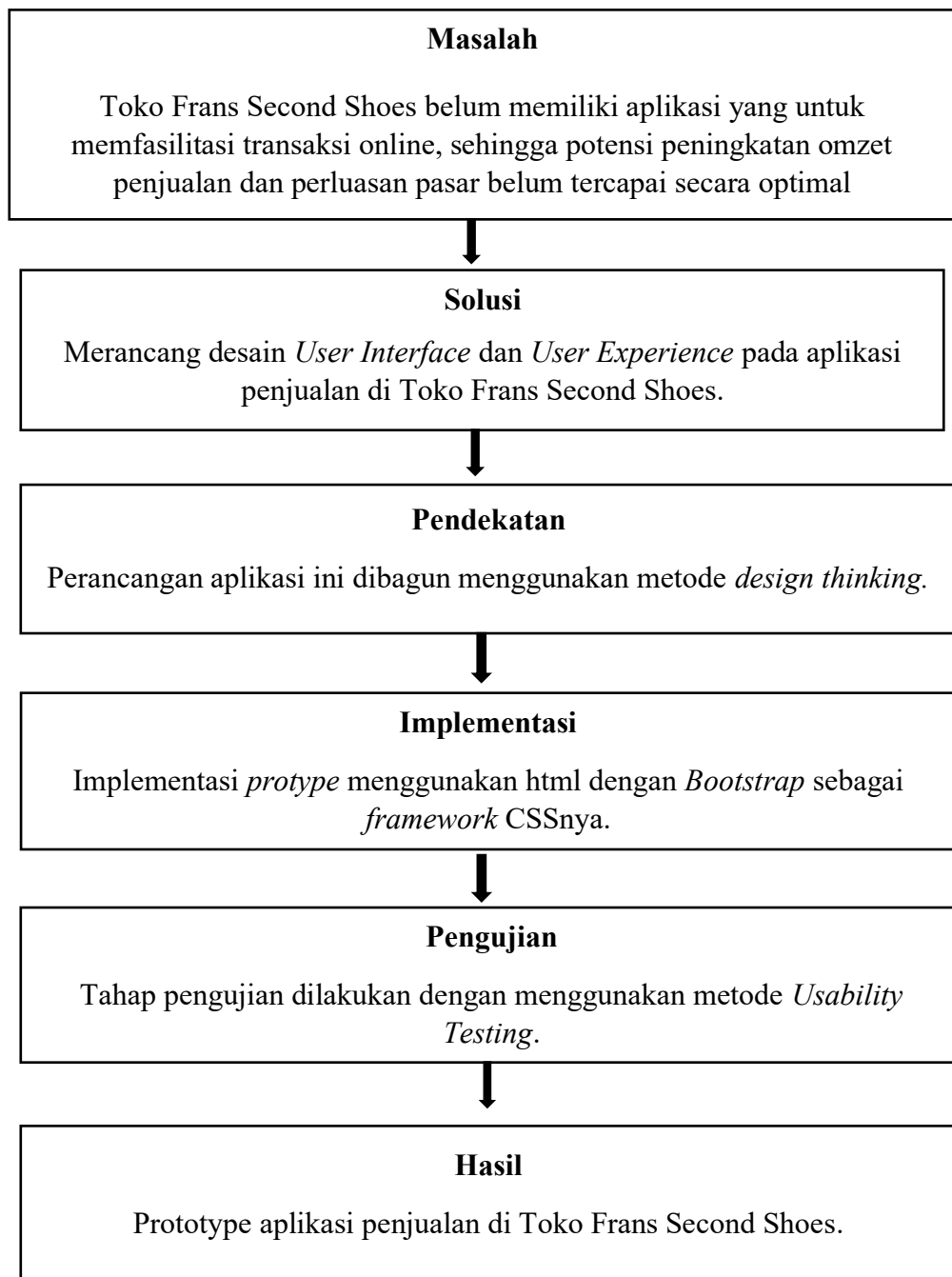
Adjective Rating adalah pendekatan lain untuk menginterpretasikan skor *usability* dengan mengaitkannya dengan deskriptor kualitatif, seperti baik, cukup, atau buruk [19]. Misalnya, skor SUS tertentu dapat diklasifikasikan sebagai *acceptable* atau *marginal* berdasarkan rentang nilai yang telah ditentukan. *Adjective Rating* membantu menyederhanakan komunikasi hasil evaluasi kepada audiens non-teknis

4) SUS Percentile

SUS Percentile Rank adalah metrik yang membandingkan skor SUS suatu sistem dengan distribusi skor SUS dari sistem lain yang telah diuji sebelumnya [20]. Metrik ini memberikan konteks yang lebih luas dengan menunjukkan bagaimana performa sistem tersebut relatif terhadap sistem lain dalam basis data yang relevan. Misalnya, jika suatu sistem memiliki SUS Percentile Rank 85%, artinya sistem tersebut lebih baik dari 85% sistem yang pernah diuji. Metrik ini sangat berguna untuk benchmarking dan menetapkan target usability yang realistis[21].

2.3 Kerangka Pikir

Kerangka pikir adalah suatu dasar yang mencakup penggabungan antara teori, fakta, observasi dan kajian Pustaka, yang akan dilakukan dapat digambarkan seperti dibawah ini:



Gambar 2. 5 Kerangka Pikir