

BAB II TINJAUAN PUSTAKA

A. Penelitian Terkait

Penelitian terkait merupakan serangkaian studi terdahulu yang dikumpulkan untuk mendukung kredibilitas riset, menyediakan acuan metodologis, serta memastikan tidak adanya replikasi hasil. Bagian ini juga berperan dalam menstimulasi penemuan baru atau pengembangan teori melalui perspektif ilmiah yang berbeda. Daftar penelitian yang menjadi rujukan dalam penelitian ini dirangkum pada tabel di bawah ini:

Tabel 1. Penelitian Terkait

No	Nama (Tahun)	Keterangan Penelitian
(1)	(2)	(3)
1	Heru Saputra et al., 2024	Judul: Pemanfaatan Aplikasi <i>E-Commerce</i> Menggunakan Metode <i>Agile</i> pada Usaha Kue dan Makanan Minang Kreatif Metode: <i>Agile</i> Software: PHP, MySQL Hasil Penelitian: <i>E-commerce</i> meningkatkan efisiensi dan jangkauan pasar UMKM di Indonesia, termasuk Usaha Kue dan Makanan Minang Kreatif. Dengan adopsi metode <i>Agile</i> dalam pengembangan aplikasi <i>e-commerce</i> , penelitian ini menunjukkan bahwa aplikasi tersebut dapat membantu mengatasi tantangan bisnis, memperluas pasar, dan meningkatkan penjualan. Pengujian perangkat lunak membuktikan bahwa aplikasi berfungsi dengan baik, memberikan solusi yang lebih efisien dan akurat dibandingkan dengan metode penjualan tradisional. Hasilnya, Usaha Kue dan Makanan Minang Kreatif dapat bersaing lebih baik di pasar, meningkatkan kualitas layanan, dan berkontribusi pada peningkatan ekonomi lokal.
2	Amani et al. n.d., 2025	Judul: Rancang Bangun Sistem Informasi Penjualan dan Pemasaran Bisnis Kuliner <i>Catering</i> Berbasis WEB Menggunakan

	<i>Woocommerce</i> (Studi Kasus Dapur Ummi Iis Tasikmalaya) Metode: <i>Research and Development</i> (R&D) dengan pendekatan Agile Software Development. Software: <i>WordPress, WooCommerce, Elementor, MySQL/MariaDB.</i> Hasil Penelitian: Penelitian ini berhasil mengembangkan sistem informasi penjualan dan pemasaran berbasis web untuk Dapur Ummi Iis Tasikmalaya menggunakan WooCommerce. Sistem ini terbukti mampu mengotomatisasi proses penjualan, menyediakan fitur pemesanan online, manajemen produk, serta pemasaran digital, sehingga meningkatkan efisiensi operasional dan kepuasan pelanggan.
3	Setiawan et al., 2024 Judul: Rancang Bangun Website Store Management System Laravel dengan Metode Agile: Studi Kasus UMKM Toko Jali Metode: <i>Agile</i> Software: <i>Laravel</i> Hasil Penelitian: Sistem manajemen toko situs web berhasil mengotomatiskan dan mengintegrasikan proses bisnis penting untuk Toko Jali, menunjukkan efektivitasnya dalam meningkatkan efisiensi operasional, produktivitas, dan kepuasan pelanggan, dan berfungsi sebagai model bagi UMKM lainnya.
4	Manuel Bayu Krisnamurti et al., 2025 Judul: Rancang Bangun Aplikasi <i>Front-End</i> Penjualan Berbasis Web Menggunakan Metode <i>Agile</i>. Metode: <i>Agile</i> Software: PHP, MySQL Hasil Penelitian: Aplikasi front-end penjualan berbasis web untuk Butik April mampu memberikan solusi terhadap permasalahan sistem

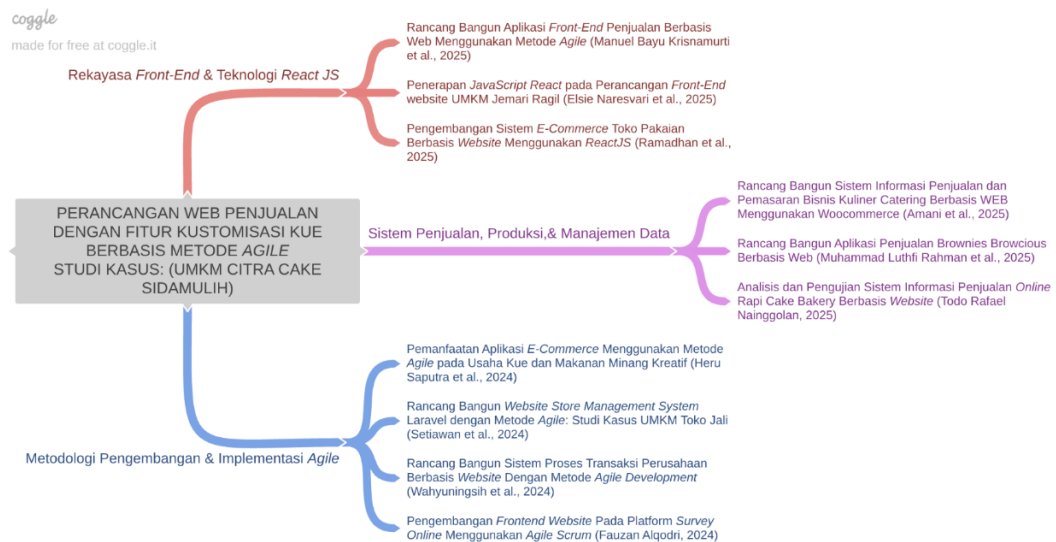
		transaksi yang sebelumnya dilakukan secara manual melalui WhatsApp. Aplikasi ini memudahkan pengguna dalam melihat katalog produk, menambahkan barang ke keranjang, melakukan <i>checkout</i>, serta memberikan bukti pembayaran secara langsung melalui sistem. Selain itu, fitur pengelolaan data barang, validasi pembelian oleh admin, dan pembuatan laporan transaksi penjualan turut mendukung kelancaran proses bisnis butik.
5	Wahyuningsih et al. n.d., 2024	Judul: Rancang Bangun Sistem Proses Transaksi Perusahaan Berbasis <i>Website</i> Dengan Metode <i>Agile Development</i> Metode: <i>Agile Development (ASD)</i> Software: <i>React JS</i> Hasil Penelitian: Metode <i>Adaptive Software Development (ASD)</i> dinilai baik untuk digunakan pada sistem yang mengharuskan beradaptasi dan menyesuaikan dengan kebutuhan klien. Metode <i>Adaptive Software Development (ASD)</i> juga dinilai dapat membangun motivasi dan komunikasi dalam tim yang baik. Sistem telah dikembangkan dengan baik serta dapat memberikan progres yang lebih cepat apabila dikerjakan melalui tim.
6	Elsie Naresvari et al., 2025	Judul: Penerapan <i>JavaScript React</i> pada perancangan <i>front-end</i> website UMKM Jemari Ragil Metode: <i>Waterfall</i> Software: <i>JavaScript React</i> Hasil Penelitian: Berhasil mengimplementasikan <i>JavaScript React</i> untuk membangun sebuah tampilan antarmuka website UMKM yang efisien dan interaktif. Hasil pengujian sistem <i>Usability Testing</i> dan diukur menggunakan <i>System Usability Scale (SUS)</i> yang menghasilkan skor 76 dengan kategori baik, menunjukkan bahwa sistem dapat diterima oleh pengguna

		dan memenuhi standar <i>usability</i> yang ditetapkan.
7	Muhammad Luthfi Rahman et al., 2025	Judul: Rancang Bangun Aplikasi Penjualan Brownies Browcious Berbasis Web Metode: SDLC <i>Waterfall</i> Software: <i>React.js</i>, dan <i>cloud database MongoDB</i> Hasil Penelitian: Berhasil memenuhi tujuan untuk mempermudah proses transaksi online bagi konsumen. Aplikasi ini menawarkan pengalaman pengguna yang optimal, manajemen data, serta keamanan dan skalabilitas yang tinggi.
8	Fauzan Alqodri, 2024	Judul: Pengembangan <i>Frontend Website</i> Pada <i>Platform Survey Online</i> Menggunakan <i>Agile Scrum</i> Metode: <i>Agile Scrum</i> Software: <i>React JS</i> Hasil Penelitian: Penerapan <i>Agile Scrum</i> dalam pengembangan <i>frontend</i> dengan <i>React.js</i> dapat meningkatkan efisiensi tim pengembang, meningkatkan kualitas estetika situs web, serta memastikan pembaruan informasi secara berkala. Namun, tantangannya meliputi kesulitan dalam memperkirakan waktu yang dibutuhkan untuk setiap fitur dalam satu iterasi secara akurat dan mempertahankan tingkat disiplin tinggi yang dibutuhkan dari tim.
9	Riki, M. Nainggolan, 2025	Judul: Analisis dan Pengujian Sistem Informasi Penjualan <i>Online</i> Rapi Cake <i>Bakery</i> Berbasis <i>Website</i> Metode: <i>Agile</i> Software: PHP, MySQL Hasil Penelitian: Toko <i>online</i> RAPI-CAKE <i>BAKERY</i> berhasil dibangun sesuai dengan kebutuhan pengguna. Sistem ini menyediakan fitur-fitur utama seperti manajemen produk,

	manajemen <i>customer</i>, proses produksi, pengelolaan <i>inventory</i>, transaksi penjualan, serta berbagai laporan (penjualan, profit, omset, pembatalan, <i>inventory</i>, produksi), dan dilengkapi dengan fitur cetak PDF serta <i>export</i> Excel. Antarmuka yang <i>user-friendly</i> memudahkan pengguna dalam mengakses seluruh fungsi sistem, baik dari sisi admin maupun <i>customer</i>. Hasil pengujian menggunakan metode <i>Black Box Testing</i> menunjukkan bahwa seluruh fitur berjalan stabil, <i>output</i> sesuai dengan yang diharapkan, dan mudah digunakan oleh pengguna akhir. Dengan demikian, sistem ini dapat membantu menyelesaikan permasalahan yang telah diidentifikasi, yaitu digitalisasi proses penjualan, produksi, dan administrasi toko roti secara terintegrasi.
10 Ramadhan et al., 2025	Judul: Pengembangan Sistem E-Commerce Toko Pakaian Berbasis Website Menggunakan <i>ReactJS</i> Metode: <i>Waterfall</i> Software: <i>React JS</i> Hasil Penelitian: Hasil penelitian ini adalah <i>platform e-commerce</i> yang mempermudah transaksi, pengelolaan produk, dan manajemen pelanggan, sehingga meningkatkan efisiensi bisnis serta jangkauan pasar secara digital dengan baik dan efektif sekali, serta memberikan kemudahan bagi pelanggan dalam berbelanja <i>online</i> dengan lebih mudah dan cepat.

Penelitian ini mengadopsi keberhasilan metode *Agile* dan teknologi *React JS* dari referensi di atas, namun memberikan nilai tambah pada fitur kustomisasi kue ulang tahun yang dinamis serta transparansi progres pesanan. Berbeda dengan penelitian sebelumnya yang mayoritas fokus pada penjualan produk statis, penelitian ini akan lebih mendalam dalam mengoptimalkan *state management* pada *front-end* untuk menangani kustomisasi kompleks, yang kemudian divalidasi dengan pengujian *black box testing* dan *customer evaluation* sebagai tahap akhir

dari tahapan *review* untuk memastikan solusi yang diberikan benar-benar memudahkan pelanggan Citra Cake.



Gambar 1. Mind Map Penelitian Terkait

B. Landasan Teori

1. Transformasi Digital pada UMKM

Transformasi digital merupakan proses perubahan mendasar dalam cara organisasi menciptakan nilai melalui pemanfaatan teknologi digital secara menyeluruh, yang tidak hanya terbatas pada otomatisasi proses bisnis, tetapi juga mencakup perubahan model bisnis, budaya organisasi, dan pengalaman pelanggan secara holistik [23]. Dalam konteks Usaha Mikro, Kecil, dan Menengah (UMKM), transformasi digital didefinisikan sebagai adopsi dan integrasi teknologi informasi dan komunikasi (TIK) ke dalam seluruh aspek operasional usaha dengan tujuan meningkatkan efisiensi, produktivitas, dan daya saing di pasar yang semakin terhubung secara digital [24].

2. Sistem Informasi Penjualan Berbasis *Website*

Sistem informasi penjualan merupakan sebuah susunan terorganisir yang mengintegrasikan sumber daya manusia, prosedur kerja, perangkat lunak, dan basis data guna mengelola seluruh aktivitas transaksi serta pemasaran secara sistematis [25]. Dalam era ekonomi digital, sistem informasi penjualan telah bertransformasi dari sekadar alat pencatat transaksi konvensional menjadi platform strategis yang mampu mengotomatisasi aliran data pesanan, pengelolaan stok, hingga pelaporan keuangan secara akurat [26]. Implementasi sistem ini pada UMKM seperti Citra Cake berfungsi sebagai tulang punggung operasional yang menghilangkan hambatan administrasi manual, sehingga

setiap permintaan pelanggan dapat diproses dengan standar yang konsisten, transparan, dan terukur.

Pemilihan platform berbasis *website* dalam pengembangan sistem ini dilakukan dengan mempertimbangkan aspek aksesibilitas serta efisiensi jangkauan pasar yang lebih luas dibandingkan aplikasi *mobile* konvensional atau perangkat lunak desktop. *Website* bersifat *platform independent*, sehingga pelanggan dapat mengakses fitur melalui berbagai perangkat seperti *smartphone*, tablet, maupun komputer tanpa harus melalui proses instalasi yang menguras memori penyimpanan perangkat pengguna [23]. Selain itu, dari sisi pemeliharaan, sistem berbasis *website* memudahkan pengelola UMKM untuk melakukan pembaruan konten katalog atau harga secara *real-time* tanpa perlu menunggu persetujuan distribusi dari *app store* pihak ketiga [27]. Dengan demikian, *website* menjadi solusi yang paling fleksibel dan ekonomis untuk menjangkau basis pelanggan yang heterogen sekaligus mendukung siklus pengembangan *Agile* yang iteratif.

3. Fitur Kustomisasi Kue

Dalam ekosistem bisnis modern, strategi kustomisasi produk atau *mass customization* telah bertransformasi menjadi salah satu instrumen inovasi paling efektif untuk menciptakan nilai tambah yang membedakan sebuah merek dari kompetitornya [28]. Fitur kustomisasi kue ulang tahun pada platform Citra Cake bukan sekadar tambahan fungsionalitas, melainkan sebuah inovasi layanan yang menjembatani batasan antara imajinasi pelanggan dan kapabilitas produksi produsen. Melalui fitur ini, peran pelanggan bergeser dari sekadar konsumen pasif menjadi partisipan aktif atau *co-creator* yang memiliki kebebasan untuk mempersonalisasi produk mereka sendiri, mulai dari penentuan komponen *base cake*, pemilihan varian rasa, hingga pemilihan elemen dekoratif seperti warna krim dan jenis *topping* secara mendetail. Dengan adanya fitur kustomisasi yang terintegrasi, UMKM dapat menawarkan pengalaman belanja yang lebih *user-centric* dan modern [29].

4. Antarmuka Pengguna (*Front-End*)

Antarmuka pengguna atau yang secara teknis dikenal sebagai *front-end* merupakan representasi visual dan gerbang utama interaksi yang menghubungkan pengguna akhir (*end-user*) dengan kompleksitas logika sistem di sisi *server* [30]. Dalam pengembangan aplikasi web modern, *front-end* tidak lagi dipandang sekadar sebagai lapisan estetika statis, melainkan sebuah entitas dinamis yang mengintegrasikan prinsip desain visual, aksesibilitas, dan performa teknis [31].

Tujuan dari implementasi *front-end* adalah menyajikan data dari *back-end* sebagai antarmuka yang responsif dan ramah pengguna (*user-friendly*)

[32]. Dalam sistem informasi penjualan pada skala UMKM, peran *front-end* menjadi sangat strategis karena bertindak sebagai etalase digital yang bertanggung jawab untuk menyajikan informasi produk secara intuitif, mengelola validasi *input* pelanggan secara langsung, serta menjamin alur navigasi yang mulus tanpa hambatan teknis. Dengan demikian, kualitas perancangan *front-end* menjadi faktor penentu dalam menciptakan pengalaman pengguna yang optimal serta efektivitas proses transaksi digital [30].

5. React JS

React JS merupakan pustaka *JavaScript* yang sangat efektif yang dirancang khusus untuk tujuan membangun antarmuka web dinamis, terbukti sangat menguntungkan di bidang *e-commerce* dan fitur rumit yang terkait dengan kustomisasi produk, memungkinkan pengembang untuk menciptakan pengalaman pengguna yang lebih interaktif dan menarik [33]. Dalam konteks studi khusus ini, React JS secara strategis digunakan untuk menerapkan sistem manajemen status yang kuat dan efisien yang memungkinkan penanganan dinamis fitur kustomisasi kue ulang tahun, memastikan pengalaman pengguna yang mulus yang menyesuaikan dengan kebutuhan individu setiap pengguna. Fokus utamanya adalah mengubah aset desain fungsional menjadi aplikasi web yang responsif dan siap untuk lingkungan produksi [34].

6. Metodologi SDLC dengan Model *Agile Development*

Software Development Life Cycle (SDLC) adalah kerangka kerja sistematis yang digunakan oleh pengembang perangkat lunak untuk merancang, membangun, dan memelihara aplikasi berkualitas tinggi [35]. *Agile Development* dipilih sebagai sebuah metodologi pengembangan yang bersifat adaptif, iteratif, dan inkremental [36]. Berbeda dengan model tradisional seperti *Waterfall* yang bersifat linier dan kaku, *Agile* menitikberatkan pada kelincahan dalam merespons perubahan kebutuhan pengguna yang sering kali dinamis selama proses pengembangan berlangsung. Berdasarkan metode *System Development Live Cycle* (SDLC), *Agile Development* dilakukan melalui beberapa tahapan sebagai berikut:

a. Plan

Tahapan ini merupakan fondasi awal untuk membangun ide dan konsep mengenai sistem yang akan dibuat. Pada tahapan ini peneliti melakukan identifikasi permasalahan yang ada, menentukan prioritas fitur, serta mempertimbangkan solusi teknologi yang paling efektif untuk mencapai tujuan [37].

b. *Design*

Pada tahap ini, ide dan konsep yang telah terkumpul dalam fase perencanaan divisualisasikan ke dalam bentuk *mockup*. Perancangan

difokuskan pada antarmuka pengguna (*User Interface*) dan pengalaman pengguna (*User Experience*) untuk memastikan fitur kustomisasi kue bersifat intuitif dan mudah dipahami sebelum masuk ke tahap pengkodean [38].

c. *Develop*

Develop merupakan tahapan eksekusi di mana rancangan desain diubah ke dalam bahasa pemrograman untuk memastikan solusi yang diusulkan selaras dengan kebutuhan fungsional dan estetika [39].

d. *Test*

Test merupakan tahap pengujian perangkat lunak untuk memastikan kode program berjalan sesuai dengan spesifikasi fungsional [10]. Proses ini bertujuan untuk mendeteksi *bug* atau ketidaksesuaian logika sebelum sistem dilanjutkan ke tahap implementasi, sehingga menjamin kualitas performa saat diakses oleh pelanggan [11]. Sebagai komponen penting dari manajemen kualitas, pengujian terintegrasi secara mendalam ke dalam siklus pengembangan untuk menilai apakah perangkat lunak yang dibangun telah memenuhi kebutuhan yang ditentukan serta meningkatkan kepercayaan bahwa sistem dapat berjalan sesuai dengan harapan [40].

Dalam pengembangan *front-end* menggunakan React JS, pengujian menjadi krusial untuk menjamin antarmuka berjalan dengan stabil dan responsif bagi pengguna [41]. Metode yang umum digunakan untuk menguji fungsionalitas antarmuka dari sudut pandang pengguna luar adalah *Black-box Testing*. Pengujian ini berfokus pada fungsionalitas antarmuka dari sudut pandang pengguna tanpa perlu melihat struktur kode internal, rincian implementasi, maupun jalur logika di dalamnya [42]. Sistem dipandang sebagai sebuah "kotak hitam", di mana pengujian dilakukan dengan memberikan *input* tertentu pada elemen antarmuka dan mengamati *output* yang dihasilkan guna memastikan kesesuaian dengan spesifikasi kebutuhan yang telah didefinisikan [43].

Dalam konteks pengembangan *front-end*, *Black-box Testing* difokuskan untuk memvalidasi bagaimana komponen-komponen antarmuka merespons aksi pengguna serta bagaimana data yang diterima dari layanan luar ditampilkan pada layar. Pengujian ini mencakup pengamatan terhadap fungsi komponen interaktif, elemen masukan data (*form input*), hingga ketepatan navigasi antarhalaman [44].

Fokus utama dari pengujian ini adalah mengidentifikasi berbagai kategori kesalahan, seperti fungsi komponen antarmuka yang tidak berjalan, kesalahan pada elemen visual (*interface errors*), hingga kegagalan sistem dalam menampilkan data eksternal. Melalui pengujian berbagai skenario masukan data yang valid maupun tidak valid,

pengujian ini dapat memastikan ketangguhan antarmuka dalam menangani kesalahan (*error handling*) guna menjaga kenyamanan interaksi pengguna [45].

e. *Deploy*

Tahapan ini merupakan penyatuan seluruh perancangan yang telah dibangun ke dalam lingkungan *hosting* atau *server* produksi agar sistem dapat diakses secara utuh oleh pengguna [46].

f. *Review*

Tahap ini merupakan evaluasi akhir di mana pengembang menunjukkan hasil kerja kepada pemangku kepentingan (*stakeholder*) atau pelanggan. Hasil dari tahap ini menjadi dasar untuk memutuskan apakah pengembangan berlanjut ke iterasi perencanaan berikutnya atau selesai [39]. Umpan balik yang diperoleh dari para pemangku kepentingan tersebut kemudian divalidasi lebih lanjut melalui proses *customer evaluation* untuk mendapatkan perspektif langsung dari sisi pengguna akhir.

Customer evaluation merupakan proses verifikasi akhir yang melibatkan pengguna untuk memastikan bahwa sistem yang dikembangkan telah memenuhi kriteria penerimaan dan kebutuhan fungsional [47]. Dalam pengujian antarmuka pengguna (*front-end*), evaluasi ini difokuskan pada aspek *usability* guna mengukur sejauh mana sistem dapat digunakan dengan mudah dan efisien [10].

Usability merupakan karakteristik kualitas yang menilai seberapa mudah pengguna dapat menggunakan antarmuka seperti aplikasi atau *website* [48]. Untuk mengukurnya, digunakan instrumen *System Usability Scale* (SUS) yang diperkenalkan oleh John Brooke pada tahun 1986 [49]. SUS merupakan sebuah skala *usability* yang reliabel, efisien secara biaya, dan dapat digunakan untuk mengevaluasi sebuah sistem secara global guna melihat kemudahan serta mengukur ketercapaian tujuan penggunaan aplikasi [14].

Pemilihan SUS sebagai instrumen landasan evaluasi didasarkan pada keunggulannya yang bersifat *technology-independent*, sehingga tetap valid digunakan untuk menguji platform modern berskala interaktif tinggi seperti React JS [50]. Selain itu, SUS memiliki tingkat reliabilitas yang tinggi dan mampu menghasilkan kesimpulan data yang valid serta stabil, meskipun diuji dengan ukuran sampel atau jumlah responden yang relatif terbatas.

Untuk menghasilkan evaluasi yang terukur, digunakan instrumen *System Usability Scale* (SUS) menggunakan kuesioner standar yang terdiri dari 10 butir pernyataan dengan skala Likert 5 poin, mulai dari nilai 1 sangat tidak setuju hingga 5 sangat setuju untuk menilai tingkat kepuasan pengguna secara kuantitatif [51]. Seluruh

pernyataan tersebut disusun secara bergantian antara kalimat positif pada nomor ganjil dan kalimat negatif pada nomor genap guna meminimalisir bias respons dari responden.

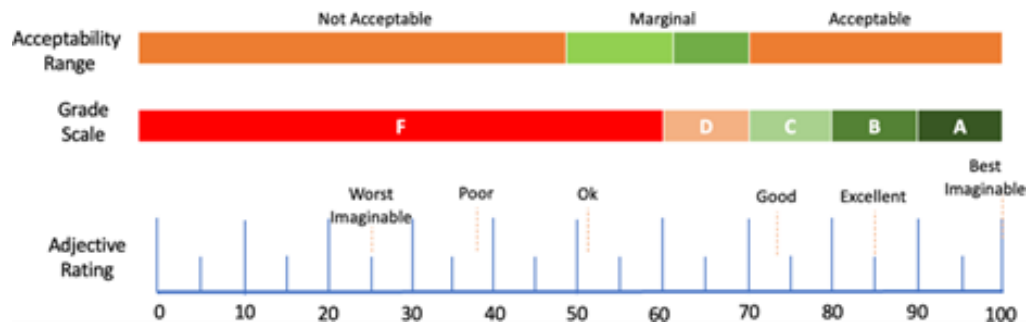
Dalam metode SUS, terdapat lima aspek pengukuran *usability* yang harus dimiliki, antara lain[48]:

- 1) *Learnability*, artinya *website* yang dibuat mudah dipelajari dan dimengerti oleh pengguna.
- 2) *Efficiency*, artinya *website* yang dibuat dapat meningkatkan kinerja pengguna.
- 3) *Memorability*, artinya *website* harus mudah diingat. Apabila pengguna tidak mengakses *website* dalam jangka waktu tertentu, pengguna tetap ingat dan dapat menggunakan semua fitur-fitur aplikasi.
- 4) *Errors*, artinya *website* memiliki tingkat kesalahan yang sekecil mungkin sehingga jika pengguna melakukan kesalahan maka dapat memperbaikinya dengan mudah.
- 5) *Satisfaction*, artinya *website* nyaman untuk digunakan oleh pengguna.

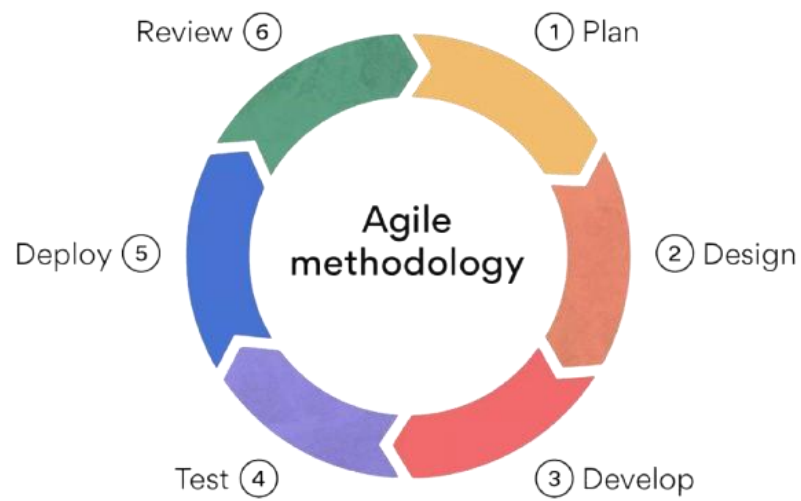
Proses penilaian SUS menggunakan kriteria yang mencakup *acceptability range*, *grade scale*, dan *adjective rating*. *Acceptability range* menunjukkan tingkat penerimaan *end user* terhadap aplikasi atau *website* yang dinilai [14]. Pada kriteria ini terdapat tiga kategori yaitu *not acceptable* apabila diperoleh skor SUS dengan rentang 0-50, marginal jika skor SUS 5-70, dan *acceptable* dengan skor SUS 70-100.

Kemudian, kriteria *grade scale* mengonversikan nilai skor akhir SUS ke dalam skala huruf mulai dari *grade A* (paling baik) sampai *grade F* (paling buruk) untuk menyajikan kesimpulan kualitas secara ringkas. Sementara itu, kriteria *adjective rating* memberikan penilaian deskriptif atas persepsi kualitatif pengguna menggunakan kata sifat seperti *worst imaginable*, *poor*, *OK*, *good*, *excellent*, dan *best imaginable*.

Penggunaan SUS dalam tahap evaluasi ini memungkinkan peneliti untuk memperoleh data yang valid mengenai kelayakan antarmuka dibangun, meskipun dengan jumlah responden yang terbatas. Hasil pengujian ini memberikan dasar ilmiah yang kuat dalam menentukan keberhasilan pengembangan sistem dari sisi pengalaman pengguna (*user experience*) [48].



Gambar 2. Skala Penentuan Hasil Penilaian SUS [14].



Gambar 3. Metode *Agile Development* [52].