

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Tinjauan Pustaka

Penelitian ini bertujuan untuk mengembangkan sebuah aplikasi pencatatan keuangan yang sesuai dengan kebutuhan pemilik warung makan menggunakan pendekatan *User-Centered Design* (UCD). Pendekatan ini telah banyak diterapkan dalam berbagai bidang, termasuk dalam pengembangan aplikasi keuangan. Penelitian sebelumnya menunjukkan bahwa menggunakan UCD dapat mengembangkan aplikasi yang sesuai dengan preferensi pengguna. Berikut adalah beberapa temuan dari penelitian sebelumnya.

Pada penelitian yang pertama [16] bertujuan untuk mengembangkan aplikasi *Mobile Learning Management System* menggunakan *Flutter Framework* untuk mahasiswa di Institut Teknologi Telkom Purwokerto dengan pendekatan *User-Centered Design* (UCD). Tujuan utamanya adalah menciptakan sebuah sistem berbasis *mobile* yang dapat diakses oleh mahasiswa dan berfungsi sesuai dengan kebutuhan pengguna dalam melaksanakan pembelajaran daring. Pada penelitian ini menggunakan metode UCD. Pengembangan aplikasi dilakukan menggunakan *Framework Flutter*, sebuah *platform open-source* yang memungkinkan pembuatan aplikasi *mobile* secara efisien. Penelitian ini melibatkan partisipasi dan pengumpulan umpan balik dari mahasiswa sebagai pengguna utama dalam proses perancangan dan pengujian aplikasi. Hasil penelitian menunjukkan bahwa *Flutter Framework* dan teknik UCD dapat digunakan untuk membuat aplikasi sistem manajemen pembelajaran *mobile* yang berfungsi secara efektif dan memenuhi tuntutan target penggunanya, yaitu siswa. Program ini berfungsi sebagaimana mestinya dan dapat membantu penyebaran pembelajaran *online* melalui ponsel. Pendekatan ini diharapkan dapat membantu Institut Teknologi Telkom Purwokerto dalam meningkatkan efektivitas dan kualitas pendidikan daringnya.

Pada penelitian yang kedua [17] bertujuan membuat aplikasi untuk melacak anak-anak di Sekolah Luar Biasa B (SLB B) Yakut Purwokerto yang memiliki kebutuhan khusus menggunakan pendekatan desain yang berpusat pada pengguna.

Tujuan utamanya adalah untuk membangun sistem pemantauan yang efisien dan berhasil dalam melacak kemajuan anak-anak berkebutuhan khusus dan memfasilitasi komunikasi antara pendidik dan wali murid. Pendekatan UCD diterapkan dalam penelitian ini. Untuk memahami keinginan dan preferensi konsumen, data dikumpulkan melalui observasi, wawancara, dan pengumpulan dokumen. Perencanaan proses yang berpusat pada pengguna, spesifikasi konteks penggunaan, spesifikasi kebutuhan pengguna dan organisasi, desain solusi desain produk, dan penilaian desain berdasarkan kebutuhan pengguna adalah lima langkah yang membentuk penelitian ini. Sistem *monitoring* Anak Berkebutuhan Khusus (ABK) di Sekolah Luar Biasa B (SLB B) Yakut Purwokerto telah berhasil mencapai tahap prototipe berkat rancangan aplikasi hasil penelitian. Aplikasi ini dapat lebih interaktif dan disesuaikan dengan kebutuhan dan preferensi penggunanya, yaitu guru dan wali murid, dengan memanfaatkan metode UCD. Hal ini dapat memberikan dampak positif terhadap pemantauan dan peningkatan kemajuan akademik siswa berkebutuhan khusus di kelas.

Pada penelitian yang ketiga [18] bertujuan untuk merombak portal web jurusan psikologi dengan menggunakan teknik *User-Centered Design* (UCD) di FISIP Universitas Brawijaya. Meningkatkan kegunaan dan pengalaman pengguna Portal Web adalah tujuan utama, karena memungkinkan siswa untuk menemukan dan mengambil materi yang berkaitan dengan kegiatan perkuliahan dengan mudah. Pendekatan UCD diterapkan dalam penelitian ini. Pengguna aktif akan dilibatkan dalam setiap langkah proses pengembangan Portal Web dengan menggunakan pendekatan ini: mulai dari menilai desain versi 2014 dan 2017 hingga menganalisis masalah, mencari tahu konteks pengguna, mendefinisikan kebutuhan, mengevaluasi saran, merancang solusi, dan melakukan tinjauan akhir. Standar UB dan rekomendasi HHS dikutip dalam aturan desain penelitian ini, yang juga menggunakan pendekatan penilaian WEBUSE. Temuan penelitian menunjukkan bahwa penggunaan teknik UCD dan saran penilaian dapat membantu menyelesaikan masalah pada versi desain 2014 dan 2017. Pengalaman mahasiswa dalam menggunakan Portal Web Jurusan Psikologi FISIP Universitas Brawijaya

meningkat dan lebih mudah digunakan dengan menggunakan kriteria desain yang didasarkan pada Standardisasi UB dan prinsip-prinsip HHS.

Pada penelitian yang keempat [19] bertujuan untuk membuat antarmuka pengguna (UI) dan pengalaman pengguna (UX) untuk platform *Placeplus*, sebuah perusahaan digital yang memfasilitasi reservasi *online* dan pengelolaan ruang kerja bersama melalui situs web. Fokusnya adalah memberikan kesan awal yang positif bagi calon pengguna dan meningkatkan kesesuaian antara desain dengan kebutuhan pengguna. Pada penelitian ini menggunakan metode UCD. Pendekatan UCD melibatkan pengguna secara aktif pada setiap tahap perancangan, meliputi analisis, desain, evaluasi, dan implementasi. Pengguna akan terlibat dalam proses evaluasi desain yang telah dibuat, sehingga jika ada kekurangan atau perbaikan yang diperlukan, dapat dilakukan desain ulang yang lebih sesuai dengan kebutuhan pengguna. Temuan penelitian menunjukkan bahwa desain UI/UX *Placeplus* dapat dimodifikasi agar lebih sesuai dengan kebutuhan dan preferensi pengguna dengan menerapkan metodologi UCD. Setelah proses desain ulang yang mempertimbangkan hasil penilaian pengguna, *Placeplus* menerima umpan balik yang baik dari pengguna. Dengan demikian, platform *Placeplus* menawarkan pengalaman pengguna yang lebih baik dan UI/UX yang lebih berkualitas berkat penggunaan UCD.

Pada penelitian yang kelima [20] bertujuan untuk mengembangkan aplikasi survei *online* yang mengikuti metodologi *User-Centered Design* (UCD). Untuk menyediakan antarmuka yang mudah digunakan dan memenuhi kebutuhan pengguna, pendekatan ini sangat menekankan peran pengguna dalam proses pengembangan sistem. Pendekatan UCD yang digunakan dalam penelitian ini memungkinkan untuk membuat antarmuka aplikasi survei berbasis web dengan mempertimbangkan konteks dan kebutuhan pengguna. Penelitian ini menunjukkan bagaimana kegunaan aplikasi survei berbasis web dapat berhasil ditingkatkan dengan menerapkan teknik UCD pada antarmuka. Aplikasi berjalan dengan tingkat keberhasilan 100% dan pengguna puas dengan sebanyak 85,6% pengguna senang dengan antarmuka yang ditawarkan.

Tabel 2.1 Studi Terdahulu

No	Judul	Cendekiawan	Pendekatan	Hasil
1	Pengembangan <i>Mobile Learning Management System</i> Dengan <i>User-Centered Design</i> (UCD) Menggunakan <i>Flutter Framework</i>	Condro Kartiko, Ariq Cahya Wardhana, Diovianto Putra Rakhmadani	Metode UCD dengan <i>Flutter Framework</i> , platform <i>open-source</i> , untuk membuat aplikasi <i>mobile</i> yang efektif Sebagai pengguna utama, siswa memberikan umpan balik penting untuk membantu proses perancangan dan pengujian aplikasi.	Sistem ini diharapkan akan berkontribusi positif dalam meningkatkan kualitas dan efisiensi pembelajaran <i>online</i> di Institut Teknologi Telkom Purwokerto. Semua fungsi aplikasi berjalan dengan baik dan memenuhi kebutuhan siswa.
2	Pengembangan Aplikasi <i>Monitoring Anak Berkebutuhan Khusus</i> di Sekolah Luar Biaya Menggunakan	Rifki Adhitama, Ariq Cahya Wardhana, Gracia Rizka Pasfica,	Melalui observasi, wawancara, dan dokumentasi, teknik UCD mengumpulkan informasi untuk memahami kebutuhan dan preferensi pengguna. Penelitian ini dibagi	Berhasil menyelesaikan tahap prototipe. Aplikasi ini diharapkan lebih interaktif dan sesuai dengan kebutuhan guru dan wali murid. Diharapkan juga dapat memberikan kontribusi positif

No	Judul	Cendekiawan	Pendekatan	Hasil
	Metode <i>User-Centered Design</i> (UCD)	Yolanda Al Hidayah Pasaribu	menjadi lima tahap: spesifikasi konteks penggunaan, desain solusi desain produk, spesifikasi kebutuhan pengguna dan organisasi, perencanaan proses yang berpusat pada pengguna, dan penilaian desain berdasarkan kebutuhan pengguna.	dalam melacak dan meningkatkan perkembangan siswa berkebutuhan khusus di sekolah.
3	Penggunaan Metode <i>User-Centered Design</i> (UCD) dalam Perancangan Ulang Web Portal Jurusan Psikologi FISIP Universitas Brawijaya	Dini Pratiwi, Mochamad Chandra Saputra, Niken Hendrakusuma	Pendekatan UCD membutuhkan partisipasi aktif pengguna dalam proses pembuatan Portal Web. mulai dari penilaian desain, analisis masalah, spesifikasi kebutuhan, analisis saran, penentuan konteks pengguna, dan versi desain solusi antara tahun 2014 dan 2017. Penelitian ini juga menggunakan pendekatan penilaian WEBUSE dan	Masalah desain antara tahun 2014 dan 2017 diperbaiki dengan menggunakan saran evaluasi dan pendekatan UCD. Pengalaman mahasiswa dalam menggunakan Portal Web Jurusan Psikologi FISIP Universitas Brawijaya diperbaiki dan dibuat lebih mudah digunakan dengan menggunakan kriteria desain yang didasarkan pada Standardisasi UB dan prinsip-prinsip HHS.

No	Judul	Cendekiawan	Pendekatan	Hasil
			mengacu pada standar UB dan HHS untuk pedoman desain.	
4	Perancangan <i>User Interface dan User Experience</i> pada <i>Placeplus</i> menggunakan pendekatan <i>User-Centered Design</i>	Muhammad Multazam, Irving V Papatungan, Beni Suranto	Pengguna secara aktif terlibat dalam setiap langkah proses desain, termasuk analisis, desain, penilaian, dan implementasi, berkat teknik UCD. Dengan melibatkan pengguna dalam proses peninjauan desain, ini memberi mereka kesempatan untuk membuat perubahan dan mendesain ulang agar lebih sesuai dengan kebutuhan mereka.	Desain UI/UX <i>Placeplus</i> dapat disesuaikan dengan kebutuhan dan preferensi pengguna dengan memanfaatkan teknik UCD. Proses desain ulang yang menggabungkan umpan balik pengguna adalah hasil dari penilaian pengguna. Platform <i>Placeplus</i> menawarkan pengalaman pengguna dan kualitas UI/UX yang unggul ketika UCD digunakan.
5	Pengembangan Tampilan Antarmuka Aplikasi Survei Berbasis Web dengan Metode <i>User-Centered Design</i> (UCD)	Dicky Larson Kaligis, Reyful Rey Fatri	Pendekatan UCD digunakan untuk membangun antarmuka aplikasi survei berbasis web melalui langkah-langkah analisis konteks pengguna, analisis kebutuhan pengguna, desain solusi, dan penilaian desain solusi.	Meningkatkan tingkat keberhasilan aspek efektivitas menjadi 100%, waktu pemrosesan aspek efisiensi dipercepat, dan 85,6% pengguna menyatakan kepuasan dengan antarmuka-semua indikasi dari peningkatan kegunaan

No	Judul	Cendekiawan	Pendekatan	Hasil
				yang berhasil ditingkatkan. Dengan memastikan bahwa antarmuka memenuhi kebutuhan dan preferensi pengguna, teknik UCD meningkatkan pengalaman pengguna dengan aplikasi survei berbasis web.

2.2 Landasan Teori

Berikut ini adalah beberapa teori pendukung yang digunakan dalam penelitian ini:

2.2.1 Pencatatan Keuangan

Pencatatan keuangan adalah proses sistematis untuk mencatat, mengklasifikasikan, dan merangkum transaksi keuangan yang dilakukan oleh suatu entitas, baik itu individu maupun organisasi, selama periode tertentu [21]. Proses ini bertujuan untuk menyediakan informasi keuangan yang akurat dan relevan yang dapat digunakan untuk pengambilan keputusan ekonomi.

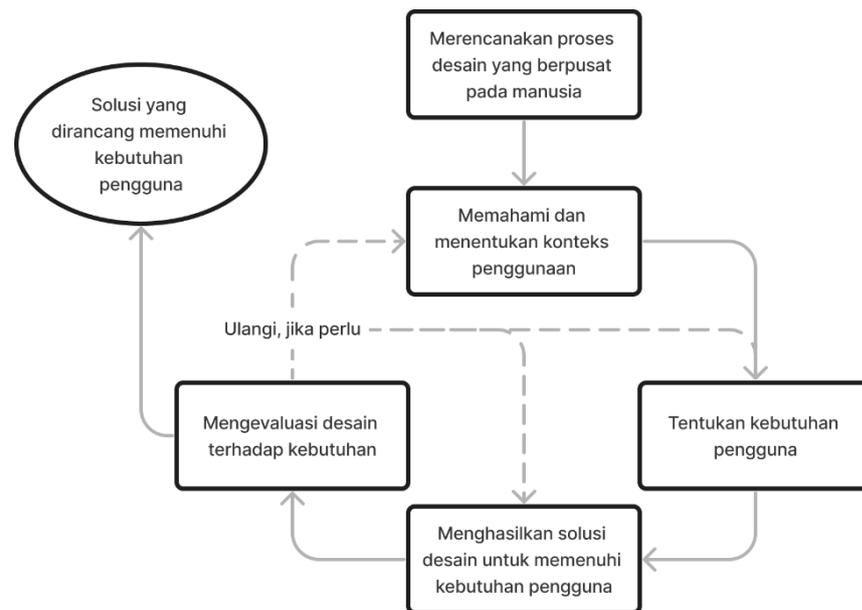
Dalam konteks usaha kecil seperti warung makan, pencatatan keuangan yang baik dapat membantu pemilik usaha untuk lebih memahami kondisi keuangan mereka, mengidentifikasi area yang memerlukan perbaikan, dan mengoptimalkan operasi harian. Dengan menggunakan teknologi dan aplikasi pencatatan keuangan, proses pencatatan ini dapat disederhanakan dan ditingkatkan efisiensinya, sehingga membantu pemilik usaha dalam menjalankan bisnis mereka dengan lebih efektif dan mengurangi risiko kesalahan manusia.

2.2.2 *User-Centered Design (UCD)*

Metodologi desain yang dikenal sebagai desain yang berpusat pada pengguna (UCD) memprioritaskan kebutuhan dan keinginan manusia. Melalui pertimbangan kegunaan, karakteristik pengguna, lingkungan, aktivitas, dan alur kerja dalam setiap iterasi desain, UCD mengikutsertakan manusia dalam proses desain [22]. Tujuan utama UCD adalah membuat pengalaman pengguna nyaman dan sesuai dengan kebutuhan mereka. Untuk meningkatkan desain produk digital, UCD memanfaatkan umpan balik pengguna.

UCD dan *Human Centered Design (HCD)* dapat dibandingkan. Penggunaan aplikasi dan penggunaan faktor manusia/ergonomi, pengetahuan, dan metodologi kegunaan merupakan fokus utama dari pendekatan desain

dan pengembangan yang berpusat pada manusia (HCD) [23]. Istilah HCD sering digunakan sebagai pengganti UCD untuk menekankan bahwa HCD juga membahas dampak pada banyak pemangku kepentingan, bukan hanya mereka yang biasanya terlibat dalam desain aplikasi. Tetapi dalam kenyataannya, kata-kata ini sering digunakan secara sinonim.



Gambar 2.1 Tahapan *User-Centered Design*

Diagram pada Gambar 2.1 merupakan tahapan dalam metode UCD [23]. Berikut penjelasan masing-masing tahapannya:

- a. Merencanakan proses desain yang berpusat pada manusia
Langkah pertama dalam UCD adalah merencanakan proses desain yang fokus pada kebutuhan pengguna. Ini melibatkan perencanaan sumber daya, waktu, dan strategi yang akan digunakan untuk melibatkan pengguna dalam proses desain.
- b. Memahami dan menentukan konteks penggunaan
Pada tahap ini, penting untuk memahami bagaimana dan dalam kondisi apa produk akan digunakan. Ini melibatkan penelitian tentang lingkungan pengguna, tugas yang mereka lakukan, dan alat yang

mereka gunakan. Observasi langsung, wawancara, dan survei sering digunakan untuk mengumpulkan informasi ini.

- c. Tentukan kebutuhan pengguna
Setelah memahami konteks penggunaan, langkah selanjutnya adalah menentukan kebutuhan spesifik pengguna. Ini melibatkan identifikasi masalah yang dihadapi pengguna, apa yang mereka harapkan dari produk, dan fitur apa yang akan memberikan nilai tambah bagi mereka.
- d. Menghasilkan solusi desain untuk memenuhi kebutuhan pengguna
Berdasarkan kebutuhan yang telah diidentifikasi, mulai masuk ke tahap menghasilkan solusi desain. Ini bisa berupa sketsa awal, *wireframe*, atau prototipe. Tujuannya adalah untuk mengembangkan konsep yang bisa diuji dan divalidasi dengan pengguna.
- e. Mengevaluasi desain terhadap kebutuhan
Tahap terakhir adalah evaluasi desain. Ini melibatkan pengujian prototipe dengan pengguna untuk memastikan bahwa desain yang dihasilkan benar-benar memenuhi kebutuhan mereka. Metode evaluasi bisa berupa *usability testing*, *A/B testing*, atau evaluasi heuristik. Hasil dari evaluasi ini digunakan untuk membuat perbaikan dan iterasi lebih lanjut pada desain.

2.2.3 Unified Modeling Language

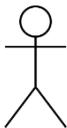
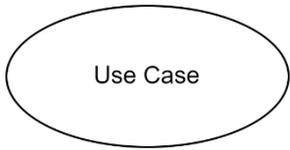
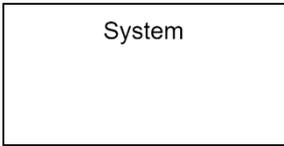
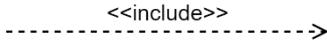
Aplikasi perangkat lunak dirancang dan didokumentasikan dengan menggunakan *Unified Modeling Language* (UML) [24]. Insinyur perangkat lunak dapat merancang, memvisualisasikan, dan merekam artefak aplikasi yang kompleks dengan bantuan UML. UML terdiri dari beberapa jenis diagram, yang masing-masing dimaksudkan untuk menjelaskan komponen yang berbeda dari program yang sedang dibuat. Ini termasuk diagram kasus penggunaan, diagram aktivitas, diagram urutan, dan diagram kelas. UML tidak hanya membantu proses rekayasa perangkat lunak dari analisis awal hingga implementasi dan pemeliharaan, tetapi juga memudahkan para

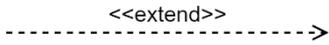
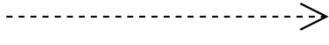
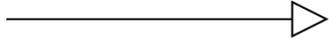
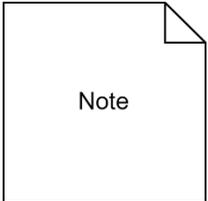
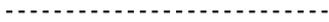
pemangku kepentingan dan anggota tim pengembangan untuk berkomunikasi dengan jelas dan efektif.

2.2.4 Use Case Diagram

Use case diagram adalah alat yang digunakan dalam UML untuk mengilustrasikan bagaimana pengguna dan sistem berinteraksi [25]. Diagram alir ini menggambarkan bagaimana pengguna akan mengoperasikan program komputer atau sistem dalam skenario tertentu. Ketika mengembangkan situs web, aplikasi seluler, atau sistem lainnya, diagram *use case* sangat penting.

Tabel 2.2 Elemen *Use Case Diagram*

Elemen	Nama	Ringkasan
 Actor	<i>Actor</i>	Orang yang terhubung ke sistem
 Use Case	<i>Use Case</i>	Kegiatan yang dapat dilakukan oleh pelaku pada sistem
	<i>Association</i>	Hubungan <i>actor-case</i>
 System	<i>System</i>	Sistem yang sedang dalam proses pengembangan
 <<include>>	<i>Include</i>	Satu <i>use case</i> berisi elemen <i>use case</i> lainnya.

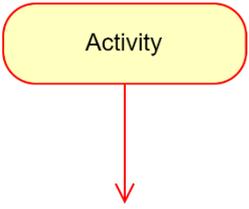
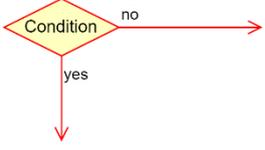
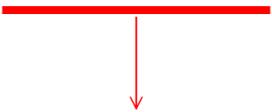
Elemen	Nama	Ringkasan
	<i>Extend</i>	Ada kemungkinan untuk memperluas satu <i>use case</i> ke <i>use case</i> lain
	<i>Dependency</i>	Ketergantungan antara komponen-komponen diagram
	<i>Generalization</i>	Satu aktor atau kasus adalah contoh dari yang lain.
	<i>Collaboration</i>	Dua atau lebih aktor dan kasus yang terkait
	<i>Note</i>	Penjelasan tambahan tentang komponen diagram
	<i>Anchor</i>	Hubungan antara nota teks dan komponen diagram lainnya

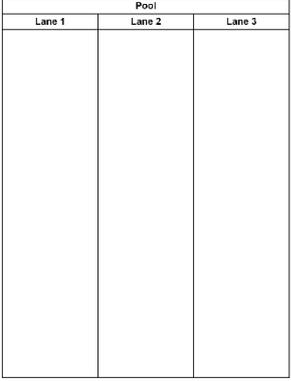
2.2.5 Activity Diagram

Diagram aktivitas, yang menggambarkan tindakan yang terjadi dalam suatu sistem, adalah jenis diagram yang sering digunakan dalam pemodelan

proses bisnis [26]. Diagram ini dapat menunjukkan proses-proses dalam diagram *use case* dan sering digunakan dalam pemodelan proses bisnis.

Tabel 2.3 Elemen *Activity Diagram*

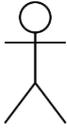
Elemen	Nama	Ringkasan
	<i>Start Point</i>	Awal status
	<i>Activity</i>	Tugas diselesaikan oleh sistem
	<i>Decision</i>	Percabangan di mana ada banyak pilihan aktivitas
	<i>Join</i>	Penggabungan adalah ketika seseorang melakukan lebih dari satu hal dan kemudian menggabungkannya menjadi satu
	<i>End Point</i>	Akhir status

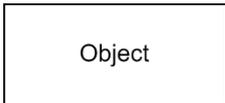
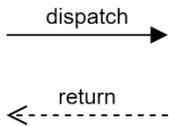
Elemen	Nama	Ringkasan
	<p><i>Swimlane</i></p>	<p>Entitas perusahaan yang berbeda dan bertanggung jawab atas tindakan yang terjadi</p>

2.2.6 Sequence Diagram

Dalam UML, *sequence diagram* adalah salah satu jenis diagram yang digunakan untuk menunjukkan dan menggambarkan secara rinci bagaimana item berinteraksi satu sama lain dalam suatu sistem [27]. Saat menjelaskan proses *login* pada suatu sistem, di mana aktor (pengguna) berinteraksi dengan objek (sistem) untuk menghasilkan keluaran (*login* berhasil atau gagal), berikut adalah contoh penggunaan *sequence diagram*.

Tabel 2.4 Elemen *Sequence Diagram*

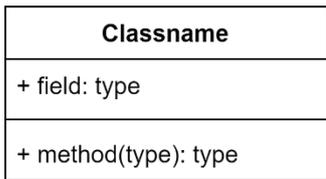
Simbol	Nama	Keterangan
 <p>Actor</p>	<p><i>Actor</i></p>	<p>Pengguna yang berinteraksi dengan sistem tetapi tidak berada di dalamnya</p>
	<p><i>Activation Box</i></p>	<p>Mewakili jumlah waktu yang dibutuhkan oleh sesuatu untuk menyelesaikan tugasnya</p>

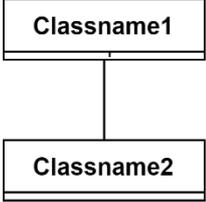
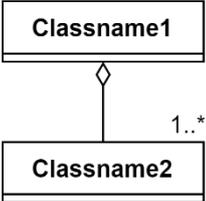
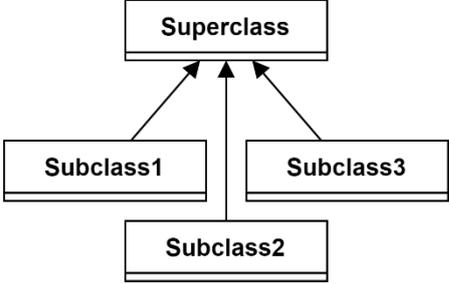
Simbol	Nama	Keterangan
	<i>Lifeline</i>	Menggambarkan bagaimana objek bekerja
	<i>Object</i>	Objektif sistem
	<i>Messages</i>	Menggambarkan cara komunikasi antar objek terjadi

2.2.7 Class Diagram

Dalam UML, *class diagram* adalah suatu bentuk diagram yang menunjukkan hubungan, metode, karakteristik, dan struktur kelas atau objek di dalam suatu sistem [28]. *Class diagram* membantu dalam penjelasan struktur organisasi sistem dan hubungan antar kelas.

Tabel 2.5 Elemen *Class Diagram*

Elemen	Nama	Ringkasan
	<i>Class</i>	<p>Tiga komponen membentuk blok bangunan mendasar dari pemrograman berorientasi objek.</p> <p>Nama kelas muncul di bagian atas. Ciri-</p>

Elemen	Nama	Ringkasan
		<p>ciri kelas terletak di bagian tengah. Operasi kelas dan metode terletak di bagian bawah.</p>
 <pre> classDiagram Classname1 --- Classname2 </pre>	<p><i>Association</i></p>	<p>Dua kelas memiliki koneksi statis.</p>
 <pre> classDiagram Classname1 o-- "1..*" Classname2 </pre>	<p><i>Aggregation</i></p>	<p>Hubungan antara dua kelas di mana satu kelas dapat berdiri sendiri-sendiri terhadap kelas lainnya, namun tetap menjadi anggotanya.</p>
 <pre> classDiagram Superclass < -- Subclass1 Superclass < -- Subclass2 Superclass < -- Subclass3 </pre>	<p><i>Inheritance</i></p>	<p>Kapasitas untuk meneruskan ke kelas baru (subkelas) semua metode dan atribut kelas asli (superkelas).</p>

2.2.1 *Node.js*

Node.js adalah perangkat lunak lintas platform *open-source* yang memungkinkan pengembangan aplikasi berbasis web yang menggunakan *JavaScript* sebagai bahasa pemrograman [29]. *Node.js* memungkinkan penggunaan *JavaScript* dari *server* dan merupakan *platform* yang memungkinkan pengembangan aplikasi berbasis web.

2.2.8 *React Native*

Kerangka kerja *JavaScript* yang disebut *React Native* sering dibuat oleh platform *Facebook* dan digunakan untuk membuat aplikasi untuk *platform iOS* dan *Android* [30]. Kerangka kerja untuk membuat aplikasi seluler yang disebut *React Native* memungkinkan pembuatan aplikasi untuk *platform iOS* dan *Android*. *React Native* menawarkan primitif berbasis "*Text*" dalam perangkat lunaknya untuk mengembangkan aplikasi seluler, yang sangat kontras dengan "*Span*" yang sering digunakan untuk membuat halaman web. *React Native* pada dasarnya adalah kerangka kerja yang memungkinkan pemrogram menulis aplikasi untuk beberapa *platform* dari satu basis kode.

2.2.9 *Black Box Testing*

Perangkat lunak yang diuji secara fungsional tanpa mengetahui struktur kodenya dikenal sebagai pengujian "kotak hitam". *Input* dan *output* perangkat lunak diuji dalam pengujian ini. Menemukan masalah dengan fungsi, antarmuka, struktur data, kinerja, inisialisasi, dan penghentian adalah tujuan dari pengujian kotak hitam. Pendekatan partisi ekuivalen diterapkan dalam pengujian ini [31].

Metode pengujian perangkat lunak yang disebut *equivalency partitions* digunakan untuk memverifikasi bahwa jenis input dan kriteria yang tercantum dalam Spesifikasi Persyaratan Perangkat Lunak (SRS) terpenuhi [32]. Masukan ini dapat dipisahkan menjadi banyak kelas kesetaraan dalam kondisi ini, termasuk valid dan tidak valid. Dengan mendefinisikan kasus uji

yang menyoroti kelas kesalahan, metode ini bertujuan untuk meminimalkan jumlah kasus uji yang perlu dibuat.

2.2.10 *System Usability Scale*

Skala Kegunaan Sistem (SUS), yang dikembangkan oleh John Brooke pada tahun 1986 [33], adalah metode yang dapat digunakan untuk menilai seberapa ramah pengguna aplikasi suatu produk pada berbagai item, termasuk situs web, aplikasi telepon seluler, perangkat keras, dan perangkat lunak. SUS adalah metode pengujian kegunaan yang umum karena mudah dihitung dan digunakan, menjadikannya pilihan populer di antara pendekatan lainnya. Ini adalah kuesioner yang menilai kegunaan yang dirasakan. Skala *Likert* dengan lima poin (satu menunjukkan sangat tidak setuju, dua tidak setuju, tiga netral, empat setuju, dan lima menunjukkan sangat setuju) digunakan dalam SUS.

Tabel 2.6 Tabel Kuesioner SUS

No	Pertanyaan Kuesioner	Skala Penilaian				
		1	2	3	4	5
1	Saya ingin menggunakan aplikasi ini sepanjang waktu.					
2	Menurut saya, aplikasi ini terlalu rumit.					
3	Menurut saya aplikasi ini mudah digunakan.					
4	Saya memerlukan bantuan teknis untuk menggunakan aplikasi ini.					
5	Aplikasi ini memiliki fungsi yang terintegrasi dengan baik.					
6	Menurut saya, ada ketidaksesuaian pada aplikasi ini.					
7	Saya yakin mayoritas orang akan menguasai penggunaan aplikasi ini dengan cepat.					
8	Aplikasi ini sangat sulit digunakan.					

No	Pertanyaan Kuesioner	Skala Penilaian				
		1	2	3	4	5
9	Saat saya menggunakan aplikasi ini, saya menjadi lebih percaya diri.					
10	Sebelum saya bisa menggunakan aplikasi ini, saya harus belajar banyak.					

Setelah mendapat data dari beberapa responden, kemudian jawaban dari keseluruhan responden dihitung. Berikut cara menghitungnya:

1. Kurangi nilai satu untuk setiap pertanyaan dalam urutan ganjil.
2. Kurangi nilai lima untuk setiap pertanyaan dalam urutan genap.
3. Tambahkan nilai-nilai dari pernyataan dengan nomor genap dan ganjil.
4. Akhirnya kalikan hasil penjumlahan dengan 2,5.

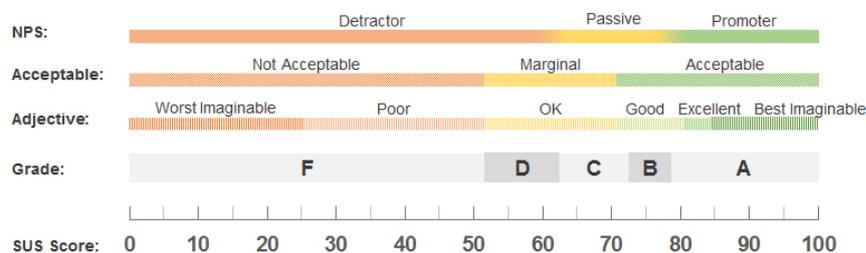
$$x = \left(\sum (a - 1) + (b - 5) \right) \times 2,5$$

a = pertanyaan urutan ganjil

b = pertanyaan urutan genap

x = hasil pengukuran SUS

Setelah hasil responden SUS dihitung, kemudian diukur untuk mengetahui hasil pengujiannya. Berikut skala pengukurannya:



Gambar 2.2 Skala Pengukuran SUS