

## **BAB II**

### **LANDASAN TEORI**

#### **2.1 Tinjauan Pustaka**

Kajian pustaka berperan penting dalam mendukung penelitian yang sedang dilakukan. Proses ini melibatkan pencarian dan pemilahan berbagai hasil studi sebelumnya yang memiliki keterkaitan dengan topik yang diteliti, dengan merujuk pada penelitian-penelitian tersebut. Peneliti dapat memperkuat argumen, memperkaya referensi serta membandingkan temuan yang ada. Kajian pustaka mencakup identifikasi secara sistematis, penelusuran, dan analisis terhadap sumber-sumber yang relevan dengan isu yang sedang diteliti. Penelitian terdahulu dijadikan sebagai landasan teori atau referensi untuk memberikan gambaran awal, pembandingan, pelengkap terhadap penelitian yang sedang dilakukan (Ridho M, 2020).

Kekuatan dari tinjauan pustaka terletak pada kemampuannya dalam mengintegrasikan berbagai temuan dan perspektif dari penelitian terdahulu guna menyediakan jawaban yang valid terhadap pertanyaan penelitian (Snyder, 2019). Selain itu, tinjauan pustaka juga dapat dimanfaatkan untuk *Book excerpt - NFS Seng Hansen*, memetakan kesenjangan penelitian yang ada, mengembangkan suatu kerangka teoritis maupun model-model konseptual yang diperlukan dalam suatu penelitian (Snyder, 2019). Namun demikian, tinjauan pustaka juga memiliki beberapa kekurangan, terutama apabila dilakukan secara tidak sistematis. Tinjauan pustaka yang dilakukan secara tradisional dapat mempengaruhi kualitas analisis yang dihasilkan. Pada pendekatan ini, terdapat kemungkinan peneliti membuat berbagai asumsi ketika mengintegrasikan dan menyintesis publikasi yang ada, atau bersikap selektif terhadap publikasi yang ada sehingga mengabaikan potensi temuan penting dari tinjauan pustaka (Snyder, 2019). Adapun beberapa sumber pustaka yang digunakan adalah sebagai berikut dalam bentuk Tabel 2.

Tabel 2.1 Tinjauan Pustaka

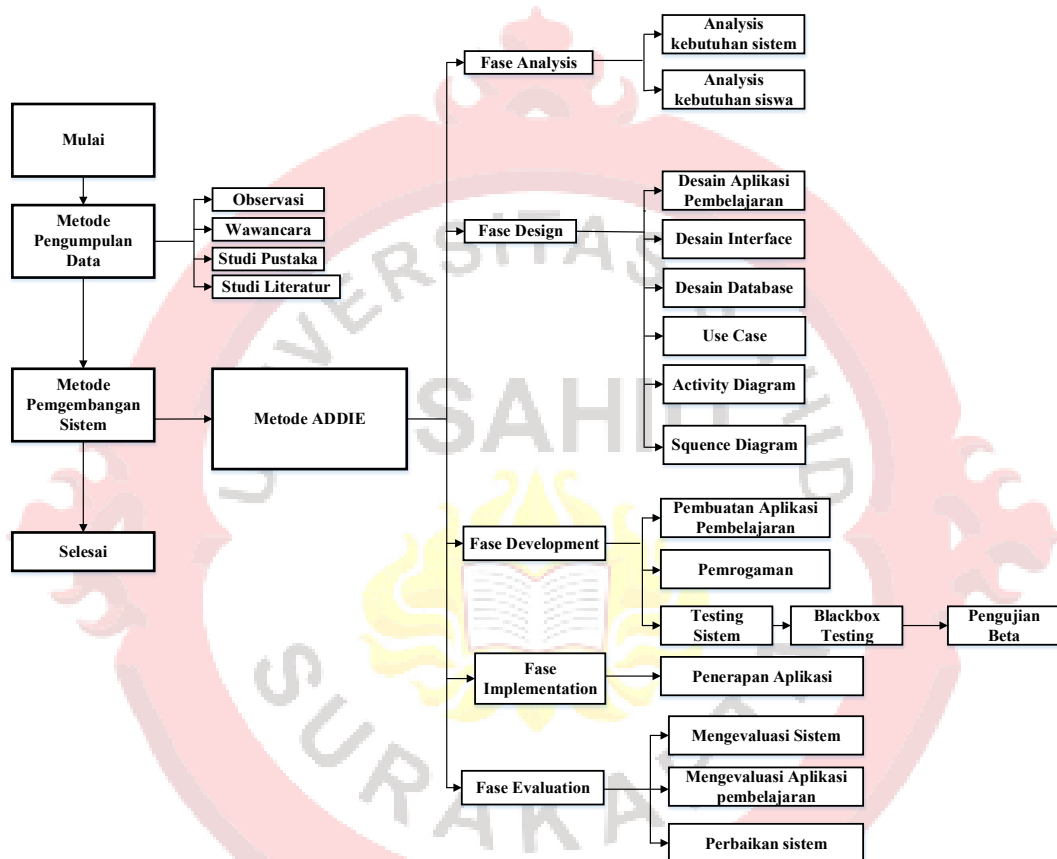
<b>Pengarang</b>	Fahrur Rozi, Ellyfiana NikmatusSholikhah Jurnal Pendidikan Informatika Vol. 7 No. 2,Desember, 2023, Hal. 178	Afri Delfiza, Ade Pratama,Haris Kurniawan Jurnal Riset dan Aplikasi Mahasiswa Informatika (JRAMI) Vol04 No 01 Tahun 2023	Fadila Shafitri, Gusnita Darmawati, Riri Okra, Firdaus Annas JATI (Jurnal Mahasiswa Teknik Informatika) Vol. 7 No.5, Oktober 2023	Arifin Nur Syahid Hudawan, Dahlan Susilo, and Firdhaus Hari Saputro. Tugas Akhir Arifin Nur Syahid Hudawan Universitas sahid Surakarta (2021)	Candra Pratama, Kaspul,M.Arsyad Jurnal Program Studi Pendidikan Biologi (Agustus), Vol. (10), No.(2) (2021)
<b>Judul Penelitian</b>	Aplikasi Pembelajaran Interaktif berbasis Androidpada Mata Pelajaran Informatika.(Rozi & Sholikhah, 2023)	Perancangan Media Pembelajaran interaktif berbasis Android pada mata pelajaran dasar desain grafis kelas x. (Delfiza, 2023)	Perancangan Media Pembelajaran interaktif mata pelajaran Informatika berbasis android kelas x menggunakan articulate storyline di SMA negeri 2 Bukittinggi. (Shafitri dkk, 2023)	Rancang Bangun dan Implementasi Company Profile Balai Latihan Kerja Surakarta Berbasis Android ( Arifin dkk,2021)	Pengembangan Media Pembelajaran Interaktif Berbasis Aplikasi Android Pada Konsep Sistem Pernapasan Manusia Jenjang SMA

<b>Tujuan Penelitian</b>	mengembangkan media pembelajaran interaktif dan mengetahui tingkat kelayakandari aplikasi tersebut	Tujuan penelitian ini yaitu untuk Merancang media pembelajaran interaktif berbasis android pada mata pelajaran Dasar Desain Grafis kelas Xdi SMKN 1 X Koto.	Tujuan dari penelitian ini adalah untuk merancang media pembelajaran interaktif mata pelajaran Informatika berbasis android kelas x menggunakan Articulate Storyline di SMA Negeri 2 Bukittinggi yang valid,praktis dan efektif.	Tujuan dari tugas akhir ini adalah untuk menghasilkan aplikasi website company profile Balai Latihan Kerja Surakarta Surakarta berbasis A	Tujuan penelitian ini adalah membuat membuat Media pembelajaran Interaktif Berbasis Aplikasi Android Pada Konsep Sistem Pernapasan Manusia Jenjang SMA
<b>Metodologi Penelitian</b>	Untuk pengembangan sistemnya penulis menggunakan metode penelitian	Metode yang digunakan Multimedia Development Life Cycle (MDLC).	Metode yang digunakan Penelitian ini Research and Development (R&D) Dengan mengadaptasi	penulis menggunakan metode <i>waterfall</i>	Metode yang digunakan penulis menggunakan metode Research and Development (R&D)

	dan pengembangan atau <i>Research and Development (R&amp;D)</i> .		model pengembangan Hannafin and Peck)		
<b>Hasil Penelitian</b>	Penelitian ini menghasilkan sebuah produk aplikasi pembelajaran android pada Mata Pelajaran Informatika pada Kurikulum Merdeka	Penelitian ini menghasilkan sebuah produk aplikasi pembelajaran android pada Mata Pelajaran dasar desain grafis kelas x	Penelitian ini menghasilkan sebuah produk aplikasi pembelajaran android pada Mata Pelajaran Informatika di SMA Negeri 2 Bukittinggi.	Penelitian ini menghasilkan Aplikasi company profile Balai Latihan Kerja Surakarta berbasis Android	Penelitian ini menghasilkan Media pembelajaran Interaktif Berbasis Aplikasi Android Pada Konsep Sistem Pernapasan Manusia Jenjang SMA yang sangat layak.

## 2.2 Kerangka Pemikiran

Pengembangan aplikasi pembelajaran pada mata Pelajaran Informatika berbasis android ini disusun melalui beberapa tahapan yang dilakukan dengan tujuan memudahkan dalam penulisan penelitian. Pembuatan kerangka berfikir ini berdasarkan penelitian yang dilakukan (Rokhman, 2020). Alur Penelitian disajikan pada Gambar 2.1.



Gambar 2.1 Kerangka Pemikiran

Kerangka pemikiran dari tugas akhir ini dapat dijelaskan atau didefinisikan sebagai berikut:

### 1. Metode Pengumpulan data

Metode pengumpulan data yang digunakan oleh penulis adalah menggunakan 4 cara, yaitu: observasi, wawancara, studi pustaka dan studi literatur sejenis:

- 1) Observasi Observasi dilakukan selama bulan Oktober 2024 di

SMK Muhammadiyah 5 Miri Berdasarkan pengamatan yang dilakukan, diperoleh informasi-informasi mengenai:

- Sejarah singkat SMK Muhammadiyah 5 Miri Memuat tentang latar belakang mengenai sejarah awal berdirinya, visi dan misinya
- Sistem atau proses Pembelajaran yang berjalan.

## 2) Wawancara

Dalam hal ini, wawancara dilakukan dengan Ibu Azizah Fathin Nur Hanifah, S.Kom selaku guru Mata Pelajaran Informatika di SMK Muhammadiyah 5 Miri, pada tanggal 10 Oktober 2024. Ada beberapa pertanyaan yang penulis berikan untuk mendapatkan kebutuhan yang diperlukan dalam membangun aplikasi pembelajaran interaktif pada mata Pelajaran Informatika berbasis Android.

Tabel 2.2 Tabel Wawancara

Topik Penelitian:	APLIKASI PEMBELAJARAN INTERAKTIF PADA MATA PELAJARAN INFORMATIKA BERBASIS ANDROID DI SMK MUHAMMADIYAH 5 MIRI SRAGEN
Nama Unit Terkait:	Titik Mandansari (Peneliti), Ibu Azizah Fathin Nur Hanifah, S.Kom (Guru Informatika)
Pembahasan:	Tentang proses belajar saat ini, masalah saat pembelajaran, kebutuhan aplikasi, tentang sekolah.
Hari dan Tanggal:	Kamis, 10 Oktober 2024
Tempat:	SMK Muhammadiyah 5 Miri Sragen
Pukul:	09.00 – 10.15 WIB
Draft Pertanyaan:	1.Bagaimana proses pembelajaran mata pelajaran Informatika ini? 2.Apakah ada kesulitan saat melakukan pembelajaran yang hanya menggunakan panduan buku LKS? 3.Solusi apa yang diharapkan dari masalah yang ada? 4. Seperti apa aplikasi yang dapat mengembangkan niat belajar siswa?

### 3) Studi Pustaka

Studi pustaka dilakukan dengan mempelajari teori-teori, buku-buku serta artikel-artikel yang berkaitan dengan aplikasi pembelajaran, analisis dan perancangan sistem, metodologi penelitian, pemrograman. Selain itu penulis juga mengumpulkan data dari situs-situs internet yang berhubungan dengan tugas akhir penulis.

### 4) Studi Literatur

Tabel 2.1 dijelaskan literatur-literatur sejenis yang berkaitan dengan tugas akhir yang dikerjakan oleh penulis. Dalam setiap literatur dijelaskan mengenai metodologi yang digunakan dalam membangun aplikasi pembelajaran berbasis android.

### 5) Metode Pengembangan ADDIE

Dalam penyusunan tugas akhir ini, metode pengembangan sistem menggunakan ADDIE (*Analysis, Design, Development, Implementation, and Evaluation*). Model pengembangan pada penelitian ini adalah mengacu pada proses-proses utama dari proses pengembangan sistem pembelajaran. Penulis menggunakan *tools* UML untuk menggambarkan use case diagram dengan software Microsoft Visio 2010. (Muzhaffar, 2022)

Dalam Metode ini ada dibagi menjadi 5 fase yaitu, *Analysis, Design, Development, Implementation, and Evaluation* :

#### - Fase *Analysis*

Fase analisis sistem untuk menentukan siapa pengguna aplikasi, materi apa yang akan diterapkan, metode apa yang sesuai dalam penerapan materi dan media yang akan digunakan.

#### - Fase *Design*

Fase *Design* ini proses menghasilkan rancangan materi, konten, antarmuka, media atau alat pembelajaran, fungsi sistem, rencana

pembelajaran untuk siswa hingga proses bagaimana pembelajaran berlangsung. Disamping itu, pada tahapan ini akan dirancang diagram UML, desain antar muka, dan desain media pembelajaran aplikasi yang dibangun.

- *Fase Development*

Fase ini mulai dikembangkan menjadi sebuah aplikasi pembelajaran. Pengembangan aplikasi pembelajaran ini akan menggunakan beberapa perangkat lunak seperti *Canva*, *Android Studio* dan beberapa aplikasi pendukung. Proses pembuatan konten media pembelajaran dilakukan satu per satu yang dimulai dari pembuatan materi pembelajaran, video pembelajaran, kuis, setelah proses-proses tersebut dikerjakan, kemudian dilakukan proses pembuatan aplikasi pada *Android Studio* hingga menjadi sebuah aplikasi pembelajaran yang dapat diujikan.

- *Fase Implementation*

Tahap *Implementation* ini merupakan proses implementasi aplikasi pembelajaran yang telah dihasilkan. Untuk diterapkan kepada siswa SMK Muhammadiyah 5 Miri khususnya kelas X. Pada tahap ini, pengaturan dan pengkondisian implementasi sistem juga diperhatikan. Hal ini berkaitan dengan perangkat keras pendukung dan lain-lain.

- *Fase Evaluation*

Tahap *Evaluation* ini merupakan tahap terakhir yang dilakukan untuk mengevaluasi konten, materi, dan aplikasi pembelajaran yang dikembangkan.

6) Tahap Pengujian *Black Box Testing*

Tahap ini dilakukan pengujian aplikasi dengan metode *Black box* untuk menguji uji fungsional, metode pengujian menekankan pada menjalankan



fungsi pemeriksaan dan data keluaran. Adapun fitur atau menu yang akan di uji ditampilkan pada Tabel 2.3

Tabel 2.3 Rancangan Pengujian *Black Box*

No	Item Uji	Detail Pengujian	Jenis Pengujian
1.	Admin/Guru	Form Login	<i>Black box</i>
		<i>Dashboard</i> Menu Admin	
		Menu Kelola User	
		Menu Kelola Materi	
		Menu Kelola Video	
		Menu Kelola Kuis	
		Menu Forum	
		Menu Kelola Nilai	
2.	User / Siswa	Form Login	
		<i>Dasboard</i> Menu User	
		Menu materi	
		Menu Video	
		Menu Forum	
		Menu Kerjakan kuis	
		Menu lihat Nilai	

#### 7) Tahap Pengujian *Beta Testing*

Tahap ini dilakukan Pengujian *beta* pengujian yang dilakukan secara *objektif*. Dimana pengujian dilakukan secara langsung terhadap pengguna, biasanya menggunakan kuisisioner mengenai tanggapan pengguna atas perangkat lunak yang telah dibangun.

### 2.3 Landasan Teori

Teori dasar atau umum yang digunakan sebagai landasan dalam penelitian ini. Hal yang akan dijelaskan pada sub bab ini adalah antara lain : Pengertian aplikasi, pengertian pembelajaran interaktif, jurusan informatika, pengertian Android, serta program yang digunakan untuk membangun aplikasi pembelajaran ini antara lain : Android Studio, *Software Development Kit* (SDK), *Java Development Kit* (JDK) bahasa pemrograman Java, UML, *SQLite*, *Black Box Testing*.

### **2.3.1 Aplikasi**

Aplikasi merupakan suatu perangkat lunak (software) atau program komputer yang beroperasi pada sistem tertentu yang diciptakan dan dikembangkan untuk melakukan perintah tertentu. Secara harfiah, aplikasi merupakan suatu penerapan software yang dikembangkan untuk tujuan melakukan tugas-tugas tertentu. (Uddin dkk, 2024)

### **2.3.2 Pembelajaran Interaktif**

Pembelajaran interaktif adalah salah satu pendekatan pembelajaran yang dapat digunakan oleh guru saat mengajar. Guru memainkan peran penting dalam menciptakan suasana pendidikan yang interaktif, di mana siswa dapat berkomunikasi satu sama lain, dengan guru, dan dengan sumber belajar mereka. Pembelajaran interaktif dirancang sedemikian rupa sehingga fokus pembelajaran terpusat pada siswa sehingga mereka dapat belajar dengan lebih baik dan lebih memahami apa yang mereka pelajari. (Ayuanita & Effendy, 2024)

### **2.3.3 Informatika**

Mata pelajaran informatika merupakan salah satu mata pelajaran dalam kategori baru di dalam Kurikulum Merdeka. Pembelajaran Informatika pada perkembangan trend saat ini wajar jika diterapkan kembali menjadi mata pelajaran wajib karena dapat menjadi bekal menuju revolusi 5.0. (Rozi & Sholikhah, 2023)

Mata pelajaran Informatika menjadi ilmu yang wajib dikuasai para pelajar di pendidikan dasar dan menengah. konsep mata pelajaran Informatika berbeda dengan pendidikan TIK meskipun ada beberapa hal yang diadaptasi. Mata pelajaran Informatika tidak hanya mempelajari beragam perangkat lunak komputer, tetapi juga memecahkan masalah dan berpikir kritis. Peserta didik dituntut berpikir komputasional dengan mempelajari beragam disiplin ilmu. (Pebriyanti dkk, 2021)

#### 2.3.4 Android

Android adalah sebuah sistem operasi perangkat *mobile* berbasis linux yang mencakup sistem operasi, *middleware* dan aplikasi. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka. ( Candraningtyas, 2020)

#### 2.3.5 Android Studio

Android Studio merupakan *Integrated Development Environment (IDE)* atau dalam artian lain adalah sebuah lingkungan pengembangan terintegrasi resmi yang memang di rancang khusus untuk pengembangan sistem operasi google Android. Android studio ini adalah lingkungan pengembangan baru dan terintegrasi dengan penuh,yang telah di rilis oleh google untuk sistem operasi Android dan di rancang untuk menjadi peralatan baru dalam pengembangan aplikasi dan memberi alternaif selain *Eclips* yang saat ini menjadi IDE yang banyak di pakai Android studio merupakan IDE pemograman Android resmi dari google dikembangkan oleh *IntelliJ*, sebelumnya IDE resmi pemograman Android adalah *Eclipse*. Sejak kemunculan Android studio google telah tertarik dan menjadikannya Android studio sebagai IDE resminya. ( Wijaya & Sanjaya, 2024)

#### 2.3.6 Software Development Kit ( SDK)

SDK adalah *tools API (Application Programming Interface)* yang diperlukan untuk mulai mengembangkan aplikasi pada *platform Android* menggunakan Bahasa Pemograman java. Saat ini disediakan Android SDK sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada platform Android menggunakan Bahasa pemograman java. SDK mencakup perangkat *tools* pengembangan yang komprehensif. Android SDK terdiri dari *debugger, libraries, handset emulator*, dokumentasi, contoh kode

program dan tutorial. (Nurasyiah dkk, 2022)

### 2.3.7 Java Development Kit (JDK)

Java Development Kit (JDK) merupakan perangkat lunak yang digunakan untuk manajemen dan membangun berbagai aplikasi java. JDK merupakan *superset* dari JRE, berisikan segala sesuatu yang ada di JRE ditambahkan *compiler* dan *debugger* yang diperlukan untuk mengembangkan aplikasi. Bahasa pemrograman java menyediakan *library-library* standar yang telah di *compile* dan dapat langsung digunakan dalam implementasi pembuatan sebuah aplikasi. (Aryo, 2021)

### 2.3.8 Java

Java merupakan bahasa pemrograman yang populer, dikembangkan oleh Sun Microsystems. Salah satu penggunaan terbesar java adalah dalam pembuatan aplikasi *native* untuk android. bahasa pemrograman ini bersifat *multiplatform* yakni bahasa ini dapat digunakan diberbagai *platform*, seperti desktop, android dan bahkan untuk sistem operasi Linux. (Putro dkk, 2023)

Menurut Sumber (Muzhaffar, 2022) Kelebihan dari java adalah sebagai berikut :

- 1) Kode program dapat berjalan pada sistem operasi yang memiliki runtime environment dan telah banyak kesalahan yang dibenahi, serta keberadaan sebuah virtual machine yang mendukung stabilitas java.
- 2) Menyediakan fungsi untuk memonitor dan mengelola aplikasi yang biasanya mempunyai skala enterprise dengan menggunakan teknologi *java management extension*.
- 3) Menyediakan integrasi dengan fasilitas desktop untuk mengatasi keterbatasan yang dimiliki oleh aplikasi berbasis *browser*.
- 4) Mendukung penggunaan xml, diantaranya adalah *digital signature* dan *streaming API* untuk xml.

### 2.3.9 *SQLite*

*SQLite* adalah perpustakaan perangkat lunak yang menerapkan database *SQL* secara mandiri, tanpa memerlukan *server*, tanpa perlu melakukan konfigurasi, dan bersifat transaksional. *SQLite* adalah *engine database SQL* yang paling banyak digunakan di dunia. *SQLite* merupakan proyek yang bersifat *public domain* yang dikerjakan oleh D. Richard Hipp. Database *SQLite* memiliki fitur lengkap dengan banyak tabel, *indexs*, *trigger*, dan tampilan, serta tersimpan pada satu file tunggal dalam *hard-disk*. Format file database-nya bersifat *cross-platform* sehingga dapat dengan bebas menyalin database antara sistem 32-bit dan 64-bit atau antara arsitektur yang berbeda Platform. (Mansawan dkk, 2024)

*SQLite* adalah basis data *open source* yang tertanam ke dalam Android. *SQLite* mendukung fitur database relasional standar seperti sintaks *SQL*, fungsi transaksi dan juga fungsi *prepared statement*. Selain itu, hanya membutuhkan sedikit memori pada saat dijalankan. (Putra dkk, 2020)




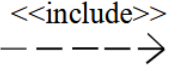
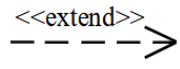
### 2.3.10 *UML (Unified Modeling Language)*

Pada perkembangan teknologi perangkat lunak, diperlukan adanya bahasa yang digunakan untuk memodelkan perangkat lunak yang akan dibuat dan perlu adanya standarisasi agar orang diberbagai negara dapat mengerti pemodelan perangkat lunak. “*Unified Modeling Language (UML)* adalah sebuah bahasa yang terdiri dari grafik atau gambar untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak yang berbasis *Object Oriented*. (Hasanah & Indriawan, 2021)

#### a) *Use Case Diagram*

*Use Case* menggambarkan kebutuhan atau sesuatu yang dapat dilakukan oleh aktor pada sistem yang dibuat. (Ayupratiwi dkk, 2025)




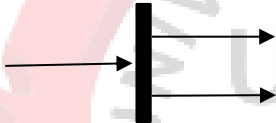


Tabel 2.4 Simbol dan Keterangan *Use Case Diagram*.

Simbol	Nama	Keterangan
	<i>Use Case</i>	Menggambarkan bagaimana seseorang akan menggunakan/memanfaatkan sistem.
	Aktor/ actor	Seseorang/sesuatu yang berinteraksi dengan sistem yang sedang kita kembangkan.
	<i>Association</i>	Relasi terjadi antara aktor dengan <i>usecase</i> biasanya berupa garis lurus dengan kepala panah disalah satu ujungnya.
	<i>Include Relationship</i>	Memungkinkan suatu <i>usecase</i> untuk menggunakan fungsionalitas yang disediakan oleh <i>usecase</i> yang lainnya.
	<i>Exclude Relationship</i>	Memungkinkan suatu <i>usecase</i> untuk menggunakan fungsionalitas yang disediakan oleh <i>usecase</i> yang lainnya.

#### b) *Activity Diagram*

*Activity Diagram* adalah bagian penting dari UML, yang menggambarkan aspek dinamis dari sistem. Logika *prosedural*, proses bisnis dan aliran kerja suatu bisnis bisa dengan mudah dideskripsikan dalam *activity diagram*, mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity diagram* bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. (Dharmawan dkk, 2021)

Tabel 2.5 Simbol *Activity Diagram*

Simbol	Nama	Keterangan
	Status Awal	Status awal aktivitas sistem, sebuah <i>diagram</i> aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	<i>Decision</i>	Asosiasi percabangan dimana jika ada pilihan aktivitas lebih dari satu.
	<i>Join</i>	Asosiasi penggabungan dimana lebih dari satu aktivitas digabungkan menjadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah <i>diagram</i> aktivitas memiliki sebuah status akhir.
	<i>Wimlane</i>	Memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

### c) *Class Diagram*

*Class diagram* yaitu salah satu jenis diagram pada UML yang digunakan untuk menampilkan kelas-kelas maupun paket-paket yang ada pada suatu sistem yang nantinya akan digunakan, jadi diagram ini dapat memberikan sebuah gambaran mengenai sistem maupun relasi-relasi yang terdapat pada sistem tersebut. *Class Diagram* adalah diagram yang menunjukkan *class-class* yang ada dari sebuah sistem dan hubungannya secara logika. *Class diagram* menggambarkan struktur statis dari sebuah sistem.

(Aditya dkk, 2021)

Tabel 2.6 Simbol *Class Diagram*

Simbol	Nama	Keterangan
	<i>Class</i>	Kelas pada struktur sistem
	<i>Association</i>	<i>Association</i> , dapat diartikan sebagai relasi. Digambarkan sebagai garis lurus antara dua kelas. Asosiasi biasanya juga disertai dengan <i>multiplicity</i> .
	<i>Generalization</i>	<i>Generalization</i> , relasi antar kelas dengan makna generalisasi-spesialisasi (umum khusus).
	<i>Cardinality</i>	Cardinality atau <i>multiplicity</i> adalah simbol yang menunjukkan jumlah keterhubungan dari satu kelas dengan kelas lainnya.
	<i>Dependency</i>	<i>Dependency</i> , Relasi antar kelas dengan makna kebergantungan antar kelas.

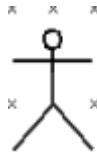



a) *Sequence Diagram*

*Sequence Diagram* adalah suatu diagram yang menjelaskan interaksi objek dan menunjukkan (memberi tanda atau petunjuk) komunikasi diantara objek-objek tersebut. *Sequence diagram* digunakan untuk menjelaskan perilaku pada sebuah skenario dan menggambarkan bagaimana entitas dan sistem berinteraksi, termasuk pesan yang dipakai saat interaksi. Semua pesan digambarkan dalam urutan pada eksekusi. *Diagram sequence*



menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. (Aditya dkk., 2021)

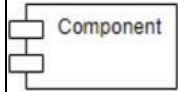

Tabel 2.7 Simbol *Sequence Diagram*

Simbol	Nama	Keterangan
	<i>Actor</i>	Seseorang atau sesuatu yang berinteraksi dengan sistem yang sedang kita kembangkan.
	aktivasi	Menggambarkan langkah-langkah dalam aliran kerja.
	<i>Object</i>	Menambah objek baru pada diagram.
	Pengulangan	Menggambarkan pesan yang menuju diri sendiri.

a) *Component Diagram*

*Component Diagram* adalah suatu gambaran yang menghubungkan komponen- komponen yang ada di dalam sistem tersebut memberikan cara kerja dan menampilkan *visual*. (Wildan dkk, 2021)




Tabel 2.8 Simbol *Component Diagram*

No	Gambar	Nama	Keterangan
1		<i>Component</i>	Sebuah component melambangkan sebuah entitas software dalam sebuah sistem
2		<i>Depedency</i>	Sebuah depedency digunakan untuk menotasikan relasi antara dua component

b) *Deployment Diagram*

Deployment diagram merupakan susunan secara fisik dari sebuah sistem, yang menegaskan bagian pada *software* berjalan pada *hardware* yang sedang bekerja. (Fetriany & Sobari, 2020)

Tabel 2.9 Simbol *Deployment Diagram*

No	Gambar	Nama	Keterangan
1		<i>Node</i>	<i>Node</i> merupakan bagian-bagian <i>hardware</i> dalam sebuah sistem
2		<i>Association</i>	<i>Association</i> menghubungkan dua <i>node</i> yang mengindikasikan jalur komunikasi antara elemen-elemen <i>hardware</i> .
3		<i>Component</i>	Pada <i>deployment diagram</i> , <i>component</i> – <i>component</i> yang ada diletakkan didalam <i>node</i> untuk memastikan keberadaan posisi mereka.

### 2.3.10 Pengujian *Black box*

Pengujian *Black box*, merupakan metode pengujian dimana data tersebut berasal dari pernyataan fungsional yang ditentukan tanpa memperhatikan struktur program akhir. Pengujian *Black box* mengacu pada uji fungsional, metode pengujian menekankan pada menjalankan fungsi pemeriksaan dan data keluaran. (Suharyanto E, 2022)

### 2.3.11 *Beta Testing*

*Beta testing* adalah pengujian dari sisi pengguna yang dilakukan dengan tujuan untuk mendapatkan umpan balik atau evaluasi dari pengguna. Salah satu cara

untuk mendapatkan umpan balik atau evaluasi dari pengguna atau *responden* adalah dengan menggunakan kuesioner. *Likert scale* adalah skala yang paling sering digunakan untuk mengukur pendapat *responden* (Menora Primasari dkk, 2023).

Rumus perhitungan yang digunakan beta testing adalah sebagai berikut :

$$Y = \left( \frac{X}{Skor Ideal} \right) \times 100$$

$$X = \sum (N \times R)$$

*Skor Ideal = nilai likert tertinggi × jumlah responden*

Sumber (Hakim dkk, 2024)

**Keterangan:**

Y = nilai presentase yang dicari

X = jumlah dari hasil perkalian setiap jawaban responden

N = nilai dari jawaban responden

R = jumlah responden