

BAB II

TINJAUAN PUSTAKA

2.1 Teori Dasar

2.1.1 *Game*

2.1.1.1 Pengertian *Game*

Game adalah teknik untuk menghilangkan rasa lelah dengan melakukan tugas yang dilakukan melalui pemahaman kognitif dan taktik yang harus digunakan untuk melibatkan mekanisme dan masalah lain yang sengaja dimaksudkan untuk memberikan kesenangan dalam bermain. Permainan adalah kegiatan yang melibatkan pemain, aturan, interaksi, dan tujuan atau sasaran (Sandy & Hidayat, 2019).

Sebuah *game* terdapat pemain berinteraksi dengan sistem dan perselisihan dihasilkan secara artifisial atau dengan rekayasa. Ada aturan dalam permainan yang mencoba mengatur perilaku pemain dan mempengaruhi jalannya permainan. Selain itu, ada tujuan yang harus dipenuhi peserta (Sandy & Hidayat, 2019).



Gambar 2.1 Definisi Game dan elemen pembentuknya

2.1.1.2 Game Mobile

Game mobile adalah *game* untuk ponsel menengah, *smartphone*, dan tablet, perangkat yang dapat dikenakan, PDA, dan pemutar media portabel. Saat ini, *game seluler* adalah salah satu yang dimainkan di *smartphone* dan tablet, terutama yang menjalankan sistem operasi Android dan iOS. *Game mobile* saat ini diunduh melalui toko aplikasi Google Playstore untuk platform Android dan toko aplikasi Apple Shop untuk platform iOS. (Sandy & Hidayat, 2019).

2.1.1.3 Jenis-jenis Game Mobile

Ada beberapa jenis *game* yang tersedia di dunia *game*. Berikut jenis *game mobile* yang bisa ditemui dan dimainkan pada umumnya, yaitu (Sandy & Hidayat, 2019) :

1. *Action*

Ini adalah salah satu jenis permainan yang paling populer. Sedangkan *game* bergenre ini biasanya menuntut kecepatan dan kelincahan pemain untuk menyelesaikannya.

2. *Adventure*

Ini termasuk permainan dengan plot yang belum sempurna. Setelah menyelesaikan sesi, peserta sering tidak dapat kembali ke turnamen sebelumnya.

3. *Action-Adventure*

Genre *action-adventure* merupakan campuran dari genre petualangan dan aksi. Pemain tetap disuguhkan dengan cerita terkini serta memiliki keterampilan memainkannya, seperti dalam *gameplay*.

4. RPG

Ada beberapa kategori sub-genre dalam genre RPG. Game RPG sering kali berbasis giliran dan memiliki tautan yang jelas. Sehingga setiap gerakan yang dilakukan pemain dapat dihitung dan dipertimbangkan kembali.

5. *Simulation*

Genre game simulasi akan menghadirkan skenario dunia nyata kepada pemain.

6. *Strategy*

Genre strategi memiliki jumlah sub-genre paling banyak. Di mana kemungkinan strategi berlimpah dan persaingan sengit di antara game lainnya. Permainan strategi membutuhkan pengambilan keputusan yang cepat dan pengembangan strategi sebelum permainan dimulai.

7. *Sports*

Genre *sports* juga menempati peringkat di antara genre game paling populer sepanjang masa. Permainan genre ini lebih tidak memihak dan memiliki basis penggemar yang besar.

8. *Idle Gaming*

Genre idle adalah genre game yang relatif baru di mana pemain tidak perlu mengerahkan banyak upaya untuk bermain. Memainkan game ini hanya membutuhkan kesabaran.

9. *Educational*

Educational game dapat digunakan untuk membantu proses pembelajaran. Pelatihan dalam berbagai subjek, dengan menggunakan *game* tersebut tidak membosankan bagi siswa. *Game* ini dilengkapi dengan fungsi tutorial, latihan dan ujian (Sutopo, 2020).

2.1.1.4 Game Edukasi (*Educational Game*)

Educational game atau edugame adalah *game* yang digunakan untuk membantu proses pembelajaran, pelatihan dan pengayaan pengetahuan. Permintaan besar akan edugame sebelum anak-anak bahkan sampai ke sekolah di sebagian besar negara (Sutopo, 2020).

Game edukatif adalah *game* dimaksudkan untuk menginstruksikan orang tentang topik tertentu atau untuk membantu orang lain dalam memperoleh pengetahuan saat bermain *game*. Aspek permainan edukatif dibangun di atas ide-ide pendidikan dasar yang mengintegrasikan unsur-unsur seperti inspirasi, kegembiraan, eksplorasi, motivasi, permainan, bakat, kebebasan, pendidikan, penalaran, antusiasme, individualitas, dan pengambilan keputusan (Setiawan et al., 2020)/

Educational game memiliki banyak manfaat, yaitu (Sutopo, 2020):

1. Kontribusi pada kemampuan manusia

Game memberikan kontribusi penting untuk akuisisi keterampilan vital dalam sejarah umat manusia, dan mendapatkan peran yang semakin penting dalam masyarakat.

2. Murah dan fleksibel

Mungkin sulit untuk membuat program pelatihan seluruh kelas yang besar.

Siswa menghabiskan sebagian besar waktu mereka dalam perjalanan ke sekolah untuk belajar.

3. Meningkatkan motivasi

Bahkan saat belajar, semua orang senang bermain dan bersenang-senang.

4. Merasakan keterlibatan dalam skenario

Skenario adalah perpaduan lokasi dan karakter yang memberikan kesan kepada siswa bahwa mereka berada dalam skenario asli. Mereka semakin dekat dengan realitas scenario, semakin banyak siswa akan terlibat dalam pembelajarannya.

5. Membantu tercapainya tujuan pembelajaran

Tujuan pembelajaran adalah unsur penting dari game yang menarik untuk memperoleh keterampilan baru.

6. Memberikan berbagai macam cara evaluasi

Salah satu keuntungan utama game untuk belajar dibandingkan dengan pembelajaran klasik adalah melakukan evaluasi diberbagai cara.

2.1.2 Android

Android adalah sistem operasi mobile yang digunakan pada smartphone berbasis Linux. Ini berisi sistem operasi, aplikasi dan middleware. Karena Android bersifat open source, siapa saja yang mengerjakannya bebas membuat aplikasi (Karman et al., 2019). Pengembang mengamati kondisi yang berubah dan memanfaatkan peluang untuk menghasilkan teknologi yang dibutuhkan banyak

orang, khususnya sistem operasi untuk ponsel yang juga dapat mempermudah pekerjaan masyarakat. Adapun hal-hal yang berusaha dicapai android sebagai berikut (Gunawan et al., 2021) :

1. Platform pengembangan perangkat lunak standar untuk ponsel
2. Konsistensi akses perangkat keras
3. Ekosistem ponsel yang kuat, kaya fitur, aman, dan hemat biaya

Android adalah sistem control Buka (Sumber Terbuka) dan Google kode versi lisensi Apache. Situasi ini menunjukkan bahwa Google memberdayakan orang hanya menyetel, mendistribusikan, membuat dan mengedit aplikasi Android. Keterbukaan dan dapat diperluas Kelola penggunaan sistem Android Jika Anda memakai perangkat Android perangkat elektronik lainnya seperti kalkulator penggajian, TV pintar, Buku pintar, dan kamera. Selain itu, Android pasang ke kacamata pintar, lihat tangan di dalam mobil, pemutar CD/DVD, pemutar media portabel, telepon VoIP dan bentuk penyajian media evaluasi lainnya (Ghea, 2012)

Tempat ke-4 berdasarkan Android Spesies antara lain (Nurjayati, 2015):

- a. Tutorial : Melihat data berbasis teks yang ditampilkan pada layar komputer, Grafik atau ilustrasi. Jika Anda menjawab pertanyaan, siswa akan didorong. Jawaban yang benar akan diberikan informasi tambahan, tetapi jika salah, peserta dapat mundur sejauh mungkin atau menerapkan Ide Restorasi.
- b. Lakukan dan Lakukan: Evaluasi ditampilkan sebagai *training*. Mengembangkan kekreatifan siswa.

- c. Simulasi: Sebuah Penelitian Baru simulasi dinamika proses berlaku di dunia nyata Berikan pengalaman tentang masalah "dunia" risiko nyata.
- d. Game pembelajaran (games Pendidikan): Perangkat Lunak mewakili model game edukasi serta aksi bersama dan Kreasi dan game bekas Kontrol permainan seperti keyboard, Mouse, gamepad, layar sentuh, dll.

2.1.3 Game Development Live Cycle (GDLC)

Game Development Life Cycle (GDLC) adalah melihat data berbasis teks yang ditampilkan pada layar komputer, Grafik atau ilustrasi. Jika Anda menjawab pertanyaan, siswa akan didorong. Respons yang benar akan diberikan informasi tambahan, tetapi jika salah, peserta dapat mundur sejauh mungkin atau menerapkan Ide Pemulihan (Krisdiawan & Darsanto, 2019). Dari 6 fase tersebut dapat dikelompokkan menjadi 3 proses utama yaitu:

1. Fase ide dan desain dari prosedur inisialisasi
2. Proses manufaktur dibagi menjadi tiga tahap: pra-produksi, produksi, dan inspeksi (Alpha dan Beta)
3. *Realease*.

2.2 Teori Khusus

2.2.1 Kosakata Bahasa Inggris (*English Vocabulary*)

Vocabulary adalah sekelompok kata yang diketahui oleh individu dan mungkin entitas lain, atau yang merupakan bagian dari bahasa tertentu Kosa kata pengguna didefinisikan sebagai kumpulan setiap kata yang dipahami atau diharapkan digunakan oleh individu tersebut untuk membuat kalimat baru. Kekayaan kosakata seseorang biasanya dilihat sebagai indikasi tingkat

pendidikannya (Arista, 2015).

Perluasan kosakata seseorang sering dianggap sebagai aspek penting dari proses penguasaan bahasa kedua atau peningkatan kemampuan orang lain dalam bahasa yang telah diperoleh sebelumnya (Arista, 2015).

Tiga hal yang paling penting dan perlu diingat dalam mempelajari bahasa inggris adalah :

1. Kata Benda (*Noun*) merupakan setiap nama orang, benda, dan objek seperti desa, sepupu, sayuran, jaket, universitas,dll.
2. Kata Sifat (*Adjective*) adalah kata yang menerangkan kata benda dan memberikan informasi tambahan tentang orang, tempat atau benda seperti keras, berwarna, manis, tenang, merah,dll.
3. Kata Kerja (*Verb*) merupakan kata yang menyatakan tindakan dan hal yang dilakukan diantaranya berolahraga, berbicara, berdiskusi,dll.

2.2.2 CorelDraw

CorelDraw adalah program grafik vektor yang dibuat dan ditawarkan oleh Corel Corporation di Ottawa, Kanada. Gambar format vektor adalah gambar yang menggunakan rumus matematika untuk menghasilkan serangkaian elemen garis dan kurva. Corel adalah program desain (perangkat lunak) yang menyediakan teknologi desain yang cukup baik (Sepriano, 2022).

CorelDraw adalah program manajemen gambar yang banyak digunakan dalam domain penerbitan, pencetakan, serta area lain yang membutuhkan visualisasi. CorelDraw sudah dirilis sebanyak 27 versi dimulai dari versi yang

pertama dirilis pada Januari tahun 1989 hingga saat ini versi yang ke 27 atau CorelDraw 2021 yang dirilis pada tanggal 3 Maret 2021 (Septino, 2021).

2.2.3 Construct 2

Construct 2 adalah alat pengembangan game yang menggunakan HTML5. Construct 2 tidak menggunakan bahasa pemrograman khusus; pengunjung tidak perlu mengetahui bahasa pemrograman yang juga relatif lebih rumit dan sulit dipahami untuk mengembangkan game dengan Construct 2 karena semua instruksi yang digunakan dalam game diatur dalam EvenSheet, yaitu Events dan Actions. (Sandy & Hidayat, 2019).

Scirra, sebuah perusahaan yang berlokasi di London, Inggris, membuat Construct 2. Construct 2 adalah contoh alat yang dapat digunakan untuk membuat game tanpa perlu membuat kode pemrograman (coding), karena mayoritas alasan untuk menghasilkan game dapat berupa dilakukan melalui menu. Game Construct 2 ditujukan untuk game 2D. Dengan menggunakan Construct 2, pengembang game dapat mendistribusikannya ke berbagai platform, termasuk Google Chrome Webstore, HTML5, situs web, Facebook Phonegap (Android), Windows 8, dan Windows Phone 8 (Fitria & Indra, 2020).

2.2.4 Unified Modelling Language (UML)

Unified Modeling Language (UML) adalah bahasa pemodelan standar yang umum di bidang rekayasa perangkat lunak. UML meliputi teknik notasi grafik untuk membuat model abstrak dari sistem tertentu. UML adalah penggambaran grafis untuk mendokumentasikan, mendeskripsikan, dan mengembangkan sistem

atau perangkat lunak. UML memberikan cara standar untuk membuat blueprint sistem, termasuk komponen konseptual seperti *actor*, proses bisnis, komponen sistem dan aktivitas. UML bertujuan untuk menjadi bahasa pemodelan standar yang dapat membuat model concurrent dan sistem terdistribusi (Sari & Utami, 2021:107)

Menurut penciptanya, UML adalah representasi simbolis untuk mendiskusikan, menentukan, membuat, memodelkan, dan mendokumentasikan fitur sistem. UML memprioritaskan penggunaan diagram untuk menjelaskan komponen sistem yang dimodelkan karena merupakan bahasa visual. UML bisa digunakan untuk: (Sugiarti, 2018:105)

1. Menggunakan use case dan aktor, menggambarkan batasan sistem dan fungsi sistem secara keseluruhan.
2. Menggunakan diagram interaksi untuk menggambarkan kegiatan atau proses kerja secara umum.
3. Menggunakan diagram kelas untuk menggambarkan penggambaran struktur statis sistem
4. Menggunakan diagram transisi keadaan, buat model perilaku yang mendefinisikan perilaku atau sifat suatu unit.
5. Dengan menggunakan diagram komponen dan pengembangan, jelaskan arsitektur aplikasi fisik.
6. Stereotip dapat digunakan untuk menyampaikan atau memperluas fungsionalitas.

Dalam bidang pengembangan sistem berorientasi objek, UML adalah alat

yang sangat dapat dipercaya. Ini karena UML menawarkan bahasa pemodelan visual yang memungkinkan pengembang sistem menghasilkan cetak biru standar untuk visi mereka. Diagram-diagram UML tersebut digunakan untuk: (Sugiarti, 2018:106)

1. Bertukar ide
2. Hasilkan ide-ide segar dan kemungkinan baru
3. Uji konsep dan buat prediksi
4. Kenali sistem dan interaksinya.

2.2.4.1 Use Case Diagram

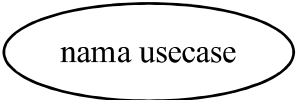
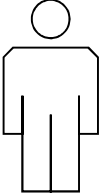
Use case adalah representasi dari fungsi sistem dari perspektif atau sudut pandang pengguna sistem. Kasus penggunaan menggambarkan "apa" yang dilakukan platform dan bagian-bagiannya, bukan "bagaimana" mereka berinteraksi satu sama lain. *Use case* berfungsi dengan memanfaatkan "skenario," yang merupakan pernyataan dari serangkaian tindakan yang menggambarkan apa yang dilakukan sistem dan sebaliknya. *Use case* diagram menunjukkan fungsionalitas sistem (*use case*), pengguna yang menjadi pengguna sistem (aktor), dan hubungan/koneksi antar pengguna dan fungsionalitas aplikasi (Sari & Utami, 2021:107).



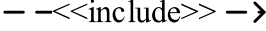
Use case diagram merupakan pemodelan untuk menjelaskan fungsionalitas dari aplikasi yang akan dibangun. Diagram *use case* menggambarkan komunikasi antara satu atau lebih aktor dan sistem yang akan dibuat. *Use case* diagram digunakan untuk dengan cepat menentukan

fungsionalitas apa yang ada di platform dan siapa yang berwenang untuk memanfaatkan fungsi-fungsi ini (Sugiarti, 2018:110).

Use case diagram dideskripsikan menggunakan banyak simbol, termasuk use case, aktor, dan koneksi. Ingatlah untuk tidak menentukan antarmuka pengguna, arsitektur sistem, kebutuhan non-fungsional, atau target kinerja. Sebuah use case diberi nama dengan cara yang sesederhana mungkin, dapat dimengerti, dan menggunakan kata kerja. Ada berbagai simbol dalam use case diagram yang harus dipahami. Berikut ini adalah simbol-simbol dalam diagram use case: (Sugiarti, 2018:111).

Tabel 2.1 Simbol *Use Case Diagram*

Simbol	Deskripsi
<p data-bbox="296 1108 422 1137"><i>Use case</i></p>  <p data-bbox="475 1218 657 1247">nama usecase</p>	<p data-bbox="852 1108 1326 1507">Fungsionalitas yang disediakan oleh platform sebagai komponen yang berkomunikasi langsung antara unit atau aktor, biasanya ditunjukkan dengan menggunakan kata di awal frasa nama kasus penggunaan.</p>
<p data-bbox="296 1552 384 1581">Aktor</p>  <p data-bbox="491 1832 641 1861">nama aktor</p>	<p data-bbox="852 1552 1326 1951">Orang, prosedur, atau perangkat lain yang berinteraksi dengan sistem informasi yang akan datang di luar sistem informasi yang akan datang. Jadi, sementara tanda aktor adalah gambar seseorang, itu tidak selalu</p>

Simbol	Deskripsi
	orang, yang biasanya ditunjukkan dengan menggunakan kata di depan dari frase nama aktor.
<p data-bbox="300 607 571 636"><i>Asosiasi/association</i></p> 	Interaksi antara aktor dan use case yang mengambil bagian dalam use case atau use case berinteraksi dengan aktor
<p data-bbox="300 875 395 904"><i>Extend</i></p> 	Koneksi antara <i>Use case</i> tambahan dan <i>Use case</i> di mana <i>Use case</i> tambahan dapat berdiri sendiri tanpa <i>Use case</i> tambahan. <i>Use case</i> baru biasanya memiliki nama awal yang sama dengan <i>Use case</i> asli, dan panah mengarah ke <i>Use case</i> utama.
<p data-bbox="300 1391 400 1420"><i>Include</i></p> 	<i>Use case</i> tambahan yang terkait dengan kasus penggunaan di mana <i>Use case</i> yang ditambahkan bergantung pada <i>Use case</i> ini untuk menyelesaikan fungsinya atau sebagai prasyarat untuk kasus penggunaan ini untuk diselesaikan. Terlibat dalam <i>Use case</i> memiliki dua implikasi utama.

Simbol	Deskripsi
	Sertakan memastikan bahwa <i>Use case</i> tambahan akan terus dipanggil setiap kali <i>Use case</i> tambahan dijalankan.

2.2.4.2 Activity Diagram

Activity diagram atau diagram aktivitas menggambarkan alur kerja atau operasi dari platform atau proses bisnis. Hal penting untuk diingat adalah fakta bahwa diagram aktivitas menggambarkan aktivitas sistem daripada apa yang dilakukan aktor, sehingga tindakan yang dapat dilakukan sistem (Sugiarti, 2018:134).



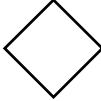


Diagram aktivitas dibuat untuk mewakili proses sistem dari tingkat bisnis ke tingkat operasional. Diagram aktivitas memiliki tujuan yang sama dengan diagram alur, namun tidak seperti diagram alur, diagram aktivitas dapat memungkinkan tindakan simultan, sedangkan diagram alur tidak (Sari & Utami, 2021:107).

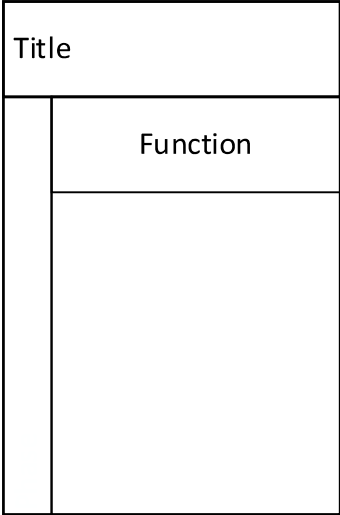
Activity diagram juga biasanya mendefinisikan hal-hal berikut (Sugiarti, 2018:134).

1. Desain proses bisnis di mana setiap urutan operasi yang dinyatakan adalah proses bisnis sistem yang ditentukan
2. Urutan atau kumpulan tampilan antarmuka sistem / pengguna di mana setiap tindakan dianggap memiliki desain antarmuka tampilan.
3. Desain pengujian di mana setiap diagram aktivitas memerlukan pengujian, yang harus direpresentasikan sebagai kasus uji.

Berikut adalah simbol *activity diagram* (Mesran et al., 2019:10)

Tabel 2.2 Simbol-simbol *Activity Diagram*

Simbol	Deskripsi
Status Awal 	Status awal dimiliki oleh diagram aktivitas
Aktivitas 	Kata kerja biasanya mengawali aktivitas
Percabangan/ <i>Decision</i> 	Ketika aktivitas lebih dari satu maka terdapat percabangan
Penggabungan/ <i>Join</i> 	Ketika lebih dari satu aktivitas memiliki sebuah status akhir maka ada penggabungan
Status akhir 	Status akhir dimiliki oleh diagram aktivitas
<i>Swimlane</i>	Pisahkan organisasi perusahaan

Simbol	Deskripsi
	terhadap aktivitas saat ini

2.2.4.3 Sequence Diagram

Sequence diagram mendokumentasikan komunikasi atau komunikasi antar-*class*. Diagram menggambarkan berbagai item dan komunikasi yang saling berhubungan dalam use case. Perhatikan bahwa dalam gambar ini, kelas dan aktor diatur secara vertikal di bagian atas gambar, dari kiri ke kanan, dengan garis yang menghubungkannya (Sari & Utami, 2021:107).

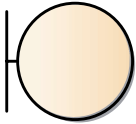
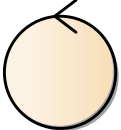

Sequence diagram menjelaskan aktivitas objek dalam *use case* dengan merinci siklus hidup objek serta pesan yang disampaikan dan dipertukarkan antar objek. Untuk merancang diagram urutan, Anda harus terlebih dahulu memahami properti dalam *use case*, serta fungsi kelas yang dihasilkan ke item tersebut..


Jumlah *sequence diagram* yang perlu digambar adalah sebanyak spesifikasi use case dengan prosesnya sendiri, atau perlu semua use case yang ditentukan berinteraksi dengan baik dengan pendekatan informasi yang disertakan dalam

sequence diagram. Akibatnya, semakin banyak use case yang ditentukan, maka semakin banyak sequence diagram yang harus dibuat. (Sugiarti, 2018:131).

Simbol pada *sequence diagram* adalah seperti berikut (Muharni, 2021:61)

Tabel 2.3 Simbol-simbol *Sequence Diagram*

Simbol	Fungsi
	<p>Entity Class, adalah salah satu komponen sistem yang mencakup sekumpulan kelas dalam format entitas yang membuat tampilan pertama sistem dan berfungsi sebagai dasar untuk membangun basis data.</p>
	<p>Boundary class, terdiri dari satu set kelas yang berfungsi sebagai antarmuka atau titik kontak dari satu atau lebih aktor dan sistem.</p>
	<p>Control class, sebuah objek yang berisi fungsionalitas aplikasi tetapi tidak bertanggung jawab kepada entitas. Pesan antara perbatasan dan entitas dikoordinasikan oleh objek kontrol.</p>
	<p>Message, symbol dalam ber kirim pesan</p>

Simbol	Fungsi
	antar class
	Recursive, ketika melakukan pengiriman pesan ke paa dirinya sendiri.
	Activation, Aktivasi mencerminkan eksekusi operasi objek; ukuran kotak ini kira-kira sama dengan durasi aktivasi operasi.
	Lifeline, Aktivasi garis putus-putus yang menghubungkan ke objek di mana saja di sepanjang garis hidup

2.2.4.4 Class Diagram

Inti dari pembuatan dan pengembangan berorientasi objek adalah diagram kelas, yang merupakan spesifikasi yang, ketika dipakai, menghasilkan objek. Kelas mewakili status sistem (atribut/properti) sementara juga menyediakan layanan untuk mengubah status (metode/fungsi). Diagram kelas menggambarkan konstruksi dan spesifikasi kelas, paket, dan objek, serta keterkaitannya seperti enkapsulasi, pewarisan asosiasi, dan sebagainya (Sari & Utami, 2021:107).

Class diagram menentukan struktur sistem dengan mengidentifikasi kelas yang akan dikembangkan untuk membangun sistem Seperti terlihat di bawah, kelas berisi karakteristik dan metode atau tindakan (Sugiarti, 2018:120).

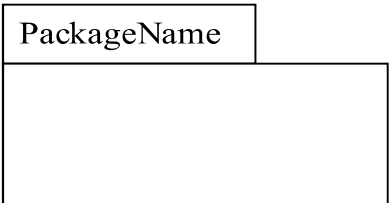
1. Atribut adalah parameter yang dimiliki oleh suatu kelas.
2. Atribut menjelaskan sifat dalam kotak kelas dengan salah satu baris teks.
3. Operasi atau metode adalah fungsi khusus kelas.

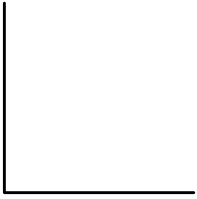
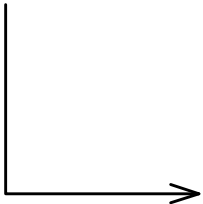
Class diagram menentukan organisasi dan deskripsi kelas, paket, dan objek, serta keterkaitannya termasuk penahanan, pewarisan, asosiasi, dan sebagainya. Sebuah kelas dibagi menjadi tiga bagian besar, yaitu sebagai berikut: (Sugiarti, 2018:121).

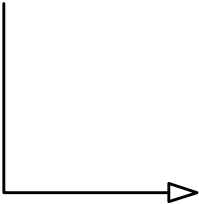
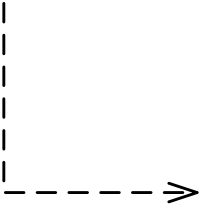
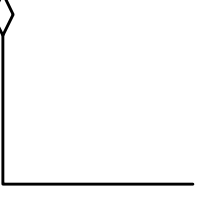
- 1) Nama adalah nama kelas.
- 2) Atribut adalah properti kelas. Atribut mewakili nilai maksimum yang bisa ada pada objek kelas.
- 3) Operasi adalah segala sesuatu yang dapat dilakukan oleh suatu kelas atau yang dapat dilakukan oleh kelas lain terhadap suatu kelas.

Simbol-simbol pada *diagram class* seperti berikut (Sugiarti, 2018:123)

Tabel 2.4 Simbol-simbol *Class Diagram*

Simbol	Deskripsi
	Bungkusan satu atau lebih dari class.
Operasi	Struktur sistem dari class

Simbol	Deskripsi
<div style="border: 1px solid black; padding: 5px;"> <p>nama kelas</p> <hr/> <p>+ Attribute 1 + Attribute 1</p> <hr style="border-top: 1px dashed black;"/> <p>+ Operation()</p> </div>	
<p>Antarmuka/interface</p> <div style="border: 1px solid black; padding: 5px;"> <p><<Interface>> InterfaceName</p> <hr/> <p>+ Attribute 1</p> <hr style="border-top: 1px dashed black;"/> <p>+ Attribute 1</p> </div>	<p>Sama dengan konsep interface dalam pemrograman berorientasi objek.</p>
<p>Asosiasi</p> 	<p>Asosiasi, hubungan antar kelas dengan makna yang luas, umumnya disertai dengan multiplisitas.</p>
<p>Asosiasi berarah / directed asosiasi</p> 	<p>Asosiasi antar kelas dengan pengertian satu kelas digunakan oleh kelas lain biasanya disertai dengan multiplisitas.</p>
<p>Generalisasi</p>	<p>Bermakna generalisasi-</p>

Simbol	Deskripsi
	spesialisasi (umum-khusus)
Kebergantungan/dependency 	Bermakna kebergantungan antarclass.
Agregasi 	Bermakna semua-bagian (whole-part)

3 *Blackbox Testing*

Black box testing adalah pemeriksaan yang dilakukan hanya dengan menyaksikan keluaran eksekusi melalui data uji dan menilai operasi perangkat lunak. Jadi, mirip dengan cara kita melihat kotak hitam, pengguna hanya bisa

melihat tampilan luarnya dan tidak tahu apa yang tersembunyi di balik penutup hitam. Mirip dengan pengujian kotak hitam, menilai hanya tampilan eksternal (antarmuka) dan berfungsi tanpa memahami apa yang sebenarnya terjadi di dalam prosedur terperinci ini (hanya mengetahui input dan output) (Habibi & Aprilian, 2020)

2.3 Penelitian Terdahulu

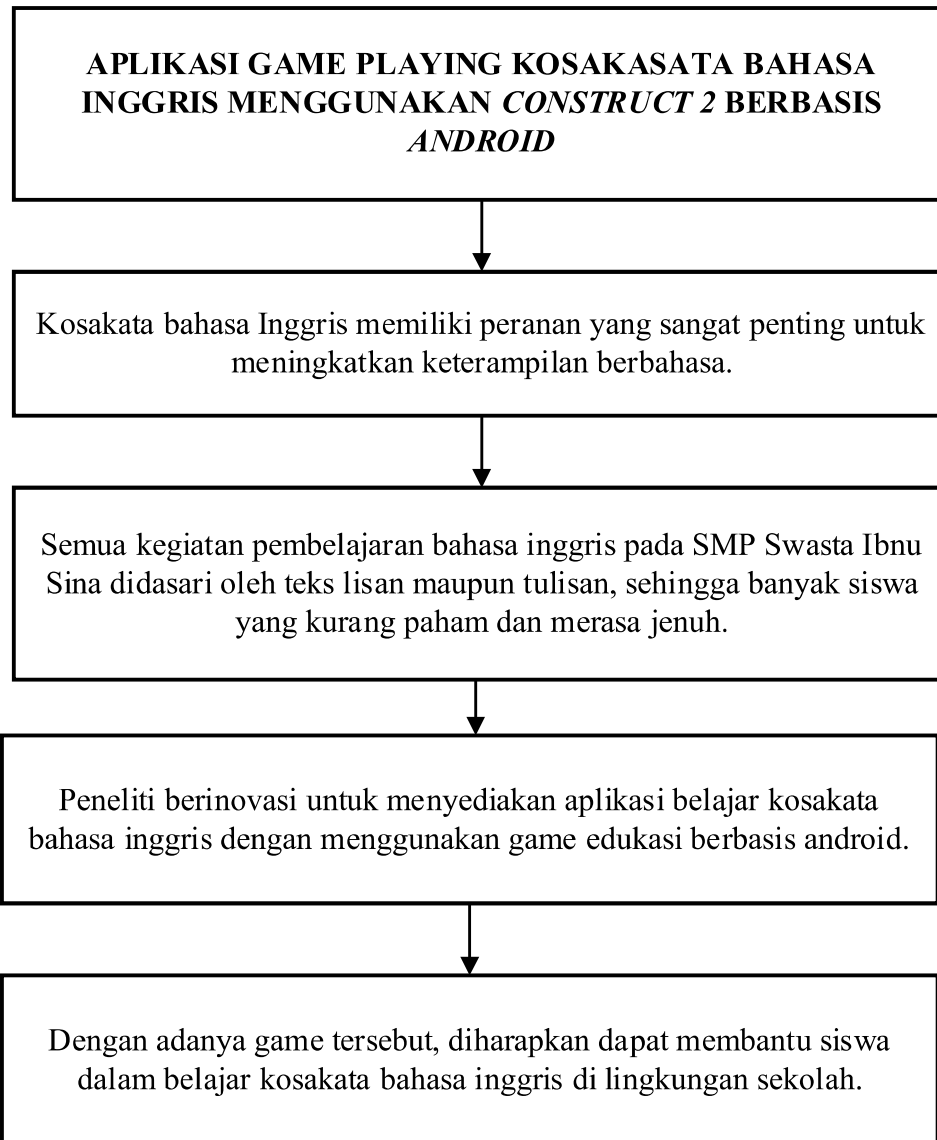
Tabel 2.5 Penelitian Terdahulu

No	Nama Author	Judul Penelitian	Hasil Penelitian
1.	Irsyadi et al. (2019)	Game Edukasi Pembelajaran Bahasa Inggris untuk Pengenalan Benda-Benda di Rumah bagi Siswa Kelas 4 Sekolah Dasar	Menurut hasil tes blackbox, program berjalan dengan baik, seperti yang diharapkan oleh pengembang. Rata-rata skor 'Sangat Setuju' dari kelima pernyataan yang diberikan dalam Uji Penerimaan Pengguna adalah 69,19 persen, menunjukkan tingkat penerimaan yang cukup tinggi dari pengguna aplikasi.
2.	Riska & Rahayu (2018)	Perancangan Game IDO untuk Pembelajaran Kosakata Bahasa Inggris Menggunakan Construct 2	Persentase yang menyatakan sangat setuju pada aspek penilaian tampilan permainan adalah 74 persen, persentase durasi waktu 50 persen, persentase keinginan belajar siswa 52 persen, persentase kemudahan belajar 49 persen, dan keunggulan permainan kosakata adalah 51 persen. Secara umum, responden setuju atau sangat setuju dengan penggunaan game English With Ido.
3.	Dias et al. (2021)	Perancangan Game Edukasi Sejarah Kemerdekaan Indonesia	Hasil menunjukkan bahwa Game Edukasi layak untuk digunakan dengan nilai akhir 88.04% dan tingkat pencapaian sangat baik sebagai Media Pembelajaran IPS, Khususnya Materi

		Menggunakan Aplikasi Construct 2 Berbasis Android	Sejarah Kemerdekaan Indonesia kelas V di SD GMTIT Kuanino 3.
4.	Posumah et al. (2021)	Game Pembelajaran Bahasa Inggris Berbasis Mobile Di SMK	Desain game pembelajaran 29nggri 29nggris menggunakan aplikasi Construct 2 dari Scirra. Game tersebut memiliki menu dan tampilan kontrol yang sangat mendasar, sehingga pemain tidak akan kesulitan memainkannya.
5.	Dewi & Ganing (2022)	Multimedia Interaktif Berbasis Game Edukasi Dua Dimensi Pada Muatan Bahasa Inggris Materi Pengenalan Kosakata	Berdasarkan hasil uji coba produk, subjek menyatakan game edukasi layak, termasuk tes ahli isi pembelajaran, yang mendapat persentase 100 persen termasuk keterampilan sangat khusus, sehingga dapat ditarik kesimpulan bahwa edutainment tersebut layak digunakan dalam pembelajaran. proses pendidikan.
6.	Kie & Simanjuntak (2022)	Perancangan Game Edukasi Menyusun Huruf Nama Hewan Berbasis Android	Aplikasi Game Edukasi Berbasis Android Siswa SD Negeri 001 kelas 1 secara efektif mengadopsi dan mengimplementasikan Konstruksi 2. Berdasarkan hasil tes, dapat disimpulkan bahwa bantuan penelitian pembelajaran berbasis game untuk mengenali nama binatang digunakan dengan benar sebagai instrumen penilaian. Berbeda dengan esensi penentuan identitas hewan yang melingkari, tema tematik
7.	Alfisyahri & Simanjuntak (2020)	Aplikasi Pembelajaran Bahasa Latin Tumbuh-Tumbuhan Berbasis	Ap Pengguna tampaknya menyukai tempat ini. Pembuatan aplikasi ini dapat mempermudah dan memperluas pengetahuan dan pemahaman tentang bahasa latin tumbuhan, sehingga menjadi media yang menarik dan

		Android	menghibur untuk pendidikan anak-anak SMP.
--	--	---------	---

2.4 Kerangka Pemikiran



Gambar 2.2 Kerangka Pemikiran