

**PEMBANGUNAN GAME EDUKASI KURO GAME PADA PLATFORM MOBILE
MENGUNAKAN METODE DIGITAL GAME BASED LEARNING**

TUGAS AKHIR

Disusun sebagai salah satu syarat untuk kelulusan
Program Strata 1, Program Studi Teknik Informatika,
Universitas Pasundan Bandung

oleh :

Hikmat Fadrial Ramadhan
Nrp. 19.304.0138



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS PASUNDAN
BANDUNG
AGUSTUS 2025**

**LEMBAR PENGESAHAN
LAPORAN TUGAS AKHIR**

Telah diujikan dan dipertahankan dalam Sidang Sarjana Program Studi Teknik Informatika Universitas Pasundan Bandung, pada hari dan tanggal sidang sesuai berita acara sidang, tugas akhir dari :

Nama : Hikmat Fadrial Ramadhan
Nrp. : 19.304.0138

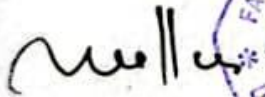
Dengan Judul :

“PEMBANGUNAN GAME EDUKASI KURO GAME PADA PLATFORM MOBILE
MENGUNAKAN METODE *DIGITAL GAME BASED LEARNING*”

Bandung, 21 Agustus 2025

Menyetujui,

Pembimbing Utama



(Mellia Liyanthy, ST., MT)



Pembimbing Pendamping



(Fajar Darmawan, ST., M.Kom)

LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR

Saya menyatakan dengan sesungguhnya bahwa :

1. Tugas Akhir ini adalah benar-benar asli dan belum pernah diajukan untuk mendapatkan gelar akademik, baik di Universitas Pasundan Bandung maupun di Perguruan Tinggi lainnya
2. Tugas Akhir ini merupakan gagasan, rumusan dan penelitian saya sendiri, tanpa bantuan pihak lain kecuali arahan dari tim Dosen Pembimbing
3. Dalam tugas akhir ini tidak terdapat karya atau pendapat orang lain, kecuali bagian-bagian tertentu dalam penulisan laporan Tugas Akhir yang saya kutip dari hasil karya orang lain telah dituliskan dalam sumbernya secara jelas sesuai dengan norma, kaidah, dan etika penulisan karya ilmiah, serta disebutkan dalam Daftar Pustaka pada tugas akhir ini
4. Kakas, perangkat lunak, dan alat bantu kerja lainnya yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab saya, bukan tanggung jawab Universitas Pasundan Bandung

Apabila di kemudian hari ditemukan seluruh atau sebagian laporan tugas akhir ini bukan hasil karya saya sendiri atau adanya plagiasi dalam bagian-bagian tertentu, saya bersedia menerima sanksi akademik, termasuk pencabutan gelar akademik yang saya sandang sesuai dengan norma yang berlaku di Universitas Pasundan, serta perundang-undangan lainnya.

Bandung, 21 Agustus 2025

Yang membuat pernyataan,



(Hikmat Fadrial Ramadhan)

NRP. 19.304.0138

ABSTRAK

Seiring berkembangnya teknologi, game digital telah mengalami banyak perubahan menjadi media edukatif yang mampu mendukung proses pembelajaran yang interaktif dan menyenangkan. Penelitian ini bertujuan untuk mengembangkan *Game* edukasi *Kuro Game*, sebuah game edukasi berbasis mobil yang dirancang untuk membantu siswa dalam memahami pelajaran Bahasa Inggris. *Game* ini dibangun menggunakan *Game Engine Unity* dan menerapkan pendekatan *Digital Game Based Learning* (DGBL), yang menggabungkan elemen hiburan dan elemen edukatif dalam sistem pembelajaran yang fleksibel dan mudah diakses kapan saja dan dimana saja.

Tujuan dari pengembangan ini adalah mengintegrasikan elemen-elemen *game* seperti interaksi dengan NPC, sistem kuis, dan navigasi antar *scene* ke dalam satu alur permainan yang mendukung pembelajaran Bahasa Inggris. Selain itu, struktur pemrograman dirancang secara modular agar alur eksekusi kode lebih efisien dan terorganisir.

Hasil dari implementasi dan pengujian menunjukkan bahwa *Kuro Game* mampu berjalan dengan baik pada perangkat mobile, dengan seluruh fitur utama bekerja secara fungsional dan responsif. Pengujian internal juga tidak menemukan error kritis, serta antarmuka *game* dapat menampilkan materi dan soal dengan baik. Dengan demikian, *Kuro Game* berhasil dikembangkan sesuai dengan tujuan perancangan, serta dapat dijalankan dengan baik pada perangkat mobile. *Game* ini menunjukkan bahwa penggabungan elemen edukatif dalam *game* digital dapat dilakukan secara terstruktur dan responsif menggunakan pendekatan *Digital Game Based Learning*.

Kata kunci : Game Edukasi, Unity Game Engine, Digital Game Based Learning, Pembelajaran Bahasa Inggris

ABSTRACT

With the advancement of technology, digital games have transformed into educational media capable of supporting interactive and enjoyable learning processes. This study aims to develop Kuro Game, an educational game based on mobile platforms designed to assist students in understanding English lessons. The game is built using the Unity Game Engine and applies the Digital Game Based Learning (DGBL) approach, which combines entertainment and educational elements into a flexible learning system accessible anytime and anywhere.

The development objectives are to integrate game elements such as interaction with NPCs, quiz systems, and scene navigation into a gameplay flow that supports English learning. Additionally, the programming structure is designed modularly to ensure efficient and organized code execution.

The implementation and testing results show that Kuro Game runs well on mobile devices, with all main features functioning properly and responsively. Internal testing found no critical errors, and the game interface effectively displays learning materials and quizzes. Thus, Kuro Game was successfully developed according to the design objectives and operates smoothly on mobile devices. This game demonstrates that the integration of educational elements in digital games can be done in a structured and responsive manner using the Digital Game Based Learning approach.

Keywords : *Educational Games, Unity Game Engine, Digital Game Based Learning, English Language Learning*

KATA PENGANTAR

Dengan memanjatkan puji dan syukur ke hadirat Allah Subhanahu Wata'ala, Tuhan yang Maha Pengasih dan Maha Penyayang, atas segala Rahmat, petunjuk, dan karunia-Nya yang telah menyertai langkah-langkah peneliti dalam menyelesaikan tugas akhir ini, Shalawat serta salam selalu tercurahkan kepada Baginda Nabi Besar Muhammad Salallahu Alaihi Wasallam, sebagai petunjuk jalan yang benar dalam hidup ini.

Tugas Akhir peneliti ini berjudul "*Pembangunan Game Edukasi Kuro Game pada Platform Mobile Menggunakan Metode Digital Game Based Learning*" Tujuan peneliti adalah untuk mengembangkan sebuah perangkat pembelajaran inovatif untuk membantu memperkaya pengalaman belajar Bahasa Inggris bagi para siswa melalui penggunaan teknologi dalam bentuk *game*, peneliti berharap dapat meningkatkan motivasi dan hasil belajar siswa.

Dalam pengembangan tugas akhir ini, peneliti telah mendapatkan bimbingan, dukungan, dan inspirasi dari berbagai pihak. Peneliti ingin mengucapkan terima kasih kepada:

1. Kepada orang tua dan keluarga yang telah mendo'akan keberlangsungan penelitian.
2. Kepada pembimbing utama Ibu Mellia Liyanthy, ST., MT
3. Kepada pembimbing pendamping Bapak Fajar Darmawan, ST., M.Kom
4. Kepada teman-teman seperjuangan satu proyek Tugas Akhir: Wahyu, Feri, Chandra
5. Kepada teman online saya yang telah memberikan dukungan moral, serta semangat selama pengerjaan Tugas Akhir: Ilyas, Rafli

Akhir kata, peneliti ingin menyampaikan harapan besar agar Tugas Akhir ini dapat bermanfaat bagi pembaca, khususnya dalam mengembangkan *Game Edukasi* yang lebih interaktif. Semoga karya ini dapat menjadi sumbangan kecil peneliti dalam Upaya meningkatkan kualitas Pendidikan dan industri *game* di Indonesia.

Terima kasih

Bandung, 21 Agustus 2025



Hikmat Fadrial Ramadhan

DAFTAR ISI

LEMBAR PENGESAHAN LAPORAN TUGAS AKHIR	
LEMBAR PERNYATAAN KEASLIAN TUGAS AKHIR	
ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR	iii
DAFTAR ISI	iv
DAFTAR GAMBAR	vi
DAFTAR TABEL	vii
DAFTAR LAMPIRAN	viii
DAFTAR SIMBOL	ix
DAFTAR ISTILAH	x
1 BAB I PENDAHULUAN	1-1
1.1 Latar Belakang	1-1
1.2 Identifikasi Masalah	1-2
1.3 Tujuan Tugas Akhir	1-2
1.4 Lingkup Tugas Akhir	1-2
1.5 Metodologi Penyelesaian Tugas Akhir	1-3
1.6 Sistematika Penulisan	1-4
BAB II LANDASAN TEORI DAN PENELITIAN TERDAHULU	2-1
2.1 Teori Pendukung	2-1
2.1.1 Game	2-1
2.1.2 Jenis-Jenis <i>Game</i>	2-1
2.1.3 <i>Game</i> Edukasi	2-3
2.1.4 Pembelajaran Bahasa Inggris	2-4
2.1.5 Unity	2-4
2.1.6 Bahasa Pemrograman C#	2-4
2.1.7 Metode Digital Game Based Learning (DGBL)	2-5
2.1.8 Android	2-7
2.1.9 Microsoft Visual Studio Code	2-7
2.1.10 Black Box Testing	2-7
2.2 Penelitian Terdahulu	2-8
BAB III SKEMA PENELITIAN	3-1
3.1 Alur Penyelesaian Tugas Akhir	3-1
3.2 Perumusan Masalah	3-5
3.2.1 Analisis Sebab Akibat	3-5
3.2.2 Solusi Masalah	3-7
3.3 Kerangka Berpikir Teoritis	3-7
3.3.1 Gambaran Produk TA	3-7

3.3.2	Skema Analisis Teori.....	3-8
3.4	Profil Penelitian.....	3-10
3.5	Analisis.....	3-11
3.5.1.	Pengguna Script.....	3-11
3.5.2.	Kebutuhan Script	3-11
BAB IV ANALISIS DAN PEMBANGUNAN GAME		4-1
4.1	Analisis Struktur Pemrograman	4-1
4.2	Perancangan Interaksi dan Kecerdasan NPC	4-1
4.3	Perancangan Interaksi Objek Interaktif.....	4-2
4.4	Flowchart	4-2
4.5	Pembangunan Game.....	4-4
4.5.1	Menu Utama	4-5
4.5.2	Interaksi NPC Satpam.....	4-6
4.5.3	Perpindahan Scene.....	4-9
4.5.4	Game Manager.....	4-11
4.5.5	GameplayData	4-13
4.5.6	GameManager_Materi.....	4-15
4.5.7	Materi Controller	4-19
4.5.8	Interactable Objects	4-22
4.5.9	Game Manager Quiz.....	4-24
4.5.10	Quiz Initiator	4-32
BAB V PENGUJIAN		5-1
5.1	Tujuan Pengujian	5-1
5.2	Metode Pengujian	5-1
5.3	Skenario dan Hasil Pengujian.....	5-2
5.4	Evaluasi Hasil Pengujian.....	5-3
6	BAB VI PENUTUP.....	6-1
6.1	Kesimpulan	6-1
6.2	Saran.....	6-1
DAFTAR PUSTAKA.....		1

DAFTAR GAMBAR

Gambar 1.1 Metodologi Digital Game Based Learning.....	1-3
Gambar 2.1 Digital Game Based learning [SOE16]	2-6
Gambar 3.1 Analisis Sebab Akibat	3-6
Gambar 3.2 Kerangka Berpikir Teoritis.....	3-7
Gambar 3.3 Skema Analisis Teori	3-9
Gambar 4.1 Flowchart Alur Game Kuro Game (Bagian 1)	4-3
Gambar 4.2 Flowchart Alur Game Kuro Game (Bagian 2)	4-4
Gambar 4.3 Menu Utama	4-6
Gambar 4.4 Interaksi NPC Satpam	4-8
Gambar 4.5 <i>NPC Dialogue Controller</i>	4-9
Gambar 4.6 <i>Inspector</i> Perpindahan Scene	4-10
Gambar 4.7 <i>MainManager</i>	4-13
Gambar 4.8 <i>GameplayData</i>	4-15
Gambar 4.9 <i>GameManager</i> Materi	4-18
Gambar 4.10 Tampilan Materi dalam Bahasa Inggris	4-21
Gambar 4.11 Tampilan Materi dalam Bahasa Indonesia	4-21
Gambar 4.12 <i>Materi Controller</i>	4-22
Gambar 4.13 Tampilan Interactable Object	4-24
Gambar 4.14 <i>Interactable Object</i>	4-24
Gambar 4.15 Tampilan Soal.....	4-30
Gambar 4.16 Tampilan Menjawab Soal Dengan Benar.....	4-30
Gambar 4.17 Tampilan Menjawab Soal Dengan Salah	4-31
Gambar 4.18 <i>GameManager Quiz</i>	4-32

DAFTAR TABEL

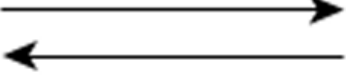
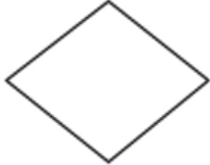
Tabel 2.1 Penelitian Terdahulu	2-8
Tabel 3.1 Alur Penyelesaian Tugas Akhir	3-2
Tabel 3.2 Solusi Masalah	3-7
Tabel 3.3 Kerangka Berpikir Teoritis	3-8
Tabel 3.4 Rencana Analisis	3-9
Tabel 4.1 Struktur Pemrograman	4-1
Tabel 5.1 Hasil Pengujian	5-2

DAFTAR LAMPIRAN

LAMPIRAN A PERIZINAN DAN PEMBAHASAN KURIKULUM.....	A-1
LAMPIRAN B BERITA ACARA	B-1
LAMPIRAN C TAMPILAN GAME.....	C-1
LAMPIRAN D GAME DESIGN DOCUMENT (GDD)	D-1
LAMPIRAN E QUISIONER	E-1

DAFTAR SIMBOL

Daftar Simbol Flowchart

No	Simbol	Nama	Deskripsi
1		Flow	Simbol yang digunakan untuk menggabungkan antara simbol yang satu dengan simbol yang lain. Simbol ini disebut juga dengan Connecting Line.
2		Terminator	Simbol yang menyatakan awal atau akhir suatu program.
3		Process	Simbol yang menyatakan suatu proses yang dilakukan.
4		Input/Output	Simbol yang menyatakan proses input atau output
5		Decision	Simbol yang menunjukkan kondisi tertentu yang akan menghasilkan dua kemungkinan jawaban, yaitu ya dan tidak.
6		Off-Page Reference	Simbol untuk keluar-masuk atau penyambungan proses dalam lembar kerja yang berbeda.

DAFTAR ISTILAH

No	Istilah Asing	Keterangan
1	Game	Game (permainan) secara umum adalah sebuah aktivitas dengan tujuan bersenang-senang, mengisi waktu luang, atau berolahraga ringan.
2	Game Design Document (GDD)	Game Design Document (GDD) ini sebagai gambaran bagaimana sebuah game itu akan dibuat.
3	Game Engine	Game Engine adalah perangkat lunak yang digunakan untuk merancang dan menjalankan elemen-elemen utama dalam pengembangan permainan digital.
4	Inspector	Inspector adalah panel antarmuka yang digunakan untuk melihat dan mengubah properti objek yang sedang dipilih di dalam scene.
5	Scene	Scene adalah ruang kerja untuk area virtual tempat berbagai objek game disusun, diatur, dan diatur interaksinya untuk membentuk satu bagian dari permainan.
6	Black Box Testing	Black Box Testing adalah metode pengujian perangkat lunak yang fokus pada fungsi atau keluaran sistem tanpa mengetahui struktur atau kode program di dalamnya.
7	Development	Development adalah proses perencanaan, perancangan, pembuatan, dan penyempurnaan suatu produk atau sistem untuk mencapai tujuan tertentu.
8	Script	Script adalah kumpulan perintah atau kode program yang digunakan untuk mengatur perilaku objek dalam suatu aplikasi atau permainan digital.
9	NPC	NPC (Non-Player Character) adalah karakter dalam permainan yang dikendalikan oleh sistem, bukan oleh pemain.
10	Pop-up	Pop-up adalah jendela atau pesan yang muncul secara tiba-tiba di layar untuk memberikan informasi atau pilihan kepada pengguna.
11	Collider	Collider adalah komponen dalam game engine yang digunakan untuk mendeteksi tabrakan atau interaksi fisik antara objek dan pemain.
12	Flowchart	Flowchart adalah diagram yang menggambarkan alur proses atau langkah-langkah dalam suatu sistem secara visual menggunakan simbol-simbol standar.
13	Game Object	Game Object adalah objek dasar dalam Unity yang menjadi wadah bagi komponen-komponen seperti model, script, dan collider untuk membentuk elemen dalam permainan.
12	First Person	First Person adalah sudut pandang dalam permainan atau cerita di mana pemain atau narator melihat dan mengalami dunia secara langsung melalui mata karakter utama.
13	Prototype	Prototype adalah versi awal dari produk atau sistem yang dibuat untuk menguji konsep, fungsi, dan desain sebelum pengembangan lebih lanjut.
14	Platform	Platform adalah lingkungan perangkat keras atau perangkat lunak tempat sebuah aplikasi atau permainan dijalankan, seperti PC, smartphone, atau konsol game.

No	Istilah Asing	Keterangan
15	Action Game	Action Game adalah jenis permainan video yang menekankan pada aksi dan refleks pemain.
16	Game Adventure	Game Adventure adalah genre game komputer yang mengandalkan ketangkasan pemainnya dalam memegang kontrol.
17	Racing Game	Racing Game merupakan suatu jenis dalam video game yang mensimulasikan pertandingan balapan sebagaimana halnya balapan dalam dunia nyata.
18	RPG	RPG (Role Playing Game) adalah salah satu jenis game dimana pemain mengontrol satu karakter tokoh utaman dalam sebuah cerita yang dimainkan. Sebagai tokoh utama, pemain dapat menjelajah, berinteraksi, dan berperan penuh dalam cerita tersebut.
19	Collectible	Collectible mengacu pada barang-barang yang dikumpulkan oleh orang karena nilai sentimental atau nilai investasi.
20	Storyline	Storyline adalah alur cerita yang akan dibuat pada sebuah karya animasi atau film sehingga jelas dan terarah.
21	Character Growth	Character Growth adalah istilah yang merujuk pada perkembangan karakter atau kepribadian seseorang dalam suatu cerita atau narasi
22	Strength	Strength adalah kata dalam bahasa Inggris yang memiliki arti kekuatan.
23	Intelligence	Intelligence adalah kemampuan untuk bertindak dengan mencapai suatu tujuan, untuk berpikir secara rasional dan untuk berhubungan dengan lingkungan secara efektif.
24	Hit Point	Hit Point adalah istilah yang digunakan dalam permainan video untuk menggambarkan jumlah kerusakan yang dapat ditangani oleh karakter atau objek sebelum kehilangan nyawa atau rusak.
25	Mana Point	Mana Point adalah istilah yang digunakan dalam permainan video untuk menggambarkan jumlah energi atau kekuatan yang dimiliki oleh karakter atau objek dalam game.
26	Classic RP	Classic RPG adalah genre permainan video yang menekankan pada cerita dan karakter, serta memungkinkan pemain untuk mengendalikan perkembangan karakter melalui pengambilan keputusan dalam permainan.
27	Turn-Based Battle	Turn-Based Battle adalah sistem pertarungan dalam permainan di mana setiap karakter atau pemain melakukan aksi secara bergiliran berdasarkan urutan yang ditentukan.
28	Hack-and-Slash	Istilah hack-and-slash adalah istilah yang digunakan untuk menggambarkan genre permainan video yang menekankan pada aksi dan pertempuran dengan menggunakan senjata tajam atau berat.
29	Tile-Based	Tile-Based adalah istilah yang digunakan dalam permainan video untuk menggambarkan jenis permainan di mana pergerakan dan

No	Istilah Asing	Keterangan
		tindakan unit atau karakter terjadi pada kotak-kotak atau grid yang terstruktur

BAB I PENDAHULUAN

1.1 Latar Belakang

Seiring berjalan nya waktu, teknologi terus berkembang, tanpa henti dan terus-menerus bergerak maju untuk menghadirkan berbagai inovasi baru di semua segikehidupan. Hal ini ditandai dengan adanya pengembangan teknologi di segala bidang, terutama untuk membantu mempermudah pekerjaan manusia. Selain itu, teknologi dalam bidang hiburan juga semakin berkembang pesat[PRA21].

Terutama bidang *game* digital sudah menjadi bagian dari aktivitas kehidupan manusia. Dukungan teknologi seluler dan computer telah membawa *game* digital sebagai salah satu bagian alat media pembelajaran atau Pendidikan. Terkaitannya dengan pembelajaran, *game-game* digital tidak hanya mampu menghadirkan kegiatan yang bersifat *immersive* dan menyenangkan, melainkan kegiatan yang terkait dengan bidang Pendidikan untuk mencapai tujuan tertentu. Memberikan manfaat sebagai alat Pendidikan dan pembelajaran, *game* digital telah mengalami transformasi menjadi *game* edukasi digital. *Game* edukasi ini memberikan peluang-peluang terhadap pembelajar dalam hal mengembangkan keterampilan diri, pengetahuan dan sikap melalui prinsip-prinsip dan fitur yang digunakan di dalam *gameplay*[PRA21].

Perkembangan teknologi pada saat ini memang sangat berdampak untuk kehidupan manusia dijamin sekarang. Banyak hal yang bisa kita rasakan dengan memanfaatkan kemajuan teknologi, tak terkecuali manfaat dalam bidang Pendidikan. Salah satunya adalah semakin banyak bimbingan *online* bermunculan dan memudahkan pelajar untuk mendapatkan materi yang mereka butuhkan disertai video-video yang dapat memberikan mereka penjelasan mengenai materi tersebut dimanapun dan kapanpun[CIP20].

Game edukasi merupakan sebuah permainan yang telah dirancang untuk mengajarkan pemainnya tentang topik tertentu, memperluas konsep, memperkuat pembangunan, memahami sebuah peristiwa sejarah atau budaya, atau membantu mereka dalam belajar keterampilan karena mereka bermain[WAJ20].

Pembelajaran dengan dukungan *game digital* diperlukan karena 3 alasan, pertama tumbuhnya generasi dengan keterampilan belajarnya berbasis *game*. Kedua, *game* memberikan kemungkinan untuk melaksanakan *experienced based learning* yang tersimulasi. Ketiga, *game* bisa dimanfaatkan untuk membangkitkan motivasi (*direction, intensity, permanence*) [RAM16].

Berdasarkan latar belakang ini, maka dibuatlah sebuah aplikasi game yang bernama Kuro *Game* yang menggunakan *Game Engine Unity*. Kuro *Game* dirancang untuk membantu siswa dalam mempelajari Bahasa Inggris. Penerapan *game* ini dipilih karena siswa sering

mengalami kesulitan dalam memahami materi yang disampaikan oleh guru. Dengan menggunakan *game* sebagai media pembelajaran, diharapkan siswa dapat mengatasi tantangan dalam memahami Bahasa Inggris dengan pendekatan yang lebih menarik dan interaktif, sehingga proses pembelajaran menjadi lebih menyenangkan.

Untuk mendukung fleksibilitas dan aksesibilitas pembelajaran, platform mobile dipilih untuk pengembangan *game* ini karena keunggulan fleksibilitas yang ditawarkannya. Platform mobile memungkinkan *game* dimainkan di mana saja dan kapan saja, menjadikannya sangat praktis bagi siswa yang ingin belajar dalam berbagai situasi dan lokasi. Fleksibilitas ini diharapkan dapat meningkatkan keterlibatan siswa dalam proses belajar.

1.2 Identifikasi Masalah

Berdasarkan latar belakang yang dipaparkan sebelumnya, maka permasalahan yang dimunculkan pada tugas akhir ini adalah :

1. Bagaimana mengintegrasikan dan menjalankan fungsi dari berbagai elemen *game* untuk menerapkan pembelajaran Bahasa Inggris?
2. Bagaimana mengelola alur eksekusi kode untuk memastikan bahwa fitur *game* berjalan sesuai dengan desain permainan dan mendukung pengalaman belajar Bahasa Inggris?
3. Bagaimana memastikan bahwa semua fungsi dan mekanika permainan yang diimplementasikan berfungsi dengan benar pada perangkat mobile?

1.3 Tujuan Tugas Akhir

Berdasarkan masalah yang dibahas, maka adapun tujuan tugas akhir sebagai berikut:

1. Mengintegrasikan dan menjalankan fungsi dari berbagai elemen *game* yang sudah dibuat sebelumnya oleh tim.
2. Mengelola alur eksekusi kode untuk memastikan bahwa fitur *game* berjalan sesuai dengan desain permainan dan mendukung pengalaman belajar Bahasa Inggris.
3. Memastikan bahwa semua fungsi dan mekanika permainan yang diimplementasikan berfungsi dengan benar dengan dilakukan pengujian terhadap *game*.

1.4 Lingkup Tugas Akhir

Dalam pembangunan media pembelajaran ini terdapat beberapa batasan masalah, yaitu :

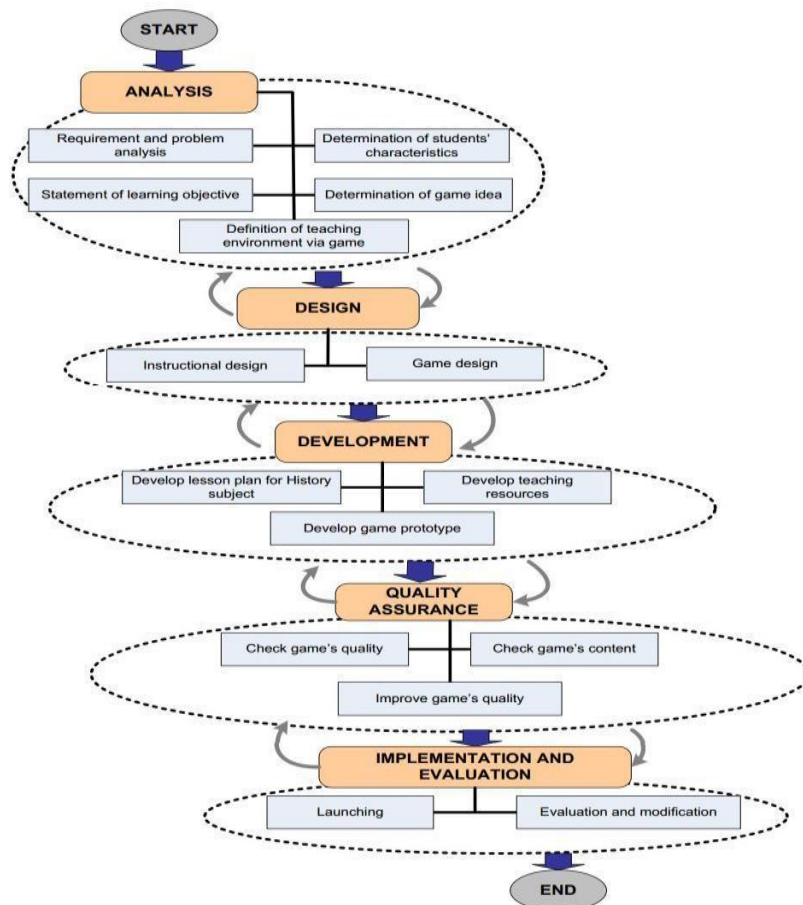
1. Aplikasi ini dikerjakan secara berkelompok dimana tugas akhir ini berfokus pada pemrograman dengan menggunakan metodologi *Digital Game Based Learning* pada fase Development.
2. Pembangunan *game* edukasi ini menggunakan *Game Engine Unity* dan Bahasa

pemrograman C#, untuk mengintegrasikan berbagai elemen *game* seperti UI, 3D Model, dan mekanika permainan.

3. Aplikasi *game* ini difokuskan pada pembuatan pengalaman pembelajaran Bahasa Inggris berbasis interaksi dalam *game*, dengan pengujian untuk memastikan bahwa semua fitur berfungsi dengan baik dan sesuai dengan desain pada perangkat *mobile*.

1.5 Metodologi Penyelesaian Tugas Akhir

Dalam penyusunan tugas akhir ini, terdapat langkah – langkah yang diterapkan dalam pengerjaan tugas akhir ini sebagai berikut :



Gambar 1.1 Metodologi Digital Game Based Learning

1.5.1 Identifikasi Masalah

Pada tahap ini adalah proses mengidentifikasi masalah yang sedang terjadi pada proses belajar mengajar melalui *Game Design Document* terkait *game* seperti apa yang dibutuhkan untuk mengatasi permasalahan belajar mengajar pada pelajaran Bahasa Inggris.

1.5.2 Studi Literatur

Pada tahap ini dilakukan pengumpulan data yang dilakukan terlebih dahulu,

yaitu melakukan penelusuran literatur yang terdapat pada buku, media, para pakar atau penelitian terdahulu mengenai bahasan yang dapat dilakukan sebagai sumber literatur untuk menyusun dasar teori yang akan digunakan dalam penelitian.

1.5.3 Analisis

Fase ini adalah proses menganalisis *Game Design Document* (GDD) untuk memahami mekanika *gameplay* dan logika yang harus diimplementasikan, sehingga dapat merancang kode yang sesuai dengan visi desain *game*. Hasil analisis ini berupa kode program yang mengimplementasikan mekanika *gameplay* dan logika interaksi pemain, yang akan digunakan dalam pengembangan *game* dan pengujian fitur.

1.5.4 Pembangunan Game

Dalam pembangunan game ini menggunakan metodologi yaitu metode *Digital Game Based Learning* (DGBL). Sebagai langkah-langkah pada game yang akan dibuat.

a. Mengembangkan Prototype Game

Tahap ini melibatkan pembuatan versi awal dari game untuk menguji konsep, mekanika, dan fitur utama. *Prototype* ini memungkinkan pengembang untuk mengevaluasi ide-ide desain, mengidentifikasi masalah pada *game* dan mendapatkan umpan balik untuk perbaikan sebelum memasuki fase pengembangan lebih lanjut.

1.5.5 Pengujian

Tahapan ini merupakan pengujian yang dilakukan terhadap *game* yang sudah dikembangkan untuk memastikan bahwa permainan berfungsi dengan baik. Tahapan ini membantu memastikan bahwa *game* yang dikembangkan tidak hanya berfungsi dengan baik secara teknis tetapi juga memberikan pengalaman yang sesuai dengan tujuan yang diinginkan bagi pemain.

1.6 Sistematika Penulisan

BAB 1 PENDAHULUAN

Bab ini memberikan penjelasan umum mengenai tugas akhir yang penulis lakukan. Penjelasan tersebut meliputi latar belakang masalah, identifikasi masalah, tujuan, ruang lingkup, metodologi penyelesaian tugas akhir, dan sistematika penulisan.

BAB 2 LANDASAN TEORI DAN PENELITIAN TERDAHULU

Dalam bab ini membahas beberapa teori yang mendasari penulisan dari tugas akhir ini. Teori umum yang digunakan meliputi teori lainnya yang mendasari penyusunan tugas akhir ini.

BAB 3 SKEMA PENELITIAN

Dalam bab ini membahas mengenai alur penyelesaian tugas akhir, analisis persoalan dan manfaat tugas akhir, dan kerangka pemikiran teoritis.

BAB 4 ANALISIS DAN PEMBANGUNAN GAME

Dalam bab ini membahas mengenai analisis untuk calon pengguna, rancangan game yang akan dibangun, serta menjelaskan tools apa saja yang digunakan dalam pembuatan game.

BAB 5 PENGUJIAN

Dalam bab ini membahas mengenai implementasi terkait data dan asset-asset yang sudah dikumpulkan sesuai dengan metode yang digunakan, serta dilakukan pengujian terhadap aplikasi yang telah dibuat.

BAB 6 PENUTUP

Dalam bab ini membahas mengenai kesimpulan dari keseluruhan tugas akhir dalam bentuk paragraph serta saran yang dapat menjadi acuan dalam penelitian atau pengembangan selanjutnya dari game.

BAB II

LANDASAN TEORI DAN PENELITIAN TERDAHULU

2.1 Teori Pendukung

Subbab ini berisi sejumlah teori atau konsep yang relevan digunakan dalam penelitian atau tugas akhir. Semua konsep pendukung di bab 3 harus disampaikan sumber dan literturnya pada bab 2 ini.

2.1.1 Game

Secara bahasa *game* berasal dari bahasa inggris yang berarti permainan. Banyak teori yang mengungkapkan tentang pengertian *game*. Seperti yang dikatakan Jasson dalam bukunya bahwa *game* merupakan suatu sistem atau program dimana satu atau lebih permainan mengambil keputusan melalui kendali pada obyek di dalam *game* untuk suatu tujuan tertentu[RIA19].

Teori lain mengatakan bahwa *game* merupakan salah satu media hiburan yang menjadi pilihan masyarakat untuk menghilangkan kejenuhan atau hanya untuk sekedar mengisi waktu luang. Selain sebagai media hiburan, *game* juga dapat meningkatkan perkembangan otak seseorang, contohnya adalah permainan catur yang dapat meingkatkan konsentrasi otak[RIA19].

Berdasarkan pengertian tersebut dapat dikatakan bahwa *game* atau permainan merupakan suatu media yang digunakan untuk tujuan hiburan dimana pemainnya melakukan suatu kegiatan guna mencapai tujuan tertentu yang biasanya dapat dilakukan secara mandiri atau *single player* maupun lebih dari satu pemain atau *multiplayer*[RIA19].

2.1.2 Jenis-Jenis Game

Jenis atau genre adalah istilah yang menggambarkan jenis gameplay yang dimiliki oleh sebuah game. Dengan adanya genre dalam sebuah game, orang akan menjadi mudah dalam mengelompokkan/ mengkategorikan game, genre game tidak bersifat *mutual-exclusive* yang artinya suatu game bisa merupakan kombinasi dari berbagai game yang ada [APR23].

A. Action

Action game mengutamakan kordinasi antara mata dan kelincahan tangan dan menuntut pemain untuk senantiasa focus menghadap layar. *Action game* umumnya beririsan dengan genre game lainnya, seperti *RPG* ataupun *Adventure*.

B. Adventure

Game Adventure merupakan game yang menitikberatkan pada eksplorasi dan petualangan yang disajikan pada pemain. *Game Adventure* biasanya dilandasi dengan adanya banyak *platform* dalam game untuk diloncati, berbagai *collectible* untuk dicari dan dilengkapi, dan tantangan-tantangan dalam bentuk teka-teki, kuis atau jebakan.

C. Racing

Game bergenre *racing* adalah game yang menyajikan balapan untuk pemain. Sebagian besar dari *game racing* menggunakan kendaraan mobil. Meskipun demikian, terdapat juga *game racing* yang menyediakan balapan menggunakan kendaraan motor, sepeda, pesawat, motorboat, dll. Tujuan utama *game racing* yaitu pemain untuk mencapai garis finish pertama.

D. Role-Playing Game (RPG)

Game bergenre *RPG* merupakan game yang mengedepankan *storyline* dan *character growth*. *RPG* ditandai dengan dunia game yang luas, *storyline* yang Panjang, beberapa *customizable* dan *character attribute* seperti *strength* (STR), *intelligence* (INT), *hit point* (HP), *mana point* (MP), dll. Sekarang telah muncul berbagai varian baru dalam genre game *RPG* seperti *game action-RPG* yang memanfaatkan *hack-and-slash/TPS*, *strategy-RPG* yang menyajikan battle dalam bentuk *tile-based* dan *classic RPG* yang masih menganut sistem *turn-based battle*.

E. Building Simulator (Tycoon)

Genre game *building simulator* menyajikan kebebasan pada *player* untuk membangun bangunan, kota, Gedung, perusahaan, dll sembari melakukan *resource management* yang terbatas. *Player* tidak dapat membangun jika kehabisan *resource*, maka diperlukan adanya perhitungan antara *cost* yang dihasilkan dengan pengeluaran dalam membangun sebuah Gedung dalam memainkan *building simulator*.

F. Simulator

Game simulator adalah game yang mensimulasikan kegiatan nyata yang terjadi di dunia nyata seperti mengendarai mobil, menjadi pilot pesawat, bercocok tanam, melakukan operasi pembedahan pasien, masak-memasak, dll. Dengan tujuan dari game simulator ini adalah untuk mendidik atau mengilustrasikan kepada *player* tentang proses bagaimana melakukan sesuatu kegiatan melalui ilustrasi *game*.

G. Sport

Genre game *sport* merupakan game yang menyajikan *gameplay* mengenai olahraga. Sebagian besar dari *sport game* menggunakan olahraga yang benar-benar ada di dunia nyata, tetapi ada sedikit game yang menyajikan olahraga fantasi yang tidak ada di dunia nyata. Oleh

karena menggunakan media olahraga seperti di dunia nyata, maka *sport game* dapat dibilang dekat dengan genre game simulator akan tetap yang membuat perbedaannya yaitu *sport game* adalah simulator di bidang olahraga.

H. Strategy

Genre *game strategy* menitikberatkan pada taktik/strategi dan *resource management* yang diperlukan untuk *player* memenangkan suatu pertempuran. Umumnya *game strategy* mengangkat tema *conquest* atau penaklukan kekuasaan. *Game strategy* menuntut pemain untuk berpikir lebih dan merencanakan Langkah-langkah sebaik-baiknya yang akan diambil sebelum bergerak.

I. Education

Genre *game education* adalah bentuk game yang dirancang untuk memberikan pengalaman dalam bidang Pendidikan atau memberikan pengalaman belajar kepada pemainnya, termasuk permainan tradisional dan modern yang diberi muatan Pendidikan dan pengajaran.

2.1.3 Game Edukasi

Game edukasi adalah permainan yang dirancang atau dibuat untuk merangsang daya pikir termasuk meingkatkan konsentrasi dan memecah masalah (Handriyantini, 2009). *Game* edukasi adalah salah satu jenis media yang digunakan untuk memberikan pengajaran, menambang pengetahuan penggunanya melalui suatu media unik dan menarik. *Game* edukasi dibuat dengan tujuan spesifik sebagai alat pendidikan, untuk belajar mengenal warna, mengenal huruf dan angka, matematika, sampai belajar bahasa asing[RIF15].

Game dengan tujuan edukasi seperti ini dapat digunakan sebagai salah satu media edukasi yang memiliki pola pembelajaran *learning by doing*. Berdasarkan pola yang dimiliki oleh *game* tersebut, pemain dituntut untuk belajar sehingga dapat menyelesaikan permasalahan yang ada. Status *game*, intruksi, dan *tools* yang disediakan oleh *game* akan membimbing pemain secara aktif untuk menggali informasi sehingga dapat memperkaya pengetahuan dan strategi saat bermain[RIF15].

Berdasarkan hasil penelitian penelitian sebelumnya, tidak diragukan lagi bahwa *game* edukasi dapat menunjang proses pendidikan (Marsh, dkk, 2005; Clark, 2006). *Game* edukasi unggul dalam beberapa aspek jika dibandingkan dengan metode pembelajaran konvensional. Salah satu keunggulan yang signifikan adalah adanya animasi yang dapat meningkatkan daya ingat sehingga anak dapat menyimpan materi pelajaran dalam waktu yang lebih lama dibandingkan dengan metode pengajaran konvensional (Clark, 2006)[RIF15].

Game edukasi adalah permainan yang digunakan dalam proses pembelajaran dan dalam permainan tersebut mengandung unsur mendidik atau nilai-nilai pendidikan[RIF15].

2.1.4 Pembelajaran Bahasa Inggris

Dalam konteks Pendidikan, Bahasa ini berfungsi sebagai alat berkomunikasi guna mengakses, menyimpan, dan berbagi informasi. Dalam keseharian dapat berfungsi sebagai alat untuk menjalin hubungan interpersonal, bertukar informasi dan menikmati aspek keindahan Bahasa tersebut [SUN].

Berpijak pada fungsinya, maka tujuan pengajaran Bahasa Inggris dalam kurikulum yang berlaku saat ini mencakup [SUN]:

1. Mengembangkan kemampuan berkomunikasi dalam Bahasa tersebut baik lisan maupun tulisan, kemampuan tersebut meliputi listening (mendengarkan), speaking (berbicara), reading (membaca), dan writing (menulis).
2. Menumbuhkan kesadaran akan hakikat dan pentingnya Bahasa Inggris sebagai salah satu Bahasa asing untuk menjadi alat utama belajar.
3. Mengembangkan pemahaman keterkaitan antara Bahasa dan budaya serta memperluas cakrawala budaya.

Dengan demikian siswa memiliki wawasan lintas budaya dan melibatkan diri dalam keragaman budaya.

2.1.5 Unity

Unity Technologies dibangun di tahun 2004 oleh David Helgason, Nicholas Prancis dan Joachim Ante, *Game Engine* ini dibangun atas dasar kepedulian mereka terhadap indie *developer* yang tidak bisa membeli *game engine* karena terlalu mahal. Fokus perusahaan ini adalah membuat sebuah perangkat lunak yang bisa digunakan oleh semua orang khususnya untuk membangun sebuah *game*. Di tahun 2009, Unity diluncurkan secara gratis di april 2012, Unity mencapai popularitas tertinggi dengan lebih dari 1 juta *developer* terdaftar di seluruh dunia[DEB19].

Selain bisa didapatkan secara gratis, Unity juga adalah sebuah *game engine* yang memungkinkan untuk digunakan baik perseorangan mau pun tim, untuk membuat sebuah *game* dengan cepat. Secara default, unity telah diatur untuk pembuatan *game* bergenre *First Person Shooting* (FPS), namun Unity juga bisa digunakan untuk *game* bergenre *Role Playing Game* (RPG), dan *Real Time Strategy* (RTS). Selain itu, Unity merupakan sebuah *engine multiplatform* yang memungkinkan *game* yang dibangun dapat *dipublish* untuk berbagai *platform* seperti Windows, Mac, Android, IOS, PS3, dan juga Wii[DEB19].

2.1.6 Bahasa Pemrograman C#

C# merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan

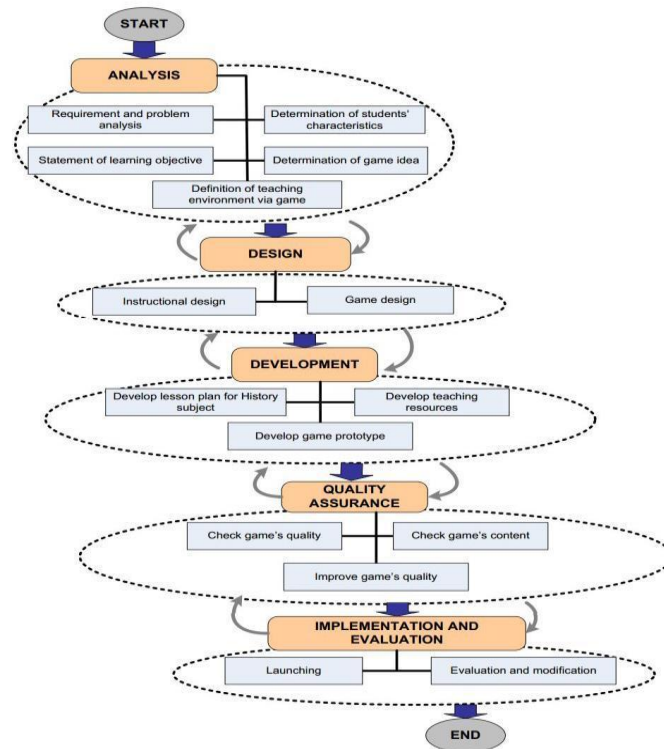
oleh Microsoft sebagai bagian dari inisiatif kerangka .NET Framework. Bahasa pemrograman ini dibuat berbasiskan bahasa C++ yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti Java, Delphi, Visual Basic, dan lain-lain, dengan beberapa penyederhanaan. Menurut standar ECMA-334 *C# Language Specification*, nama C# terdiri atas sebuah huruf Latin C (U+0043) yang diikuti oleh tanda pagar yang menandakan angka # (U+0023). Tanda pagar # yang digunakan memang bukan tanda kres dalam seni musik (U+266F *keyboard* standar[ERI16]).

C# merupakan salah satu aplikasi yang memiliki kemampuan dalam penguatan Framework.NET. C# dibuat sejalan dengan perkembangan Framework.NET, C# sendiri dikembangkan oleh Microsoft. Dalam penerapannya C-Sharp (C#) menjanjikan produktifitas, fleksibilitas serta kemudahan yang ada dari aplikasi sebelumnya yaitu Visual Basic, Java dan C++. C# mengadopsi kemampuan dari penggabungan aplikasi sebelumnya (mempelajari-c) Microsoft membuat C# seiring dengan pembuatan Framework.NET. *Chief Architect* dalam pembuatan C# adalah Anders Hejlsberg yang sebelumnya berperan dalam pembuatan Borland Delphi dan Turbo Pascal. C# menjanjikan produktifitas dan kemudahan yang ada di Visual Basic dengan kemampuan dan fleksibilitas yang ada di C/C++. Menurut spesifikasi bahasanya, “C# (pronounced “C Sharp”) is a simple, modern, object oriented, and type-safe programming language. It will immediately be familiar to C and C++ programmers. C# combines the high productivity of Rapid Application Development (RAD) languages and the raw of C++.”. Untuk mencapai produktifitas tinggi ini konsep-konsep sulit C++ disederhanakan dan fitur-fitur baru ditambahkan. Hal ini mungkin terasa mirip dengan Java, karena itulah C# bisa dianggap sebagai sepupu Java (Yahya & Nur, 2018)[PUT21].

2.1.7 Metode Digital Game Based Learning (DGBL)

Digital Game Based Learning (DGBL) atau pembelajaran yang berbasis *game* digital adalah suatu metode yang menggunakan teknologi *game* era masa kini, sebuah *game* digital apa pun bisa saja disebut sebagai media atau alat pembelajaran jika terdapat unsur kognitif pembelajaran di dalamnya[ANG21].

Demikian dapat dikatakan dengan menerapkan metode *Digital Game Based Learning* (DGBL), belajar bukan hanya menarik namun juga memotivasi karena bentuk medianya sebuah *game* yang di dalamnya ada sebuah misi atau permasalahan kedalam bentuk permainan yang menarik minat siswa[ANG21].



Gambar 2.1 Digital Game Based learning [SOE16]

Berdasarkan Gambar 2.1, dapat dijelaskan bahwa :

1. Fase *Analysis*

- a. Analisis masalah dan kebutuhan
- b. Penentuan karakteristik siswa
- c. Tujuan dari pembelajaran
- d. Menentukan ide game
- e. Mendefinisikan lingkungan mengajar melalui game

2. Fase *Design*

- a. Desain intruksional
- b. Desain game

3. Fase *Development*

- a. Mengembangkan rencana pembelajaran
- b. Mengembangkan sumber daya pengejaran
- c. Mengembangkan *prototype* game

4. Fase *Quality Assurance*

- a. Mengecek kualitas game
- b. Mengecek konten game
- c. Memperbaiki kualitas game

5. Fase *Implementation and Evaluation*

- a. Perilisan
- b. Evaluasi dan modifikasi

2.1.8 Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet. Android awalnya dikembangkan oleh *android, inc.* Dengan dukungan finansial dari *Google* yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007. Bersamaan dengan didirikannya *Open Handset Alliance*, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Ponsel *android* pertama mulai dijual pada bulan Oktober 2008, (Ramdani, 2017)[NIN20].

2.1.9 Microsoft Visual Studio Code

Microsoft Visual Studio merupakan sebuah perangkat lunak lengkap (*suite*) yang dapat digunakan untuk melakukan pengembangan aplikasi, baik itu aplikasi bisnis, aplikasi personal, ataupun komponen aplikasinya, dalam bentuk aplikasi console, aplikasi Windows, ataupun aplikasi Web. Visual Studio mencakup kompiler, SDK, Integrated Development Environment (IDE), dan dokumentasi (umumnya berupa MSDN Library). Kompiler yang dimasukkan ke dalam paket Visual 6 Studio antara lain Visual C++, Visual C#, Visual Basic, Visual Basic.NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe (Kahlert, 2016)[SAK].

2.1.10 Black Box Testing

Pengujian *black box* memiliki peran penting dalam pengujian perangkat lunak yaitu untuk memvalidasi fungsi keseluruhan sistem apakah telah bekerja dengan baik. Pengujian *black box* bersifat dinamis[PAR20].

Seseorang yang menguji perangkat lunak dengan metode pengujian *black box* tidak perlu memiliki pengetahuan pemrograman / struktur dalam perangkat lunak. Penguji yang menggunakan pengujian *black box* tidak memiliki akses untuk mengetahui kode sumber dan arsitektur sistem, hanya melalui antarmuka dengan memberikan input dan memeriksa output tanpa mengetahui bagaimana input dioperasikan hingga menjadi sebuah output. Pemodelan *black box* tergantung pada akurasi yang diinginkan serta struktur opsional dipilih untuk memetakan data yang diukur dari sistem termasuk input dan output[PAR20].

Tingkat keberhasilan suatu pengujian dapat dilihat dari hasil akhir suatu perangkat lunak yang sudah sesuai mulai dari spesifikasi kebutuhan untuk kepuasan pengguna, skenario, dan rancangan. Langkah pertama dalam pengembangan suatu perangkat lunak sebaiknya pilih metode pengujian yang tepat. Maka dari itu perencanaan pengujian dapat dimulai sejak awal proses perangkat lunak dikembangkan[PAR20].

Pengujian black box harus membuat kasus uji dengan dua perbandingan antara benar atau salah. Contoh pada saat pengguna masuk pada aplikasi (perangkat lunak), maka uji kasusnya:

- Jika pengguna memasukkan username dan password yang benar.
- Jika pengguna memasukkan username yang salah dan password yang benar, atau sebaliknya[PAR20].

Tujuan Black Box Testing adalah untuk mencari:

1. Fungsi yang salah atau hilang.
2. Kesalahan antarmuka.
3. Kesalahan dalam struktur data atau akses database eksternal.
4. Kesalahan kinerja.
5. Inisialisasi dan kesalahan terminasi.
6. Validasi fungsional.
7. Kesensitifan sistem terhadap nilai input tertentu.
8. Batasan suatu data[PAR20].

2.2 Penelitian Terdahulu

Penelitian terdahulu yang menjadi inspirasi diantaranya :

Tabel 2.1 Penelitian Terdahulu

No	Peneliti	Hasil Penelitian	Alasan digunakan	Komponen apa yang digunakan
1	Hayu Ika Anggraini, Nurhayati, Shirly Rizki Kusumaningrum	Penerapan media pembelajaran game matematika dengan menggunakan metode <i>Digital Game Based Learning</i> (DGBL)	Memadukan konsep belajar dan bermain dengan penerapan metode <i>Digital Game Based Learning</i> (DGBL)	Metode pengembangan game edukasi
2	Noviana Sandra Debbie	Penelitian ini berhasil membuat <i>game</i> yang berguna untuk pembelajaran.	Tahapan dalam pembuatan <i>game</i> menggunakan aplikasi Unity	Pengembangan <i>game</i> menggunakan Unity
3	Chrisnaji Banindra Yudha	Terdapat perbedaan yang signifikan keefektifan game edukasi	Pembangunan <i>game</i> edukasi pembelajaran	Pengaruh game edukasi dengan

No	Peneliti	Hasil Penelitian	Alasan digunakan	Komponen apa yang digunakan
		pembelajaran matematika berbasis android dan tanpa media dalam pembelajaran perkalian di SDN Rempoa 4 Tangerang Selatan. Game edukasi pembelajaran matematika berbasis android lebih efektif dibandingkan tanpa media.	matematika dengan berbasis android	berbasis android sebagai media pembelajaran
4	Raden Muhammad Mirza Prasetyo, Hadi Syaputra, Widya Cholil, Siti Sauda.	Pengembangan aplikasi dan Algoritma penyelesaian masalah dengan Flowchart.	Tahapan dalam pembangunan <i>game</i> menggunakan Flowchart sebagai penyelesaian dalam masalah	Algoritma penyelesaian masalah dengan Flowchart
5	Tioya Nurdiansyah, Suhud, Akip Suhendar	Penelitian ini berhasil membuat game dengan menggunakan bahasa pemrograman C# dalam pembangunannya	Tahapan dalam pembangunan <i>game</i> menggunakan C# sebagai bahasa pemrograman yang digunakan	Pengembangan <i>game</i> menggunakan Bahasa Pemrograman C#

Penjelasan penelitian terdahulu sebagai berikut :

1. Literatur pertama, diambil karena memiliki alur pembangunan *game* dengan menggunakan *Digital Game Based Learning* (DGBL)
2. Literatur kedua, diambil karena memiliki kesamaan dalam pembangunan *game* dengan menggunakan Unity.
3. Literatur ketiga, diambil karena memiliki kesamaan dalam pembangunan *game* berbasis Android.
4. Literatur keempat, diambil karena dalam pembangunan *game* tersebut dibutuhkan flowchart sebagai alur dari awal *game* hingga *game* selesai dimainkan.
5. Literatur kelima, diambil karena dalam pembangunan *game* tersebut menggunakan bahasa C# dan memiliki hasil pengujian terhadap kelayakan *game* edukasi bahasa Arab

BAB III

SKEMA PENELITIAN

Skema penelitian berisi gambaran detail terkait gagasan tugas akhir, serta penjelasan detail tentang produk yang ingin dibuat. Bab ini dimulai dengan memunjukkan alur detail pengerjaan tugas akhir, menyampaikan analisis persoalan dan solusi yang ditawarkan, hingga menyajikan kerangka berpikir teoritis.

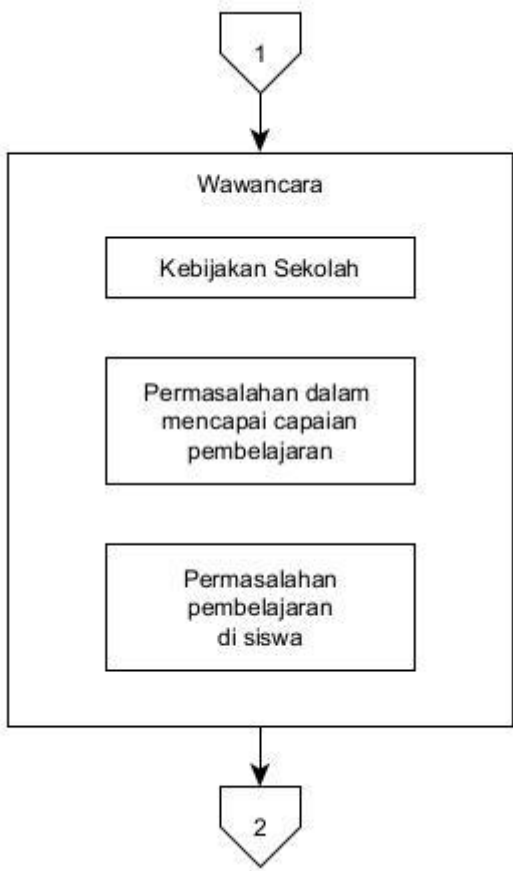
3.1 Alur Penyelesaian Tugas Akhir

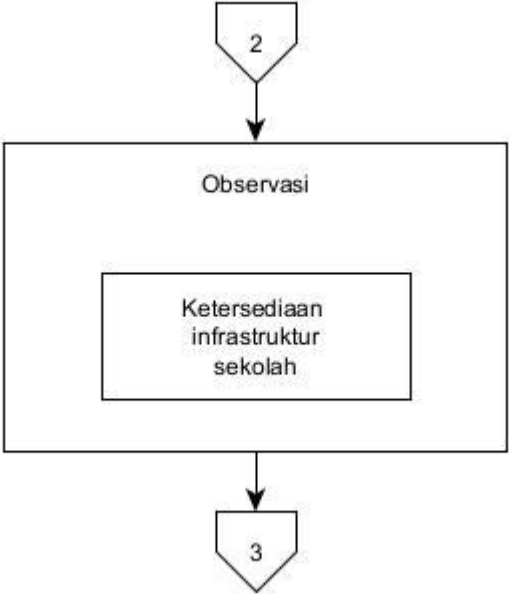
Alur penelitian merupakan kerangka pengerjaan tugas akhir yang menjelaskan mengenai tahapan dan langkah detail pengerjaan tugas akhir. Pada tugas akhir ini, alur penelitian dibagi menjadi tiga tahap komponen, yaitu :

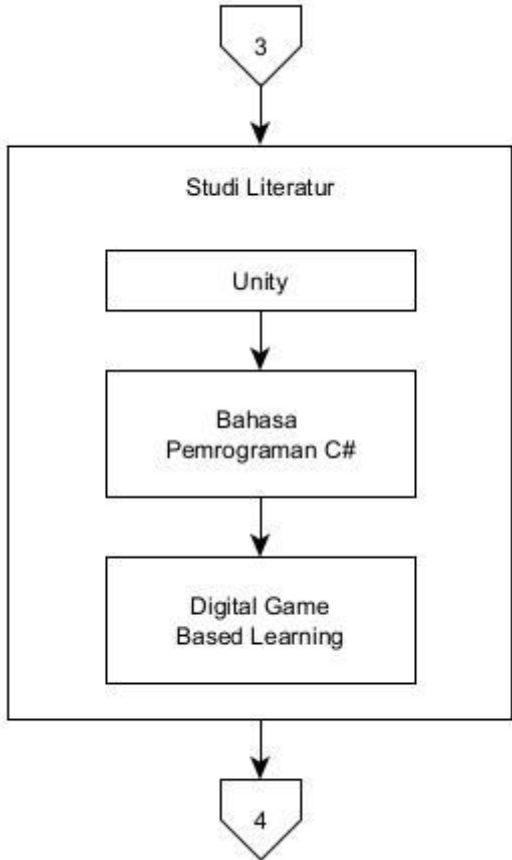
1. Tahap, adalah komponen yang berisi gambaran yang akan dikerjakan dalam tugas akhir.
2. Langkah, adalah komponen yang berisi hal-hal yang akan dikerjakan dalam tugas akhir yang disesuaikan dengan metode yang digunakan.
3. Literatur atau referensi, adalah komponen yang berisi sumber pengetahuan yang dijadikan landasan dalam mengerjakan tugas akhir.

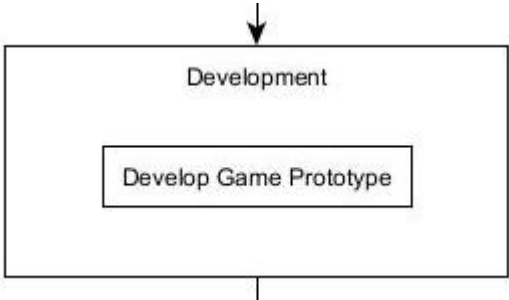
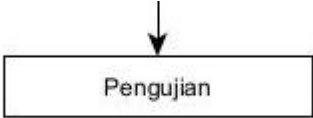
Terdapat lima tahap dalam pengerjaan tugas akhir ini, meliputi : tahap perencanaan, tahap analisis, tahap perancangan, tahap implementasi, dan tahap penanaman simpulan. Alur penelitian selengkapnya dapat dilihat pada tabel 3.1.

Tabel 3.1 Alur Penyelesaian Tugas Akhir

Tahap dan Hasil	Langkah Penelitian	Literatur & Referensi
Tahap 1 : Wawancara narasumber yang terlibat seperti Kepala Sekolah, Guru Bahasa Inggris, dan quisioner kepada siswa untuk mengetahui kebijakan sekolah dan permasalahan yang dihadapi oleh guru. Hasil : Data Kebijakan dan permasalahan yang dihadapi. Kontribusi : berguna untuk tahap analisis	 <pre> graph TD 1{{1}} --> Wawancara subgraph Wawancara_Box [Wawancara] A[Kebijakan Sekolah] B[Permasalahan dalam mencapai capaian pembelajaran] C[Permasalahan pembelajaran di siswa] end Wawancara_Box --> 2{{2}} </pre>	

Tahap dan Hasil	Langkah Penelitian	Literatur & Referensi
Tahap 2: Observasi terhadap ketersediaan infrastruktur sekolah dan spesifikasi perangkat yang digunakan. Hasil: Data ketersediaan perangkat keras beserta spesifikasi minimumnya. Kontribusi: Berguna untuk fase analisis.	 <pre> graph TD 2{{2}} --> Observasi[Observasi] Observasi --> 3{{3}} subgraph Observasi_Box [Observasi] Ketersediaan[Ketersediaan infrastruktur sekolah] end </pre>	

Tahap dan Hasil	Langkah Penelitian	Literatur & Referensi
Tahap 3: Studi literatur (mengetahui teori yang mendukung penyelesaian produk tugas akhir) Hasil: Teori-teori pendukung penyelesaian produk Tugas Akhir Kontribusi: Berguna untuk pembangunan pada game.	 <pre> graph TD 3{{3}} --> SL[Studi Literatur] subgraph SL_Box [Studi Literatur] direction TB U[Unity] --> BCP[Bahasa Pemrograman C#] BCP --> DGBL[Digital Game Based Learning] end SL_Box --> 4{{4}} </pre>	1. <i>Unity</i> adalah sebuah <i>game engine</i> yang memungkinkan untuk digunakan baik perseorangan mau pun tim, untuk membuat sebuah game dengan cepat. 2. C# merupakan Bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif kerangka .NET Framework. 3. DGBL-ID (<i>Digital Game Based Learning</i>) sebagai arsitektur perancangan Game Edukasi (Soeheri, 2016)

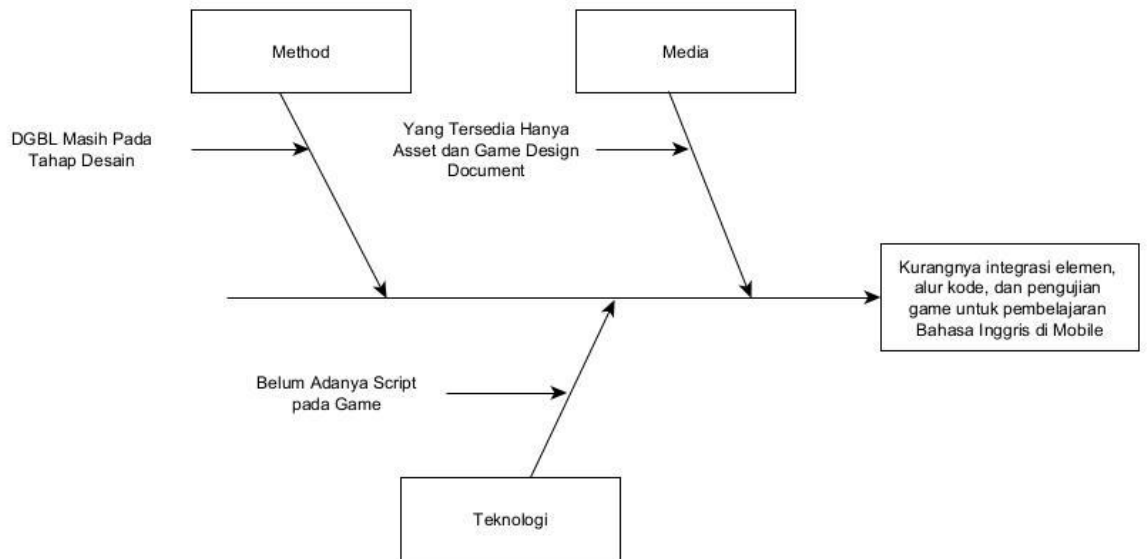
Tahap dan Hasil	Langkah Penelitian	Literatur & Referensi
Tahap 4: Development Hasil: Game yang sudah dapat dimainkan. Kontribusi: Berguna untuk memastikan game dapat dimainkan, diuji, diperbaiki, dan disempurnakan sebelum rilis	 <pre> graph TD A[] --> B[Development] B --> C[Develop Game Prototype] C --> D[] style A fill:none,stroke:none style D fill:none,stroke:none </pre>	
Tahap 5: Pengujian Hasil: Mengavaluasi gameplay, menemukan bug, memastikan fitur berjalan sesuai dengan desain yang telah ditentukan. Kontribusi: Memastikan game stabil dan minim bug.	 <pre> graph TD A[] --> B[Pengujian] style A fill:none,stroke:none </pre>	

3.2 Perumusan Masalah

Persoalan yang dikemukakan pada bab satu adalah hasil pengamatan beserta studi pada literatur terkait dengan pembangunan game 3d dengan menggunakan Unity dan Bahasa Pemrograman C# untuk *game mobile* yang digunakan sebagai penelitian yang diangkat oleh penulis. Adapun perumusan masalah dari tugas akhir penulis lakukan, terbagi menjadi dua kelompok yaitu analisis sebab akibat dan solusi masalah, sebagai berikut :

3.2.1 Analisis Sebab Akibat

Analisis peran pembangunan game 3D menggunakan Unity dan Bahasa Pemrograman C# untuk *game mobile* bertujuan untuk mengetahui ketepatan solusi terhadap permasalahan pada tugas akhir. Sehingga permasalahan yang telah disebutkan dapat diselesaikan dengan solusi yang tepat. Tahap analisis pada bagian ini menggunakan *cause and effect* diagram atau disebut juga dengan nama *fishbone* diagram. Berikut ini merupakan gambar *fishbone* diagram analisis persoalan pembangunan game 3D menggunakan Unity dan bahasa pemrograman C# untuk *game mobile* yang dapat dilihat pada gambar 3.1.



Gambar 3.1 Analisis Sebab Akibat

Berdasarkan hasil analisis yang digambarkan pada diagram *fishbone* pada gambar 3.1 diatas, dapat menjelaskan penyebab terjadi dan mengapa akibat – akibat tersebut dapat terjadi, adapun penyebab yang muncul dari hasil identifikasi, sebagai berikut :

1. Asset

Dari sisi asset terdapat permasalahan bahwa asset yang telah disediakan oleh tim baik 3D maupun 2D yaitu bagian UI, belum bisa berfungsi sesuai dengan game desain yang telah dirancang oleh game desainer, sehingga membuat asset-asset tersebut belum diimplementasikan pada game.

2. Alur Permainan

Dari sisi alur permainan terdapat permasalahan bahwa alur game belum berjalan sesuai dengan yang dirancang oleh game desainer, karena belum adanya script untuk menjalankan fungsi sesuai dengan alur tersebut. Sehingga dibutuhkan script untuk menjalankan setiap fungsinya agar game dapat berjalan dari awal permainan hingga permainan selesai.

3. Pengujian

Pengujian dilakukan untuk memastikan setiap fungsi yang ditulis berjalan sesuai dengan desain dan tidak menimbulkan error saat dijalankan. Ini mencakup pengecekan alur logika, integrasi asset dengan script, serta kestabilan fitur saat game dijalankan diperangkat asli.

4. Scripting

Dari sisi scripting terdapat permasalahan karena semua fungsi di dalam game harus dibuat dari awal dan disesuaikan agar bisa berjalan sesuai dengan rancangan permainan. Proses ini membutuhkan pengaturan urutan jalannya fitur, bagaimana setiap bagian saling terhubung, dan memastikan semua itu bisa

dikendalikan melalui kode secara tepat.

3.2.2 Solusi Masalah

Berdasarkan akar masalah pada subbab sebelumnya, berikutnya disampaikan apa saja solusi yang mungkin dari setiap penyebab tersebut.

Tabel 3.2 Solusi Masalah

NO	Penyebab	Solusi
1	DGBL masih pada tahap desain	Proses pada DGBL dilanjutkan pada tahap Development dan Pengujian
2	Yang tersedia hanya Asset dan <i>Game Design Document</i>	Menambahkan implementasi script dan fitur permainan
3	Belum adanya script pada game	Pembuatan script pada <i>Igame</i>

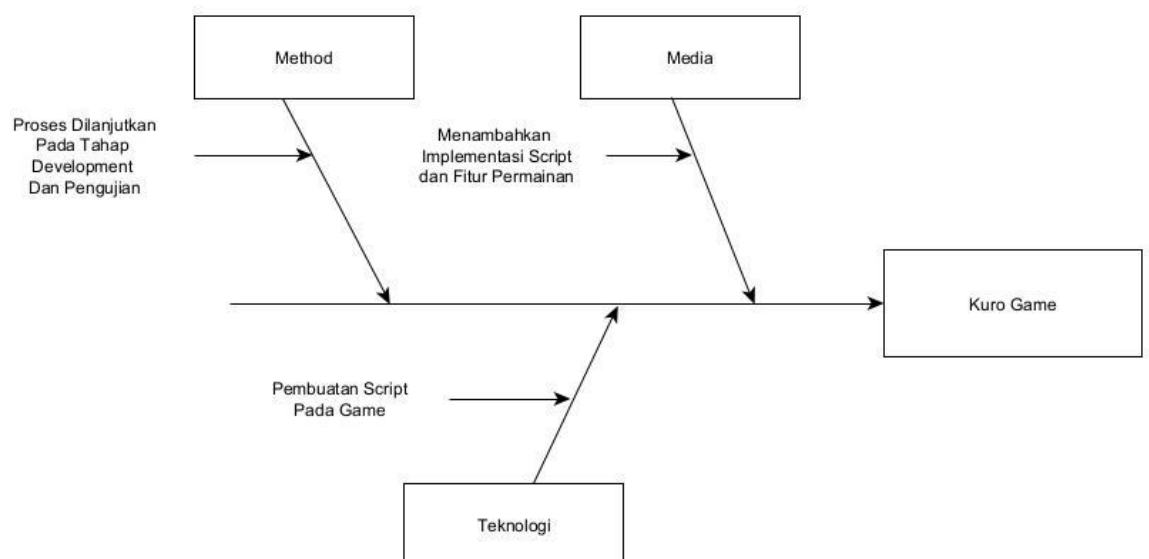
3.3 Kerangka Berpikir Teoritis

Hasil dari pengamatan berserta studi pada literatur terkait dengan pembangunan *game* 3D untuk mobile menggunakan bahasa pemrograman *C#* yang digunakan sebagai penelitian. Adapun kerangka berfikir teoritis dari Tugas Akhir, sebagai berikut :

3.3.1 Gambaran Produk TA

Kerangka pemikiran teoritis arsitektur informasi bertujuan untuk mengetahui teori-teori yang dijadikan landasanyang akan digunakan untuk menjelaskan fenomena yang diteliti serta untuk menjelaskan keterkaitan antara konsep-konsep yang akan digunakan dalam penelitian ini.

Berikut ini adalah kerangka pemikiran teoritis dari penelitian tugas akhir yang direpresentasikan menggunakan *fishbone* diagram pada gambar 3.2.



Gambar 3.2 Kerangka Berpikir Teoritis

Tabel 3.3 menjelaskan detail kerangka pemikiran perancangan arsitektur informasi pada tugas akhir ini.

Tabel 3.3 Kerangka Berpikir Teoritis

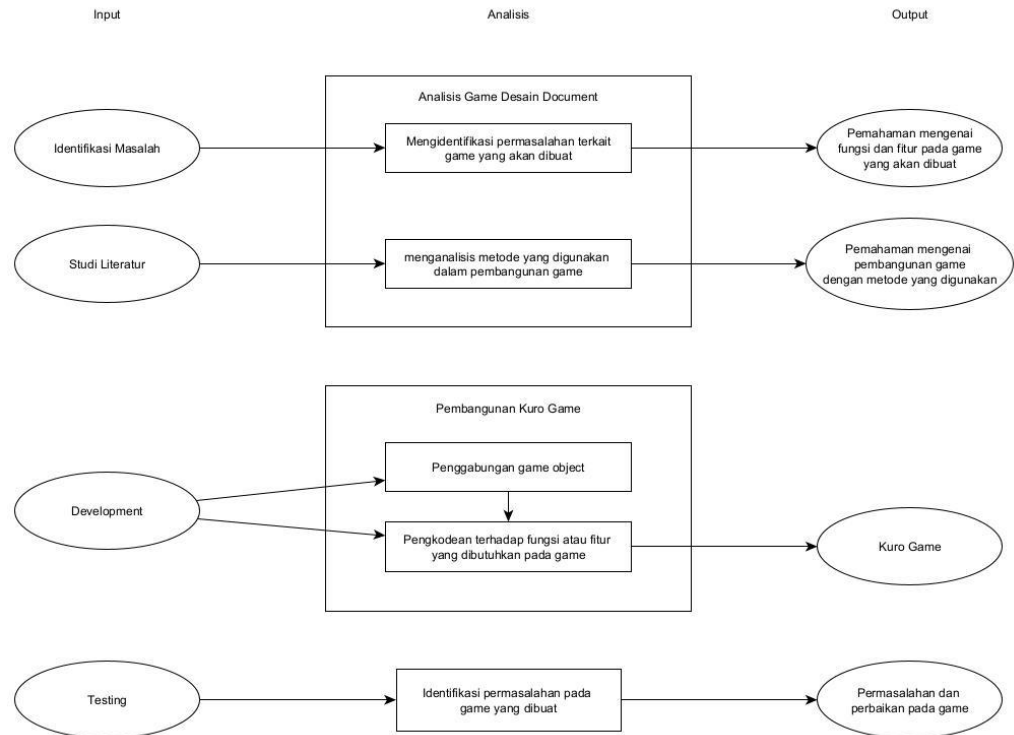
No	Komponen Utama	Detail Komponen	Penjelasan
1	Method	Proses dilanjutkan pada tahap <i>development</i> dan pengujian	Melanjutkan proses pada DGBL yang hanya pada tahap desain, dilanjutkan pada <i>development</i> dan pengujian
2	Media	Menambahkan implementasi script dan fitur pemain	Menambahkan script pada asset yang sudah tersedia dan mengatur fitur permainan agar sesuai dengan alur pada <i>Game Design Document</i>
3	Teknologi	Pembuatan script pada <i>game</i>	Membuat script untuk mengatur logika permainan, mengaktifkan fungsi dan mekanika permainan, dan menghubungkan asset permainan.

3.3.2 Skema Analisis Teori

Skema analisis menjelaskan mengenai analisis yang dilakukan selama penelitian tugas akhir. Skema analisis ini memperlihatkan *input* yang diperlukan untuk kebutuhan analisis dan *output* dari hasil analisis yang dilakukan. Terdapat tiga komponen utama yang terdapat pada skema analisis yaitu sebagai berikut :

1. Input, merupakan komponen yang menjelaskan mengenai masukan yang digunakan sebagai acuan atau referensi untuk tahap analisis. Input bisa berupa data dan informasi yang dapat dijadikan dasar analisis.
2. Analisis, merupakan komponen yang menjelaskan mengenai tahapan analisis yang dilakukan dalam penelitian tugas akhir. Tujuannya untuk mengetahui keadaan sebenarnya dalam pengerjaan tugas akhir terkait dengan objek penelitian.
3. Output, merupakan komponen yang menjelaskan mengenai keluaran yang dihasilkan dari tahap analisis yang telah dilakukan.

Berikut merupakan gambar skema analisis penelitian tugas akhir yang dilakukan



Gambar 3.3 Skema Analisis Teori

Setelah digambarkan rencana analisis, kemudian akan dijelaskan tahapan yang dilakukan dan maksud dari setiap tahapan yang dilakukan. Pada tabel 3.4 merupakan tahapan rencana analisis.

Tabel 3.4 Rencana Analisis

No	Input	Langkah Analisis	Objek Analisis	Output	Penjelasan
1	Analisis terkait kebutuhan game yang akan di buat pada <i>Game Design Document</i> (GDD)	Mengidentifikasi permasalahan pembangunan pada game yang akan dibuat	Desain Game	Pemahaman mengenai fungsi dan fitur pada game yang akan dibuat	Tahap ini untuk mengetahui dan menjelaskan mengenai identifikasi masalah, literatur-
		- Menganalisis metode yang akan digunakan - Menganalisis konsep game serta alur dari game yang akan dibuat		Pemahaman mengenai pembangunan game menggunakan bahasa pemrograman C# dan Unity	literatur yang digunakan serta proses pembangunan game menggunakan bahasa pemrograman C# dan Unity sebagai <i>game engine</i> nya

No	Input	Langkah Analisis	Objek Analisis	Output	Penjelasan
				sebagai <i>Game Engine</i> nya	
2	• Development	• Penggabungan <i>game object</i> • Pengkodean terhadap fungsi yang dibutuhkan pada game	Desain Game	• Kuro Game	Pada tahap ini dilakukan pembangunan game dari awal game dimulai sampai game selesai dimainkan, termasuk menjalankan semua fitur dan fungsinya
3	• Testing	• Identifikasi permasalahan pada game yang dibuat	Kuro Game	• Permasalahan dan perbaikan pada game	Pada tahap ini dilakukan pengujian terhadap game yang sudah dibangun, yang bertujuan untuk mengetahui segala permasalahan yang ada pada game, serta dilakukan perbaikan terhadap permasalahan yang telah ditemukan

3.4 Profil Penelitian

Objek penelitian yang diteliti oleh penulis fokus pada pembangunan game 3D dengan menggunakan bahasa pemrograman C# untuk sebuah game mobile. Untuk mencapai tujuan ini, penulis melakukan penggunaan dua aplikasi utama, yaitu Microsoft Visual Studio dan Unity 3D. Microsoft Visual Studio digunakan sebagai pembuatan *script* atau pengkodean terhadap fitur dari keseluruhan game, dari game dimulai hingga selesai dimainkan, dengan bahasa C# sebagai bahasa pemrogramannya.

Setelah pengkodean dilakukan, langkah berikutnya adalah melakukan pengujian terhadap *script* tersebut untuk mengetahui bahwa fungsi yang dibuat sudah berhasil dengan menggunakan Unity 3D, dan memastikan bahwa tidak ada *error* yang dapat menyebabkan game tidak dapat dimainkan. Unity 3D berperan sebagai platform penggabungan semua objek yang telah dibuat, seperti *script* atau pengkodean yang telah dibuat ditambahkan pada *game object* yang telah disediakan. Dalam tahap ini, penulis

membuat *script* atau pengkodean terhadap fitur yang telah dibuat sebelumnya menggunakan Microsoft Visual Studio dan menggunakan Unity 3D untuk mengimplementasikan *script* tersebut pada fitur atau fungsi yang akan dibuat. Selain itu, Unity 3D juga memberikan fasilitas untuk mengimplementasikan fitur-fitur *game*, mengatur logika *game*, dan menguji kinerja terhadap pengkodean yang telah dibuat. Dengan demikian, kolaborasi antara Microsoft Visual Studio dan Unity 3D menjadi fondasi utama dalam pembangunan *game* 3D untuk mobile ini, memungkinkan penulis untuk menciptakan *game* 3D interaktif yang siap dieksplorasi oleh pengguna.

3.5 Analisis

Pada tahap ini ditetapkan konsep dari *script* atau pengkodean *game* yang akan dibuat, yaitu menetapkan identifikasi pengguna dan identifikasi kebutuhan

3.5.1. Pengguna Script

Dalam proses analisis pengguna yang dilakukan, telah diidentifikasi kelompok target pengguna yang memanfaatkan pengkodean dari *game* yang dibuat dalam rangka pembangunan *game*. Berdasarkan identifikasi ini, terlihat dengan jelas bahwa pengkodean pada *game* 3D ini akan memiliki tiga kelompok pengguna utama :

Pertama-tama, pengkodean ini akan menjadi alat yang sangat relevan bagi *game* desain. Mereka akan memanfaatkan *script* ini untuk menciptakan, mengintegrasikan, dan mengatur berbagai objek sebuah fungsi atau fitur yang akan menjadi bagian keseluruhan dari *game* yang sedang dibangun. Pengkodean ini juga akan menjadi program dari *game* yang interaktif dan dapat menjalankan fungsi atau tugas tertentu sesuai dengan visi dan rencana pengembangan yang telah ditetapkan.

Selain itu, *Game Player* juga akan menjadi kelompok pengguna yang penting dalam pengkodean pada *game* 3D ini. Mereka akan menggunakan pengkodean pada *game* ini untuk memainkan dan menikmati permainan yang telah dikembangkan. Mereka menjadi target utama dari pengkodean terhadap *game* yang telah dibuat, dan pembangunan *game* ini dilakukan agar pemain dapat menikmati pengalaman yang memuaskan.

3.5.2. Kebutuhan Script

Analisis kebutuhan yang dilakukan untuk mengetahui *script* pada *game* 3D yang digunakan untuk pembangunan *game* 3D pada *game* mobile itu didapatkan dari hasil identifikasi pengguna, yang memberikan kebutuhan *script* atau pengkodean yang akan digunakan khususnya dari *game* desain. Dari identifikasi yang dilakukan *game* desain menginginkan pengkodean sebagai berikut :

1. Technology

Konsep element technology bertujuan untuk media dimana Aesthetic akan berlangsung, dimana Mechanic akan terjadi, dan bagaimana Story akan disampaikan

pada game tersebut. Unsur Technology yang digunakan dalam game edukasi ini yaitu teknologi sistem operasi Android sebagai platform game edukasi ini, OS Android dipilih agar dapat dengan mudah diakses di manapun dan kapanpun selama siswa membawa Smartphone nya.

2. Mechanic

Pendekatan Mechanic pada game yang akan dibangun berdasarkan core action yaitu berjalan dan berlari, mundur, belok ke kiri, belok ke kanan, dan loncat. Berdasarkan hal tersebut maka terdapat kemampuan pemain untuk bergerak ke kiri dan kanan, dapat melompat untuk menghindari rintangan dan menaiki tangga, pemain dapat mengambil barang, pemain dapat berinteraksi dengan guru dan satpam.

3. Control

Berdasarkan Mechanic, maka dibuatlah kontrol pemain dengan menggunakan UI berupa virtual button untuk menyesuaikan kebutuhan berdasarkan pada Technology, yaitu Android. Maka dengan UI tersebut game dapat dikendalikan dengan cara menyentuh layar sesuai dengan UI yang sudah tersedia pada game, serta pemain dapat melakukan aksi pada game sesuai dengan yang sudah ditentukan.

BAB IV ANALISIS DAN PEMBANGUNAN GAME

Bab ini akan membahas analisis dan Pembangunan *game* edukasi Bahasa Inggris dengan menggunakan metode *Digital Game Based Learning* (DGBL). Metode DGBL merupakan metode pembelajaran yang menggunakan teknologi *game* (permainan) ke dalam lingkungan belajar. Dalam bab ini berisi Langkah-langkah yang dilakukan dalam analisis kebutuhan *game* yang ada pada *Game Design Documen* (GDD) dan Pembangunan *game* mengacu pada *Game Design Document* dengan menggunakan metode DGBL.

4.1 Analisis Struktur Pemrograman

Tahap analisis struktur pemrograman bertujuan untuk mengetahui apa yang dibutuhkan dalam pembangunan *game* edukasi Kuro Game. Kebutuhan pemrograman meliputi identifikasi komponen-komponen utama yang akan digunakan dalam *game*, alur kerja masing-masing sistem, serta bagaimana tiap bagian saling berinteraksi. Berikut adalah struktur komponen utama dalam sistem pemrograman Kuro Game, beserta tanggung jawab masing-masing :

Tabel 4.1 Struktur Pemrograman

No	Komponen	Tanggung Jawab Utama
1	Menu Utama	Menampilkan tampilan awal <i>game</i> seperti tombol mulai / play.
2	NPCDialogueController	Mengatur interaksi dialog antara pemain dengan karakter NPC seperti guru dan satpam.
3	OnExitScene	Menangani transisi perpindahan antar scene.
4	GameManager_Main	Mengelola alur utama <i>gameplay</i> dalam scene, termasuk akses area dan pengaturan karakter berdasarkan progress
5	GameplayData	Menyimpan data progress pemain secara global agar tetap tersedia meskipun terjadi perpindahan scene
6	GameManager_Materi	Mangatur urutan dan alur penyelesaian materi pembelajaran dalam satu ruangan
7	MateriController	Menampilkan konten materi saat pemain berinteraksi dengan objek tertentu di dalam scene.
8	InteractableObjectController	Mengatur objek-objek interaktif dalam scene yang bisa disentuh atau diklik oleh pemain.
9	GameManager_Quiz	Mengatur keseluruhan alur kuis, mulai dari soal, jawaban, waktu pengerjaan, <i>score</i> kuis, dan kesempatan menjawab kuis.
10	QuizInitiator	Sebagai penghubung antara objek interaktif di scene dengan sistem kuis yang akan dijalankan.

4.2 Perancangan Interaksi dan Kecerdasan NPC

Interaksi pemain dan NPC merupakan elemen penting dalam *game* edukasi. NPC dapat memberikan informasi terkait hal apa yang perlu dilakukan oleh pemain. Maka, sistem ini dirancang agar fleksibel dan responsive terhadap aksi pemain.

Skema Interaksi Dasar NPC, sebagai berikut :

- Setiap NPC memiliki area interaksi berbasis trigger collider. Ketika pemain memasuki area tersebut, sistem mengenali keberadaan pemain sebagai objek untuk memulai interaksi.
- Interaksi dengan NPC dilakukan secara otomatis. Saat pemain mendekat, sistem langsung menampilkan dialog atau pop-up informasi menggunakan NPCDialogueController.
- Dialog yang muncul dapat berupa arahan atau instruksi yang harus dilakukan oleh pemain.
- Setelah dialog selesai, sistem akan melanjutkan aksi tertentu, seperti membuka akses ruang atau *scene* baru, memicu kuis, atau menyimpan progress melalui GameplayData.

4.3 Perancangan Interaksi Objek Interaktif

Skema Interaksi Game Objek, sebagai berikut :

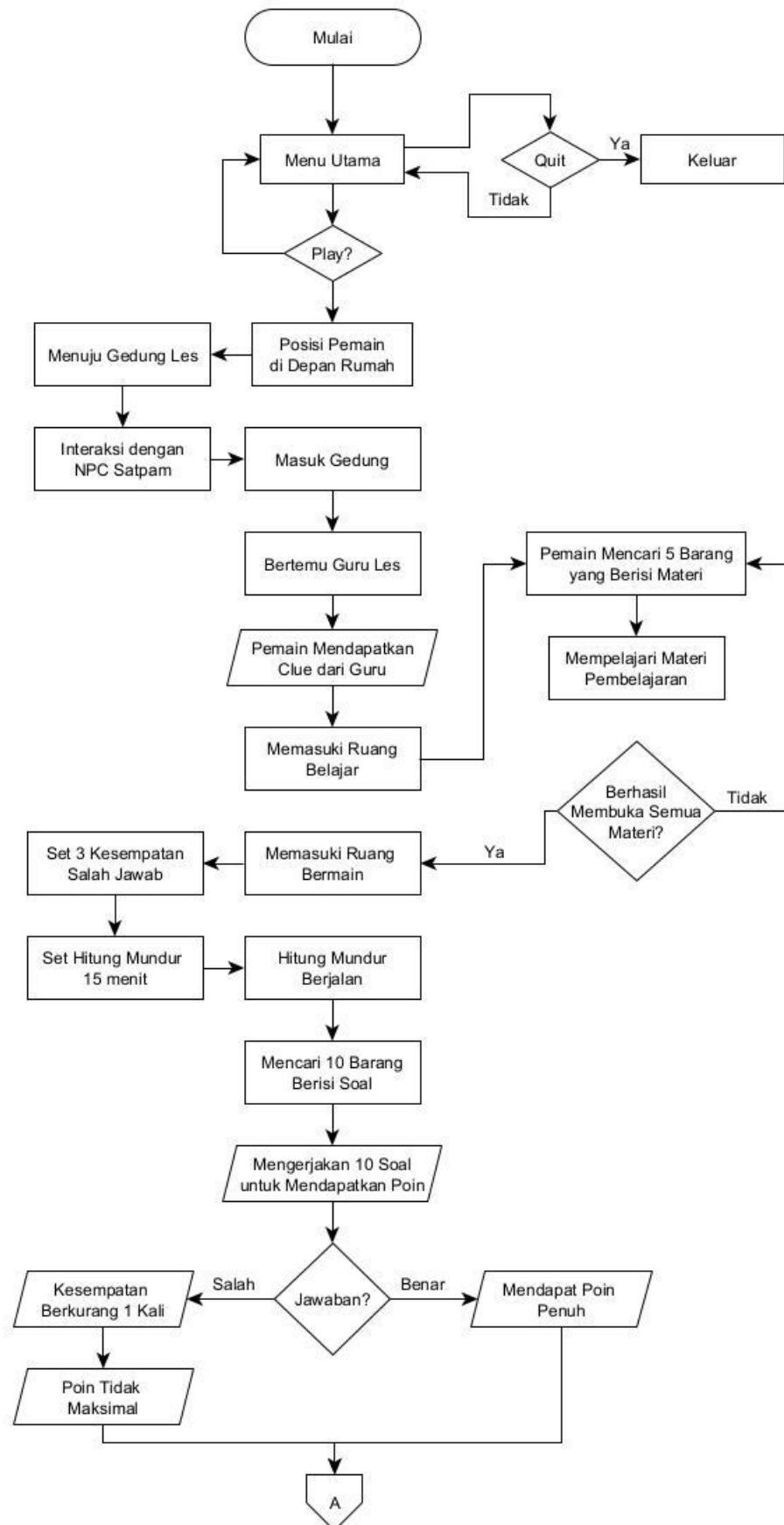
Berbeda dengan interaksi NPC yang berbasis area trigger, interaksi dengan objek interaktif dalam Kuro Game menggunakan pendekatan berbasis deteksi visual melalui crosshair. Sistem ini dirancang agar pemain dapat secara aktif memilih objek yang ingin diakses menggunakan arah pandang kamera.

- Setiap objek interaksi memiliki komponen Collider yang dapat terdeteksi oleh sistem saat berada di tengah layar (crosshair).
- Ketika pemain mengarahkan *crosshair* ke objek tertentu, sistem akan mengenali objek tersebut sebagai interaktif, lalu menampilkan tombol aksi di layar seperti menampilkan tombol.
- Setelah tombol muncul, pemain dapat menekan tombol tersebut untuk memulai interaksi.
- Sistem kemudian memanggil fungsi yang sesuai, seperti :
 1. MateriController untuk menampilkan konten pembelajaran.
 2. QuizInitiator untuk memulai sesi soal kuis melalui GameManager_Quiz.
- Status atau hasil dari interaksi tersebut kemudian dicatat oleh GameplayData sebagai bagian dari progres pemain.

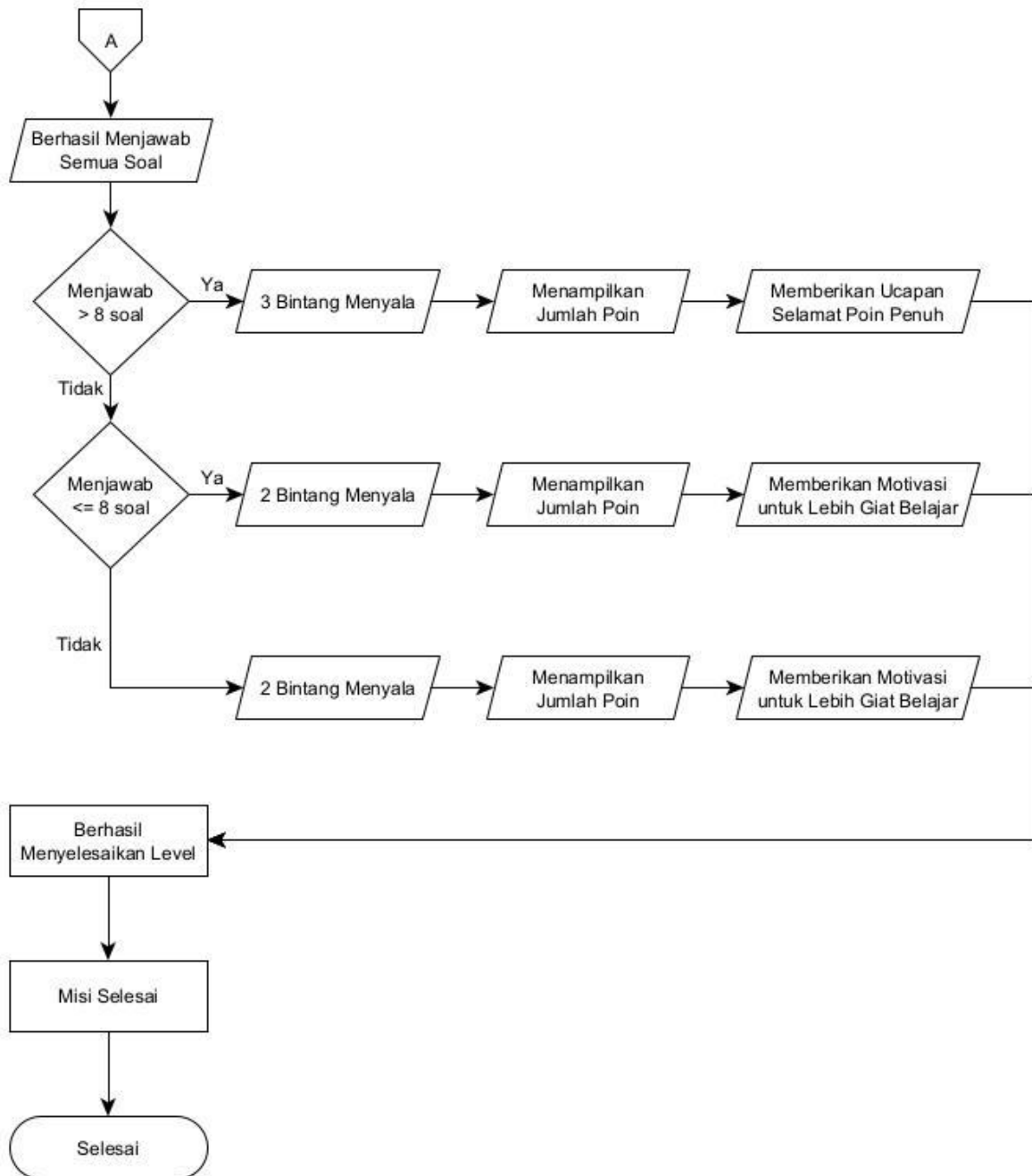
Pendekatan ini memberikan pengalaman eksploratif yang lebih dinamis dan memberi pemain kontrol penuh terhadap objek yang ingin diakses. Skema ini juga cocok untuk platform mobile dengan gaya permainan *first-person* yang mengandalkan arah pandang kamera sebagai alat navigasi.

4.4 Flowchart

Pada Gambar 4.1 dan 4.2, flowchart tersebut menggambarkan alur permainan Kuro Game, dimulai dari tampilan Main Menu hingga game selesai dimainkan, yang ditandai dengan hasil evaluasi pemain setelah menjawab soal. Setiap Langkah dalam alur ini menunjukkan proses yang harus dilalui pemain, termasuk navigasi menu, interaksi dengan karakter, pencarian materi, dan tahap ujian pengetahuan. Dengan demikian, flowchart ini memberikan panduan yang jelas mengenai perjalanan permainan dalam game, serta hasil akhir yang mencerminkan pencapaian pemain.



Gambar 4.1 Flowchart Alur Game Kuro Game (Bagian 1)



Gambar 4.2 Flowchart Alur Game Kuro Game (Bagian 2)

4.5 Pembangunan Game

Pada bagian ini, akan dijelaskan tentang implementasi teknis dari *game* berdasarkan analisis yang telah dilakukan pada GDD. Penulis akan membahas Langkah-langkah Pembangunan dan bagaimana kode program diterapkan untuk memenuhi kebutuhan yang telah dianalisis. Bagian ini akan mencakup penjelasan tentang berbagai sistem yang telah dibangun, serta contoh kode program yang digunakan untuk mengimplementasikan setiap fitur dalam *game*.

4.5.1 Menu Utama

Pada tahap ini, penulis mengimplementasikan menu utama yang terdiri dari dua tombol utama, yaitu play untuk memulai permainan dan quit untuk keluar dari aplikasi. Menu utama ini dibangun menggunakan Unity UI dan diprogram dengan C#.

Berikut adalah kode program untuk menangani aksi yang terjadi saat tombol-tombol tersebut diklik oleh pemain :

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;
public class MainMenu : MonoBehaviour
{
    private void Start()
    {
    }
    public void OnPlayButton() {
        SceneManager.LoadScene(1);
    }
    public void OnQuitButton() {
        Application.Quit();
    }
}
```

Penjelasan Kode :

1. Using UnityEngine.SceneManagement; : Ini adalah namespace yang diperlukan untuk menangani manajemen scene dalam Unity. Dalam kode ini, kita menggunakan SceneManager untuk memuat scene tertentu berdasarkan ID.
2. SceneManager.LoadScene(1) : Fungsi ini digunakan untuk memuat scene permainan dengan ID 1. Scene dengan ID 1 biasanya adalah scene permainan utama yang akan dilakukan Ketika tombol play ditekan.
3. Application.Quit(): Fungsi ini digunakan untuk menutup aplikasi saat tombol Quit ditekan. Fungsi ini bekerja saat aplikasi dijalankan Ketika aplikasi sudah dibangun, dan fungsi ini tidak akan berfungsi dalam Editor Unity.
4. OnPlayButton() dan OnQuitButton() adalah fungsi yang dipanggil Ketika tombol Play dan Quit diklik. Kedua fungsi ini terhubung dengan UI di Unity melalui Event System, di mana setiap tombol dihubungkan dengan fungsi di dalam editor.



Gambar 4.3 Menu Utama

Pada gambar 4.3, ditampilkan menu utama dari game Kuro Game, yang terdiri dari tombol Play, Setting, Help, dan Quit. Tombol play digunakan untuk memulai permainan dan akan memuat scene permainan dengan ID 1 ketika ditekan. Tombol setting memungkinkan pemain untuk mengakses pengaturan. Tombol help memungkinkan pemain untuk mengetahui cara bermain dari game Kuro Game. Dan tombol Quit digunakan untuk menutup aplikasi game.

Tampilan ini diimplementasikan menggunakan Unity UI dan dihubungkan dengan kode melalui Event System untuk menanggapi klik tombol. Dengan mengklik tombol Play, pemain akan diarahkan ke scene permainan, sedangkan klik tombol Quit akan menutup aplikasi game.

4.5.2 Interaksi NPC Satpam

Pada tahap ini, penulis mengimplementasikan pemain yang dapat berinteraksi dengan satpam, yaitu :

Berikut adalah kode program untuk menangani interaksi antara pemain dengan satpam :

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.Events;

public class NPCDialogueController : MonoBehaviour
{
    [System.Serializable]
```

```

public class Dialogue
{
    public string name;
    public string dialogue;
}

private void OnTriggerEnter(Collider other)
{
    Debug.Log($"{other.gameObject} is entering the area");
    if (other.CompareTag("Player"))
    {
        OnEnterArea.Invoke();
    }
}

public UnityEvent OnEnterArea;
public UnityEvent OnDialogueDone;

public List<Dialogue> dialogueList;
public bool DialogueDone;
public bool AllowRepeat;
private void Start()
{
    OnEnterArea.AddListener(StartDialogue);
}

public void StartDialogue()
{
    if (DialogueDone && AllowRepeat == false)
        return;

    FindObjectOfType<DialogueCanvas>().StartDialogue(dialogueList, this);
    GetComponent<BoxCollider>().enabled = false;
    DialogueDone = true;
}
}

```

Penjelasan kode :

1. Class Dialogue

Merupakan class sederhana untuk menyimpan data dialog berupa nama karakter dengan nama variabel name dan teks percakapan dengan nama variabel dialogue.

2. Fungsi OnTriggerEnter(Collider other)

Fungsi ini mendeteksi Ketika objek lain memasuki area yang dimana terdapat collider. Jika objek tersebut memiliki tag “Player”, maka event OnEnterArea akan dijalankan.

3. Event OnEnterArea dan OnDialogueDone

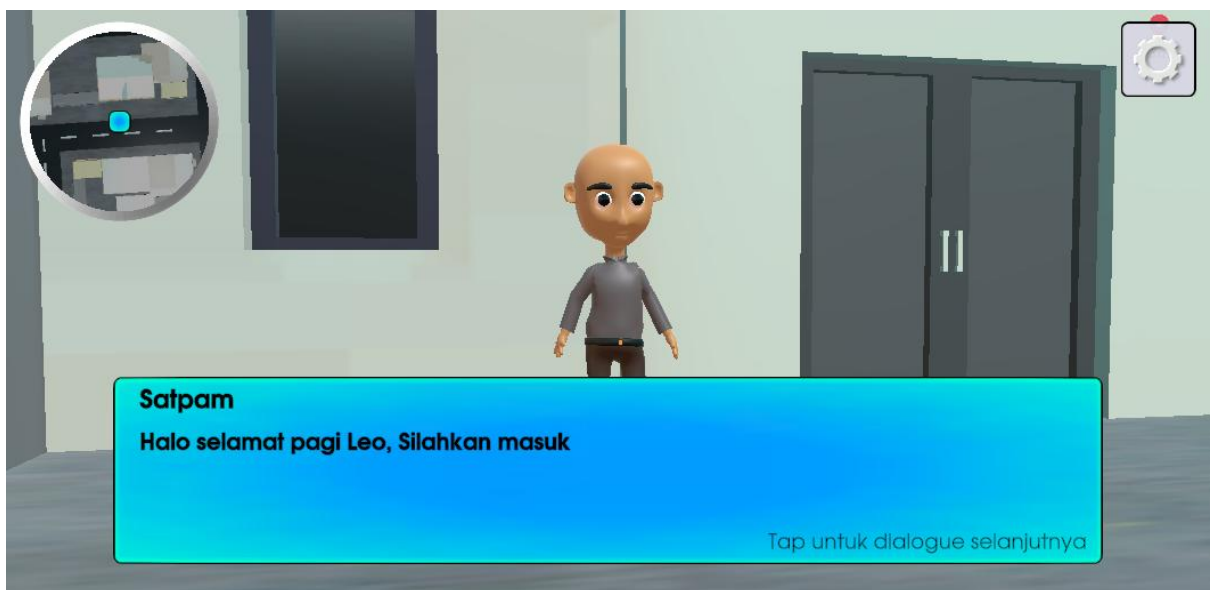
Menggunakan UnityEvent untuk memicu fungsi tertentu. OnEnterArea akan dipanggil saat pemain memasuki area NPC, sedangkan OnDialogueDone dapat digunakan untuk aksi lanjutan setelah dialogue selesai.

4. Fungsi Start()

Digunakan untuk menambahkan *listener* ke *event* OnEnterArea, yaitu agar saat *event* tersebut dipanggil, fungsi StartDialogue() akan dijalankan.

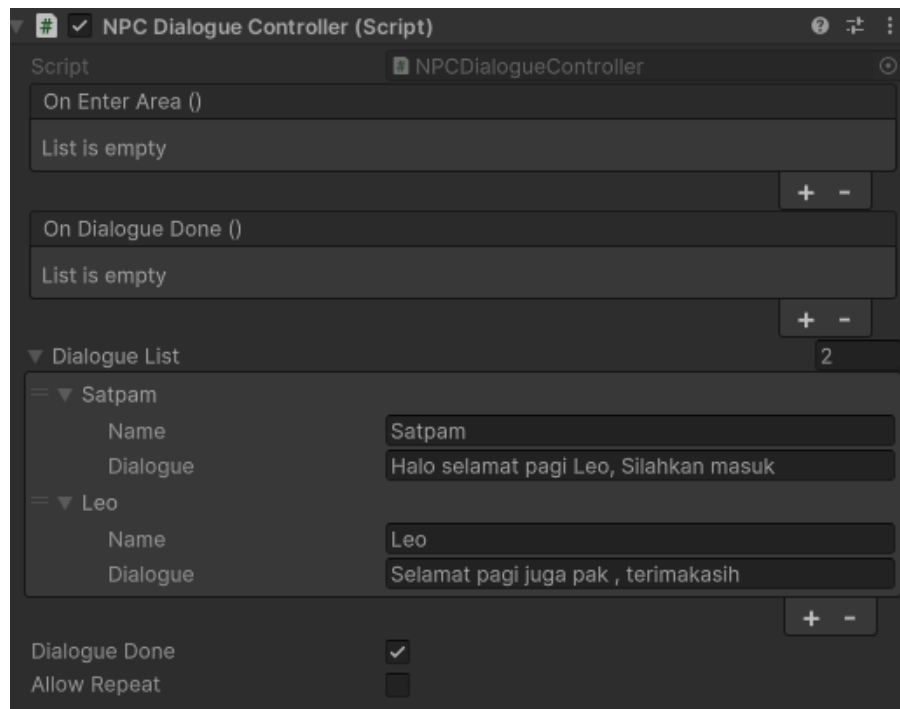
5. Fungsi StartDialogue()

StartDialogue() berfungsi untuk memulai dialog. Terdapat pengecekan agar dialog tidak diulang jika tidak diizinkan fungsi AllowRepeat. Setelah dialog dimulai, collider akan dimatikan agar tidak bisa memicu dialog ulang, dan status DialogueDone akan diubah menjadi true.



Gambar 4.4 Interaksi NPC Satpam

Pada Gambar 4.4, ditampilkan hasil dari interaksi antara pemain dengan NPC satpam, dengan menampilkan nama NPC, dan dialog yang sedang berlangsung.



Gambar 4.5 NPC Dialogue Controller

Pada gambar 4.5, adalah fungsi ketika *script* NPCDialogueController ditambahkan pada objek NPC satpam. Dengan menambahkan dua dialogue list yang berfungsi untuk menambahkan nama NPC dan dialogue yang sedang berlangsung.

4.5.3 Perpindahan Scene

Pada Tahap ini, penulis mengimplementasikan fitur perpindahan ruangan yang memungkinkan pemain memasuki area baru atau pemain dapat berpindah scene sesuai dengan alur *game* nya.

Berikut adalah kode program untuk menangani aksi yang terjadi ketika pemain menyentuh area tertentu, yaitu dengan collider :

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class OnExitScene : MonoBehaviour
{
    public string SceneName;
    public bool isNextScene = true;

    [SerializeField] public SceneInfo sceneInfo;

    void OnTriggerEnter(Collider other)
    {
        sceneInfo.isNextScene = isNextScene;
    }
}
```

```

        SceneManager.LoadScene (SceneName) ;
    }
}

```

Script ini digunakan untuk mengatur perpindahan antar scene ketika pemain memasuki area tertentu. Perpindahan ini terjadi setelah pemain menyelesaikan interaksi tertentu atau mencapai titik tertentu di dalam permainan. Berikut adalah penjelasan pada script :

1. Public String SceneName

Fungsi ini dapat menyimpan nama scene yang akan dimuat saat pemain memasuki area. Nama scene ini diatur langsung melalui *inspector Unity*, sehingga bisa disesuaikan tanpa mengubah kode program.

2. Public bool isNextScene = true

Menandai apakah scene yang akan dimuat adalah scene lanjutan (next scene) atau bukan. Nilai ini juga bisa diatur melalui *inspector*.

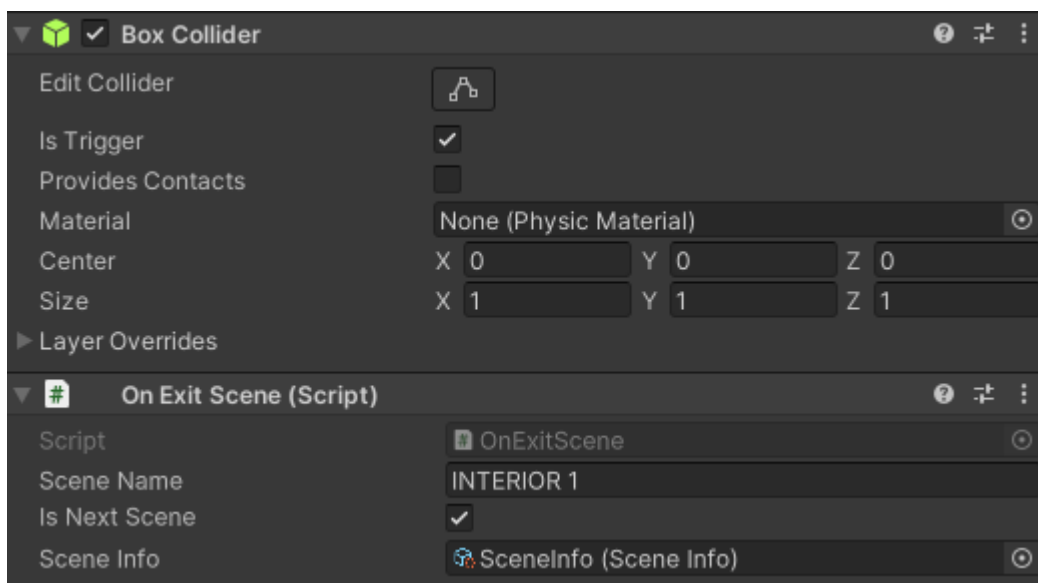
3. [SerializeField] public SceneInfo sceneInfo

Variabel ini merupakan referensi ke sebuah *ScriptableObject* bernama SceneInfo, yang menyimpan data secara global terkait status perpindahan antar scene. Dengan menggunakan sceneInfo, script dapat meng-*update* informasi isNextScene secara global sehingga dapat diakses oleh *script* lain di *scene* selanjutnya.

4. Void OnTriggerEnter(Collider other)

Fungsi ini dijalankan secara otomatis oleh Unity ketika suatu objek dengan collider memasuki trigger collider pada objek ini. Di dalam fungsi :

- a. Nilai isNextScene disimpan ke sceneInfo.isNextScene untuk menandai status scene berikutnya
- b. Scene tujuan dimuat menggunakan perintah SceneManager.LoadScene(SceneName);



Gambar 4.6 *Inspector* Perpindahan Scene

Pada Gambar 4.6, adalah tampilan *inspector* pada *GameObject* yang dapat men-trigger perpindahan scene. Pada gambar terlihat komponen *Box Collider* yang diaktifkan sebagai trigger, sehingga saat pemain memasuki areal tersebut, event perpindahan scene dapat dipicu. Selain itu, terdapat komponen script *OnExitScene* yang berisi script pengatur perpindahan scene, termasuk variabel untuk menentukan nama scene yang berupa scene yang dituju (*Scene Name*), status apakah scene tersebut lanjutan (*Is Next Scene*), serta referensi ke *SceneInfo* sebagai penyimpanan data global terkait perpindahan scene.

Pengaturan ini memungkinkan perpindahan scene terjadi secara otomatis ketika pemain memasuki ke area trigger.

4.5.4 Game Manager

Pada bagian ini, script *GameManager_Main* bertindak sebagai pengatur utama pada game dalam scene, di mana script ini mengelola status tampilan karakter dan akses area berdasarkan progress pemain. Script ini dijalankan saat scene dimulai (*Start()*), dengan memanfaatkan data dari *GameplayData*, yaitu class global yang menyimpan informasi progress pemain lintas scene.

Penjelasan kode :

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class GameManager_Main : MonoBehaviour
{
    public Collider ruangMateri1Entrance;
    public Collider ruangKuis1Entrance;

    public GameObject pakGuruBeforeMateri1;
    public GameObject pakGuruAfterMateri1;
    public GameObject pakGuruAfterKuis1;
}
```

1. RuangMateri1Entrance dan ruangKuis1Entrance (Collider)

Digunakan sebagai trigger untuk masuk ke ruangan materi dan kuis. Collider ini akan diaktifkan atau dinonaktifkan tergantung dari progress permainan.

2. pakGuruBeforeMateri1, pakGuruAfterMateri1, dan pakGuruAfterKuis1 (GameObject)

GameObject ini adalah representasi karakter (NPC Guru) dalam kondisi yang berbeda :

- a. pakGuruBeforeMateri1, muncul sebelum pemain menyelesaikan materi.

- b. `pakGuruAfterMateri1`, muncul setelah materi selesai, dan sebelum kuis dimulai.
- c. `pakGuruAfterKuis1`, muncul setelah pemain menyelesaikan kuis.

3. Fungsi `Start()`

- a. Fungsi NPC guru sebelum materi

```
pakGuruBeforeMateri1.gameObject.SetActive(GameplayData.instance.SessionProgress.ruangMateri1Done == false);
```

Fungsi ini dapat menampilkan NPC guru sebelum materi diselesaikan.

- b. Fungsi NPC guru sesudah materi

```
pakGuruAfterMateri1.gameObject.SetActive(
GameplayData.instance.SessionProgress.ruangMateri1Done == true
&& GameplayData.instance.SessionProgress.ruangQuiz1Done == false);
```

Fungsi ini dapat menampilkan NPC guru setelah materi diselesaikan, dan kuis belum diselesaikan.

- c. Fungsi NPC guru setelah materi dan kuis selesai

```
pakGuruAfterKuis1.gameObject.SetActive(
GameplayData.instance.SessionProgress.ruangMateri1Done == true
&& GameplayData.instance.SessionProgress.ruangQuiz1Done == true);
```

Fungsi ini dapat berjalan ketika pemain telah menyelesaikan materi dan kuis.

- d. Fungsi untuk akses ke ruangan materi

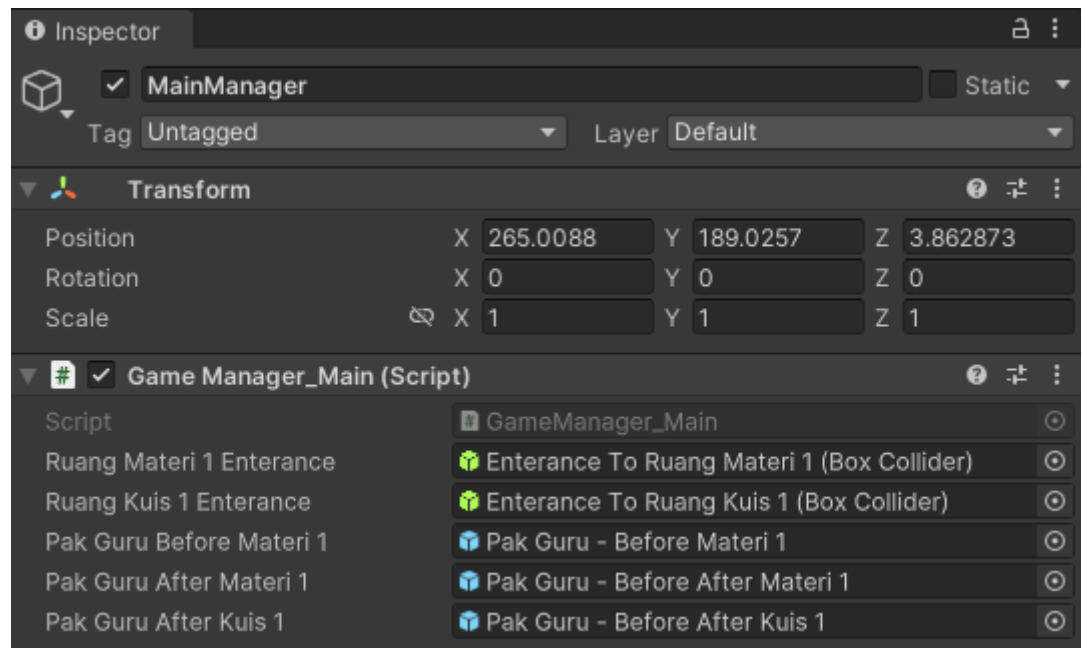
```
ruangMateri1Entrance.enabled =
GameplayData.instance.SessionProgress.ruangMateri1Done == false;
```

Fungsi ini memungkinkan pemain dapat mengakses ke ruangan materi, fungsi ini dapat berjalan ketika pemain belum menyelesaikannya.

- e. Fungsi untuk akses ke ruangan kuis

```
ruangKuis1Entrance.enabled =
(GameplayData.instance.SessionProgress.ruangMateri1Done ==
true &&
GameplayData.instance.SessionProgress.ruangQuiz1Done ==
false);
```

Fungsi ini memungkinkan pemain untuk mengakses ke ruangan kuis, fungsi ini dapat berjalan ketika pemain telah menyelesaikan materi, dan belum menyelesaikan kuis.



Gambar 4.7 MainManager

Pada gambar 4.7 menunjukkan inspector dari GameObject MainManager yang memiliki komponen script GameManager_Main. Di dalam inspector terdapat beberapa referensi GameObject dan Collider yang telah di-assign, antara lain :

4. Entrance To Ruang Materi 1 dan Entrance To Ruang Kuis 1
Collider yang berfungsi sebagai *trigger* masuk ke ruangan materi dan kuis.
5. Pak Guru – Before Materi 1, Pak Guru – Before After Materi 1, dan Pak Guru- Before After Kuis 1
Masing-masing mewakili kondisi NPC guru yang akan ditampilkan berdasarkan progress pemain.

4.5.5 GameplayData

Script GameplayData berfungsi sebagai penyimpan data progres pemain secara global dan memungkinkan data tersebut tetap tersedia meskipun scene berpindah. Pada script ini terdapat DontDestroyOnLoad agar objeknya tidak terhapus saat berpindah scene.

Berikut adalah kode program untuk GameplayData yang dapat menangani progres pada game :

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class GameplayData : MonoBehaviour
```

```

{

[System.Serializable]
public class Progress
{
    public bool ruangMateri1Done;
    public bool ruangQuiz1Done;
    public bool ruangMateri2Done;
    public bool ruangQuiz2Done;

    public float quiz1Score;
    public float quiz2Score;
}

public Progress SessionProgress;
public static GameplayData instance;

private void Awake()
{
    if(instance != null)
    {
        Destroy(gameObject);
        return;
    }
    instance = this;
    DontDestroyOnLoad(gameObject);
}
}

```

Penjelasan kode :

1. Class Progress

Class ini dibuat sebagai struktur untuk menyimpan status progres permainan. Beberapa variabel di dalamnya yaitu :

a. ruangMateri1Done dan ruangQuiz1Done

Fungsi ini untuk menyimpan status apakah materi atau kuis di masing-masing ruangan sudah diselesaikan oleh pemain.

b. quiz1Score

Fungsi ini untuk menyimpan score yang didapatkan dari kuis.

2. SessionProgress

Objek dari class Progress, digunakan untuk mengakses dan memodifikasi progres pemain selama sesi permainan.

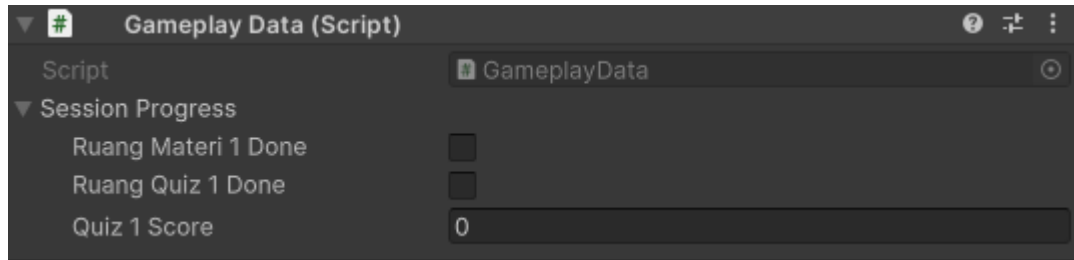
3. GameplayData.instance

Variabel ini menyimpan referensi objek GameplayData yang sedang aktif agar bisa diakses

oleh script lain dengan mudah, misalnya melalui `GameplayData.instance`.

4. Method Awake()

Pada saat objek ini dibuat, script akan mengecek apakah sudah ada objek `GameplayData` lain yang aktif. Jika sudah ada, objek yang baru akan dihapus agar hanya ada satu objek yang menyimpan data progress. Objek yang terseisa ini tidak akan dihapus saat berpindah scene sehingga data progres tetap tersimpan.



Gambar 4.8 *GameplayData*

Pada gambar 4.8, menunjukkan tampilan inspector dari `GameObject` yang memiliki script `GameplayData.cs`. Di dalamnya terdapat objek `SessionProgress` yang berisi beberapa variabel yang menyimpan progres permainan, seperti :

5. Ruang Materi 1 Done, Ruang Quiz 1 Done

Menunjukkan apakah pemain sudah menyelesaikan materi atau kuis pada ruangan materi dan ruangan kuis.

6. Quiz 1 Score

Untuk menyimpan skor kuis yang diperoleh pemain.

4.5.6 GameManager_Materi

Script `GameManager_Materi` bertanggung jawab untuk mengelola alur penyelesaian materi dalam satu ruangan. Script ini akan mencatat jumlah materi yang sudah dibuka oleh pemain, dan hanya akan mengizinkan pemain keluar ruangan jika semua materi sudah dipelajari.

Berikut adalah kode program untuk mengelola materi :

```
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using UnityEngine;

public class GameManager_Materi : MonoBehaviour
{
    // Start is called before the first frame update
    public Collider exitCollider;
    public GameObject clueCanvas_Done;
    public int targetMateri;
```

```

public int currentMateriCount = 0;

public List<MateriController.Materi> materiList;
private void Start()
{
    exitCollider.enabled = false;

    var materiControllers =
FindObjectsOfType<MateriController>().ToList();
    targetMateri = materiList.Count;

    if (materiList.Count > materiControllers.Count)
    {
        Debug.LogError("Jumlah MateriController pada scene lebih
sedikit dari materi yang ada");
        return;
    }

    foreach (var materi in materiList)
    {
        materiControllers[materiList.IndexOf(materi)].materi =
materi;
    }
}

public void SetMateriDone()
{
    currentMateriCount++;
    if (currentMateriCount == targetMateri)
    {
        exitCollider.enabled = true;
        clueCanvas_Done.gameObject.SetActive(true);
        GameplayData.instance.SessionProgress.ruangMateri1Done =
true;
    }
}
}

```

Penjelasan kode :

1. exitCollider

Collider yang mengatur akses keluar dari ruangan. Pada awal permainan, collider ini dinonaktifkan agar pemain belum bisa meninggalkan ruangan sebelum menyelesaikan seluruh materi.

2. clueCanvas_done

GameObject berupa UI yang akan ditampilkan sebagai notifikasi bahwa semua materi telah

selesai dipelajari.

3. `targetMateri`

Menyimpan jumlah total materi yang perlu diselesaikan pemain. Nilainya diambil dari jumlah item dalam `materiList`.

4. `currentMateriCount`

Digunakan untuk mencatat jumlah materi yang sudah diselesaikan oleh pemain.

5. `materiList`

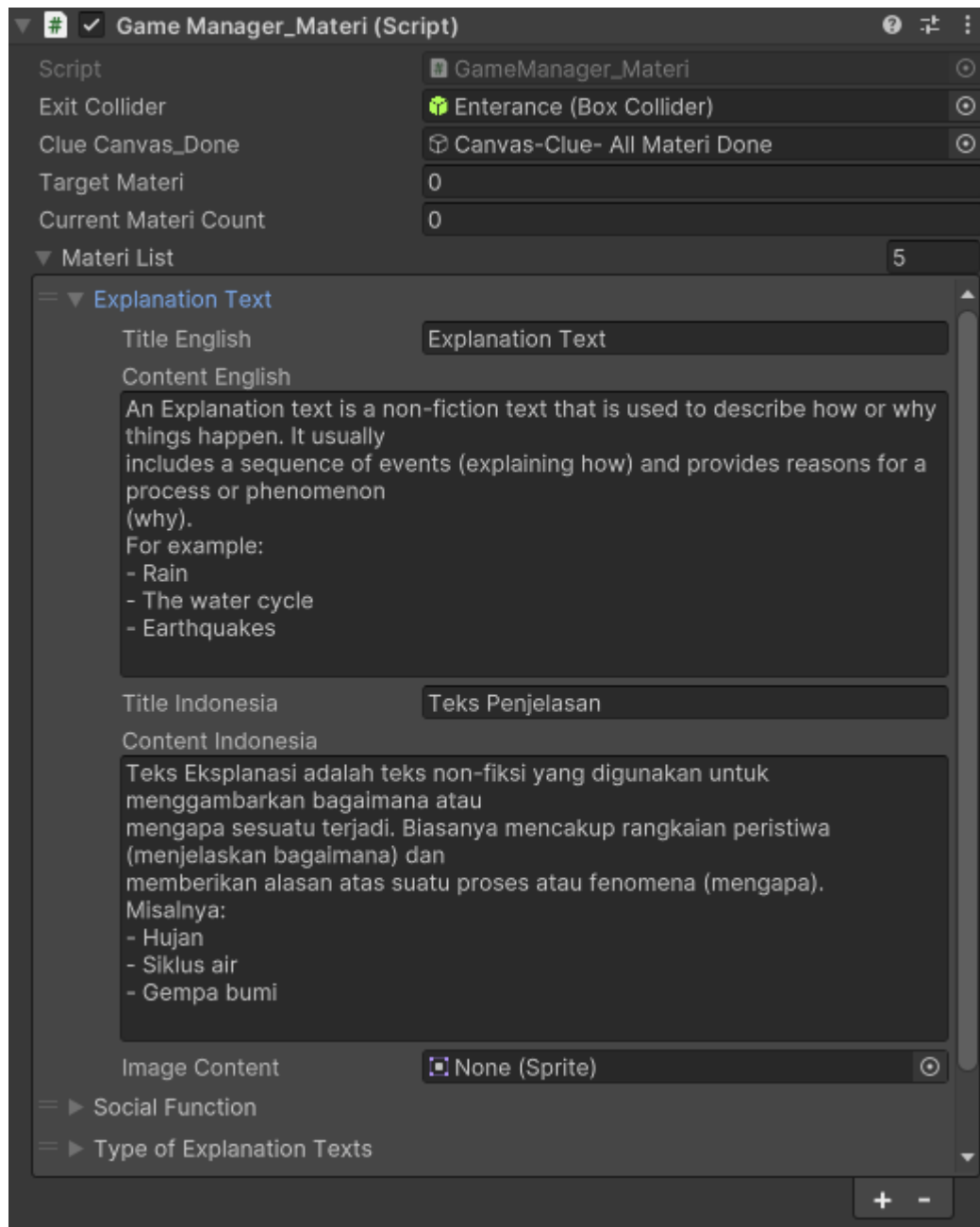
Merupakan daftar data materi yang akan ditampilkan di scene. Masing-masing data akan dipasangkan ke objek `MateriController`.

6. `Start()`

Pada bagian ini, script akan mencari semua objek `MateriController` di scene dan mencocokkan daftar materi (`materiList`) dengan jumlah objek tersebut. Jika jumlah materi lebih banyak dari objek yang ada, akan muncul error. Setelah itu, script memasang data materi ke tiap objek `MateriController` yang sesuai.

7. `SetMateriDone()`

Fungsi ini akan dipanggil ketika pemain menyelesaikan sebuah materi. Fungsi ini menambah hitungan materi yang telah selesai. Jika semua materi sudah selesai (`currentMateriCount == targetMateri`), maka collider untuk keluar ruangan diaktifkan dan petunjuk selesai (`clueCanvas_Done`) ditampilkan. Selain itu, status progress materi pertama diupdate di `GameplayData` agar permainan tahu bahwa materi sudah selesai.

Gambar 4.9 *GameManager Materi*

Pada gambar 4.9, menunjukkan tampilan komponen `GameManager_Materi` yang digunakan untuk mengelola alur penyelesaian materi dalam scene. Beberapa property yang terlihat di inspector antara lain :

8. Exit Collider

Collider yang berfungsi sebagai pemicu untuk keluar ruangan. Collider ini akan diaktifkan secara otomatis apabila seluruh materi telah selesai dipelajari.

9. Clue Canvas Done

Objek UI yang akan muncul sebagai indicator bahwa semua materi telah diselesaikan oleh pemain.

10. Target Materi

Menyimpan jumlah total materi yang harus diselesaikan. Nilai ini diatur secara otomatis berdasarkan isi dari materiList.

11. Current Materi Count

Menyimpan jumlah materi yang telah berhasil diselesaikan oleh pemain. Nilainya akan bertambah melalui script saat pemain membuka materi.

12. Materi List

Daftar data materi yang akan ditampilkan di dalam scene. Data ini dipasangkan ke setiap objek materi agar dapat ditampilkan dan diinteraksikan oleh pemain.

4.5.7 Materi Controller

Script MateriController berfungsi sebagai pengatur tampilan konten materi yang dapat diakses oleh pemain saat melakukan interaksi dengan objek tertentu dalam scene. Script ini merupakan penghubung antara data materi dan antarmuka pengguna (UI) yang menampilkannya.

Berikut adalah kode program MateriController :

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class MateriController : MonoBehaviour
{
    // Start is called before the first frame update
    [System.Serializable]
    public class Materi
    {
        public string titleEnglish;
        [TextArea(10,10)]
        public string contentEnglish;

        public string titleIndonesia;
        [TextArea(10, 10)]
        public string contentIndonesia;

        public Sprite imageContent = null;
    }

    public Materi materi;
    void Start()
    {
        GetComponent<InteractableObjectController>().OnInteracted.AddLi
```

```

stener(OpenMateri);
    }

    public void OpenMateri()
    {
        FindObjectOfType<MaterialCanvas>(true).SetContent(
            materi.titleEnglish,
            materi.titleIndonesia,
            materi.contentEnglish,
            materi.contentIndonesia
        );
    }
}

```

Penjelasan kode :

1. Class Materi

Class ini dibuat sebagai struktur untuk menyimpan informasi dari satu buah materi. Beberapa atribut penting yang dimilikinya antara lain :

a. titleEnglish dan titleIndonesia

Menyimpan judul materi dalam dua Bahasa, yaitu Bahasa Inggris dan Bahasa Indonesia.

b. contentEnglish dan contentIndonesia

menyimpan isi teks materi yang ditulis secara Panjang. Dengan atribut [TextArea(10,10)] digunakan agar konten dapat ditampilkan lebih luas di inspector dan memudahkan pengisian.

c. ImageContent

Gambar pendukung materi, bertipe sprite, bersifat opsional.

2. Materi

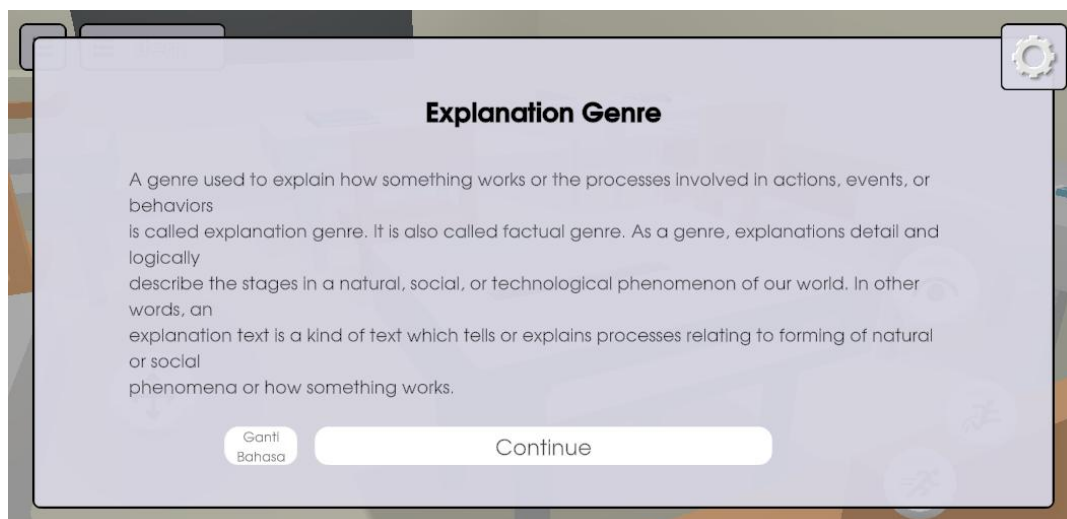
Variabel public bertipe Materi, digunakan untuk memasukkan data materi ke masing-masing GameObject materi di dalam scene.

3. Start()

Pada saat scene dimulai, script akan mencari komponen InteractableObjectController pada GameObject yang sama, kemudian menambahkan fungsi OpenMateri() ke dalam event OnInteracted. Artinya, ketika pemain mengarahkan cursor dan menekan tombol intraksi pada objek materi fungsi OpenMateri() otomatis dipanggil.

4. Openmateri()

Fungsi ini bertugas untuk menampilkan isi dari materi ke antarmuka pengguna. Caranya adalah dengan mencari komponen MaterialCanvas di scene, lalu mengatur isi teks dan judul dari materi berdasarkan data yang sudah dimasukkan ke dalam variabel materi.



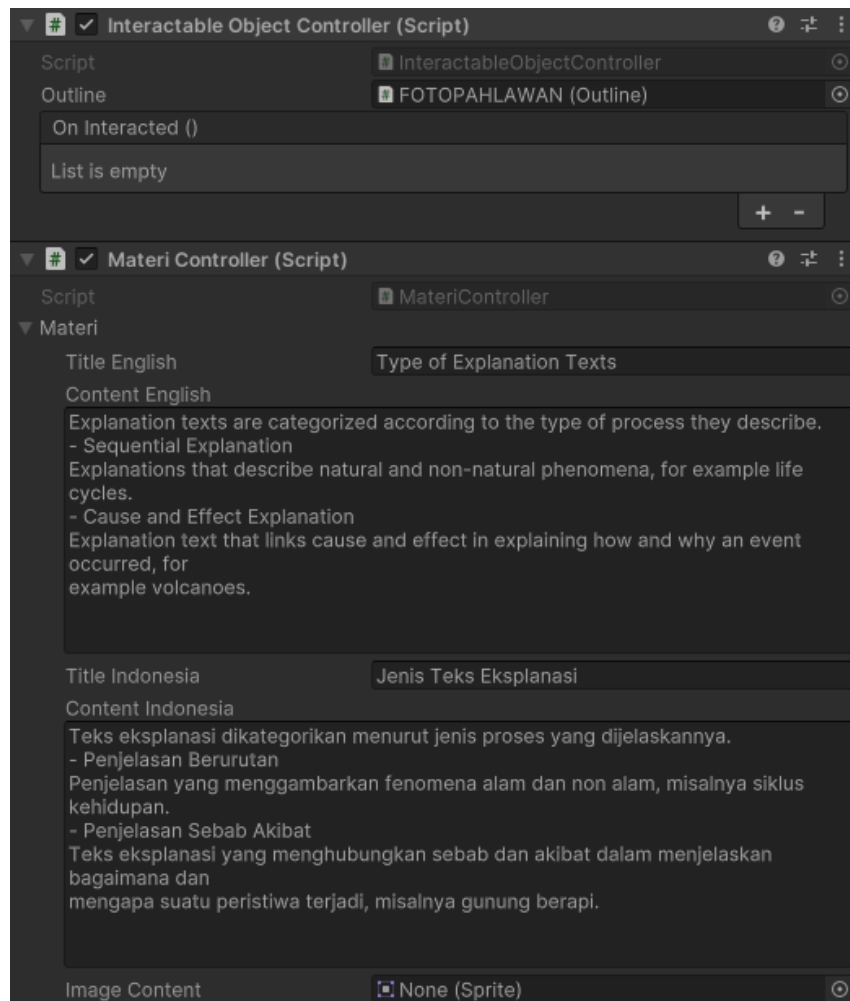
Gambar 4.10 Tampilan Materi dalam Bahasa Inggris

Pada gambar 4.4.7.1, memperlihatkan tampilan materi yang disajikan dalam bahasa Inggris pada antarmuka pengguna (UI). Pada gambar tersebut judul dan isi materi yang ditampilkan menggunakan teks berbahasa Inggris, sesuai dengan data yang tersimpan di variabel `titleEnglish` dan `contentEnglish` pada script `MateriController`. Tampilan ini memberikan opsi bagi pemain untuk membaca materi dalam bahasa Inggris, yang merupakan salah satu fitur dalam game edukasi ini.



Gambar 4.11 Tampilan Materi dalam Bahasa Indonesia

Pada gambar 4.11, Memperlihatkan tampilan materi yang disajikan dalam bahasa Indonesia, pemain dapat mengubah opsi materi dari bahasa Inggris ke bahasa Indonesia. Judul serta isi materi yang ditampilkan menggunakan teks berbahasa Indonesia.

Gambar 4.12 *Materi Controller*

Pada gambar 4.12, menunjukkan tampilan komponen MateriController yang berfungsi untuk mengelola materi pada game di dalam scene. Beberapa property yang terlihat di inspector antara lain :

5. Title English dan Content English
Untuk judul dan isi dari materi dalam Bahasa Inggris.
6. Title Indonesia dan Content Indonesia
Untuk judul dan isi materi dalam Bahasa Indonesia.
7. Image Content
Gambar pendukung yang dapat ditampilkan Bersama materi.

4.5.8 Interactable Objects

Script ini berfungsi untuk mengatur interaksi objek di dalam scene, sekaligus mengelola tampilan outline pada objek tersebut sebagai indikator bahwa objek dapat diinteraksi oleh pemain.

Berikut adalah kode program InteractableObjectController :

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.Events;

public class InteractableObjectController : MonoBehaviour
{
    public Outline outline;
    public UnityEvent OnInteracted;
    void Start()
    {
        outline = GetComponent<Outline>();
    }

    void Update()
    {
    }
}
```

Penjelasan kode :

1. Variabel outline

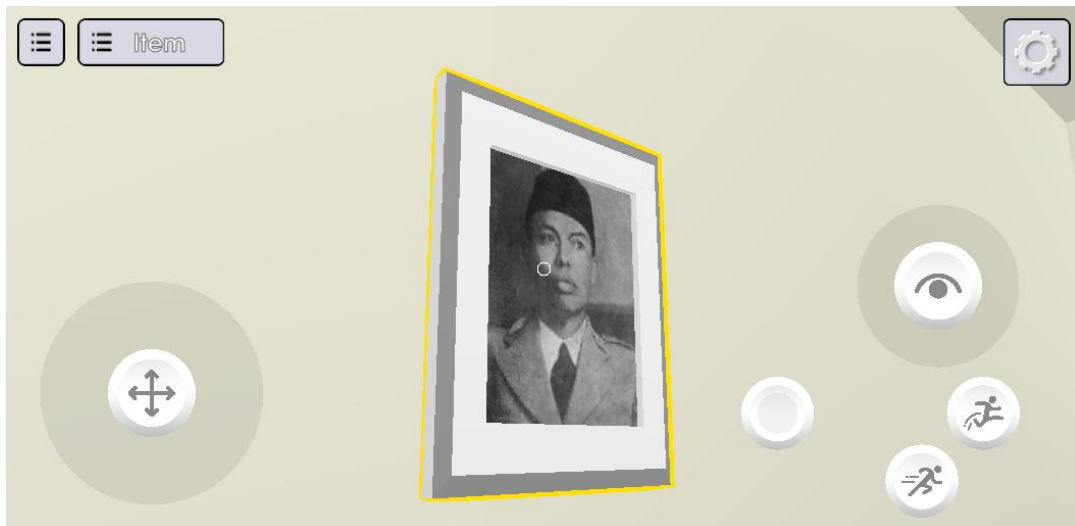
Variabel bertipe Outline yang digunakan untuk mengakses komponen Outline pada GameObject. Komponen Outline berfungsi untuk memberikan efek *highlight* saat objek dapat diinteraksi.

2. Variabel OnInteracted

Variabel bertipe UnityEvent yang berfungsi sebagai event ketika pemain melakukan interaksi dengan objek. Event ini dihubungkan dengan fungsi lain agar dijalankan saat interaksi terjadi.

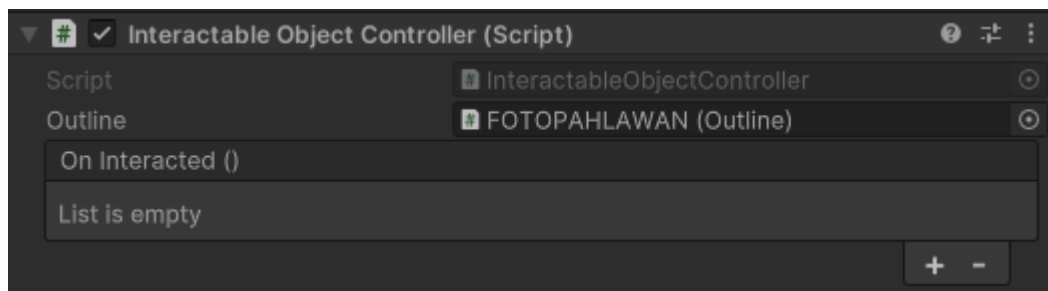
3. Fungsi Start()

Pada awal scene ini dimainkan script akan mengambil komponen outline yang terpasang pada GameObject yang sama. Ini memastikan bahwa script dapat mengontrol efek outline sesuai kebutuhan.



Gambar 4.13 Tampilan Interactable Object

Pada gambar 4.4.8.1, menampilkan bagaimana pemain dapat berinteraksi dengan GameObject yang berisi materi, dengan menampilkan outline yang menandakan bahwa objek tersebut dapat berinteraksi dengan pemain.

Gambar 4.14 *Interactable Object*

Pada gambar 4.4.8.1, ditampilkan komponen `InteractableObjectController` yang terpasang pada sebuah GameObject di dalam scene. Berikut adalah beberapa *property* yang dapat diatur melalui inspector.

4. Outline

Ini adalah referensi komponen Outline yang terpasang pada GameObject yang sama. Komponen ini berfungsi memberikan efek garis tepi (*highlight*) pada objek saat objek tersebut dapat diinteraksi oleh pemain. Dan pada fungsi ini sudah otomatis diisi pada fungsi `Start()` dan tidak perlu diatur manual di inspector.

4.5.9 Game Manager Quiz

Script `GameManager_Quiz` bertanggung jawab mengelola alur kuis yang dimainkan dalam suatu scene. Script ini mencatat jawaban pemain, menghitung skor, mengatur

waktu, serta memberikan akses keluar ruangan jika semua kuis telah selesai atau waktu habis.

Berikut adalah kode program untuk mengelola kuis :

```
using System.Collections;
using System.Collections.Generic;
using System.Linq;
using TMPPro;
using UnityEngine;
using UnityEngine.UI;

public class GameManager_Quiz : MonoBehaviour
{
    // Start is called before the first frame update
    [System.Serializable]
    public class Quiz
    {
        [TextArea(5,5)]
        public string monologue;
        [TextArea(5, 5)]
        public string question;
        public List<Options> options;
        public bool done;
    }
    [System.Serializable]
    public class Options
    {
        [TextArea(2, 2)]
        public string text;
        public bool isCorrect;
    }

    [Header("User Data")]
    public int health;
    public int time;

    [Header("UI")]

    public TMP_Text timerTMP;
    public List<GameObject> healthObject;
    public List<QuizInitiator> quizInitiator;

    public Collider exitCollider;
    public GameObject clueCanvas_Done;

    public int scorePerQuiz;
```

```

public int quizIndex = 0;
public List<Quiz> quizList;
public int totalScore;
[Header("Current Quiz")]
public Quiz activeQuiz;
public bool QuizIsFinished;
private void Start()
{
    exitCollider.enabled = false;
    var initiator = FindObjectsOfType<QuizInitiator>().ToList();
    if (initiator.Count < quizList.Count)
    {
        Debug.LogError("Jumlah Quiz Initiator tidak mencukupi");
        return;
    }

    foreach (var quiz in quizList) {

        initiator[quizList.IndexOf(quiz)].quiz = quiz;

    }
    StartCoroutine(IE_Timer());
}

public IEnumerator IE_Timer()
{
    int counter = time;

    while (counter > 0)
    {
        counter--;
        timerTMP.text = ConvertSecondsToMinutesAndSeconds(counter);
        yield return new WaitForSeconds(1);
        yield return null;
    }

    EndGame(false);
}

public static string ConvertSecondsToMinutesAndSeconds(int
totalSeconds)
{
    int minutes = totalSeconds / 60;
    int seconds = totalSeconds % 60;

    return $"{minutes}:{seconds:D2}"; // Format seconds as two digits
}

```

```

    }

    public void InitiateQuiz(Quiz quiz)
    {
        if (QuizIsFinished)
            return;
        activeQuiz = quiz;
        FindObjectOfType<QuizCanvas>(true).SetupQuiz(activeQuiz);
    }

    public void SetCorrect(bool flag)
    {
        if(flag == false)
        {
            health--;

            healthObject.ForEach(x => x.GetComponent<Image>().color =
healthObject.IndexOf(x) < health ? Color.red: Color.gray);

            if(health == 0)
            {
                EndGame(false);
                return;
            }
        }
        else
        {
            FindObjectsOfType<QuizInitiator>().ToList().Find(x => x.quiz
== activeQuiz).
                GetComponent<InteractableObjectController>().OnInteract
ed.RemoveAllListeners();
            activeQuiz.done = true;
            activeQuiz = null;
            quizIndex++;

        }
        totalScore += flag ? scorePerQuiz : 0;
        if(quizIndex == quizList.Count)
        {
            Debug.Log("Done All Quiz");
            EndGame(true);
        }
    }

    public void EndGame(bool win)
    {
        StopAllCoroutines();
        QuizIsFinished = true;
    }

```

```

        if (win)
        {
            FindObjectOfType<QuizCanvas> (true) .ShowFinalScore();
            GameplayData.instance.SessionProgress.ruangQuiz1Done = true;
        }
        else
        {
            FindObjectOfType<QuizCanvas> (true) .ShowFinalScore();
            GameplayData.instance.SessionProgress.ruangQuiz1Done = true;
        }
        exitCollider.enabled = true;
    }
}

```

Penjelasan kode :

1. Class Quiz
Struktur data untuk menyimpan 1 buah soal kuis. Setiap Quiz memiliki monolog, pertanyaan, pilihan jawaban, dan status done untuk memastikan bahwa soal telah dijawab atau belum.
2. Class Option
Menyimpan pilihan jawaban untuk kuis, terdiri dari text yang merupakan isi dari pilihan jawaban, dan isCorrect untuk memastikan apakah jawaban tersebut tepat atau tidak.
3. Health
Jumlah nyawa pemain saat menjawab kuis. Akan berkurang jika pemain salah menjawab.
4. Time
Waktu yang diberikan untuk menyelesaikan seluruh kuis.
5. healthObject
Daftar UI yang menampilkan visual nyawa. Dalam game ini, visual nyawa diberi dengan ikon hati berwarna.
6. quizInitiator
Daftar objek pemicu kuis di scene. Masing-masing dihubungkan dengan data Quiz.
7. exitCollider
Collider untuk pintu keluar ruangan yang hanya aktif setelah kuis selesai dikerjakan.
8. clueCanvas_Done
UI yang ditampilkan ketika kuis sudah selesai dimainkan.
9. scorePerQuiz
Jumlah skor yang diperoleh untuk setiap kuis yang dijawab dengan benar.
10. quizIndex
Indeks kuis saat ini yang sedang dimainkan.
11. quizList

Daftar semua soal kuis yang akan ditampilkan di scene.

12. totalScore

TotalScore yang dikumpulkan pemain.

13. activeQuiz

Soal kuis yang sedang aktif dimainkan.

14. QuizIsFinished

Menandakan apakah sesi kuis telah berakhir atau belum.

15. Start()

- a. Menonaktifkan exitCollider.
- b. Mengambil semua QuizInitiator dari scene, kemudian memasangkan data Quiz dari quizList ke masing-masing initiator.
- c. Memulai timer dengan coroutine IE_Timer()

16. IE_Timer()

Coroutine yang menghitung waktu mundur dari nilai time. Jika waktu habis sebelum semua kuis selesai, permainan berakhir.

17. ConvertSecondsToMinutesAndSeconds()

Fungsi untuk mengubah detik menjadi format menit:detik.

18. InitiateQuiz(Quiz quiz)

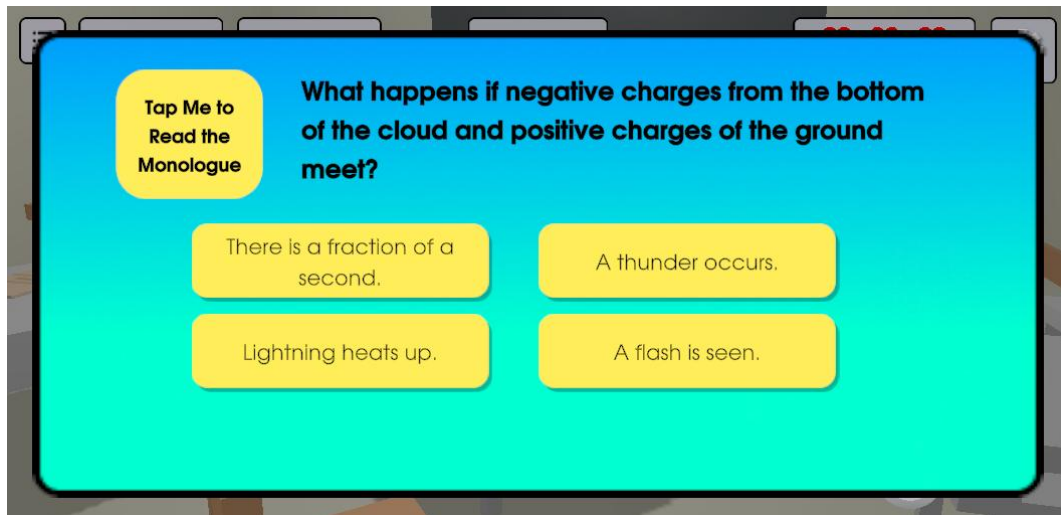
Fungsi untuk memulai kuis tertentu. Memanggil QuizCanvas dan menampilkan soal dari activeQuiz.

19. SetCorrect(bool flag)

Dipanggil ketika pemain menjawab soal. Jika flag == false, nyawa akan dikurangi. Dan jika nyawa habis maka game akan berakhir. Jika flag == true, soal ditandai selesai, skor ditambah, dan lanjut ke kuis berikutnya. Dan jika semua kuis sudah dikerjakan maka akan memanggil EndGame(true).

20. EndGame(bool win)

Mengakhiri sesi kuis. Apapun hasilnya baik menang atau kalah, skor akhir ditampilkan, progress diupdate, dan pintu keluar exitCollider dibuka.



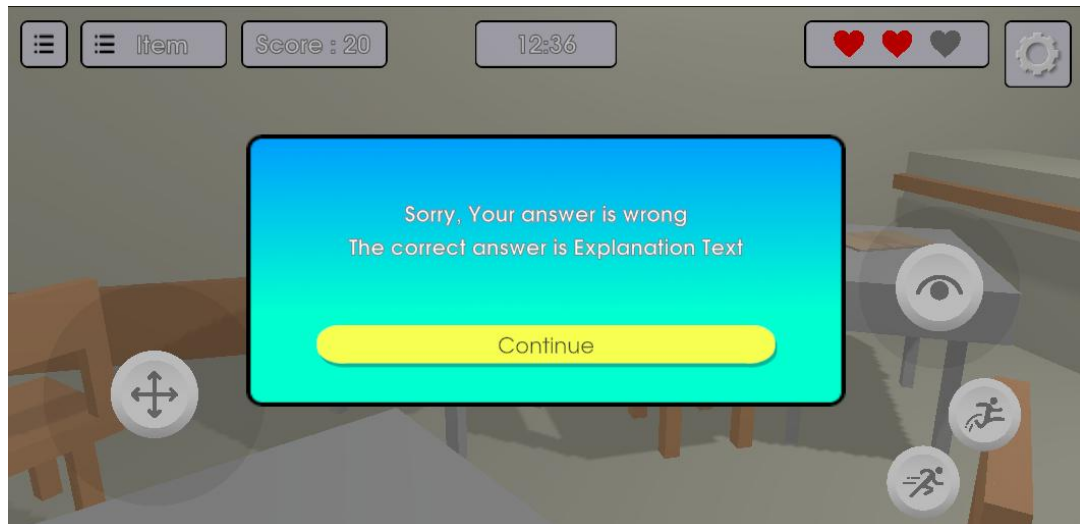
Gambar 4.15 Tampilan Soal

Pada gambar 4.14, menampilkan tampilan soal kuis yang terdiri dari monolog pengantar, pertanyaan, dan beberapa pilihan jawaban yang bisa dipilih oleh pemain.



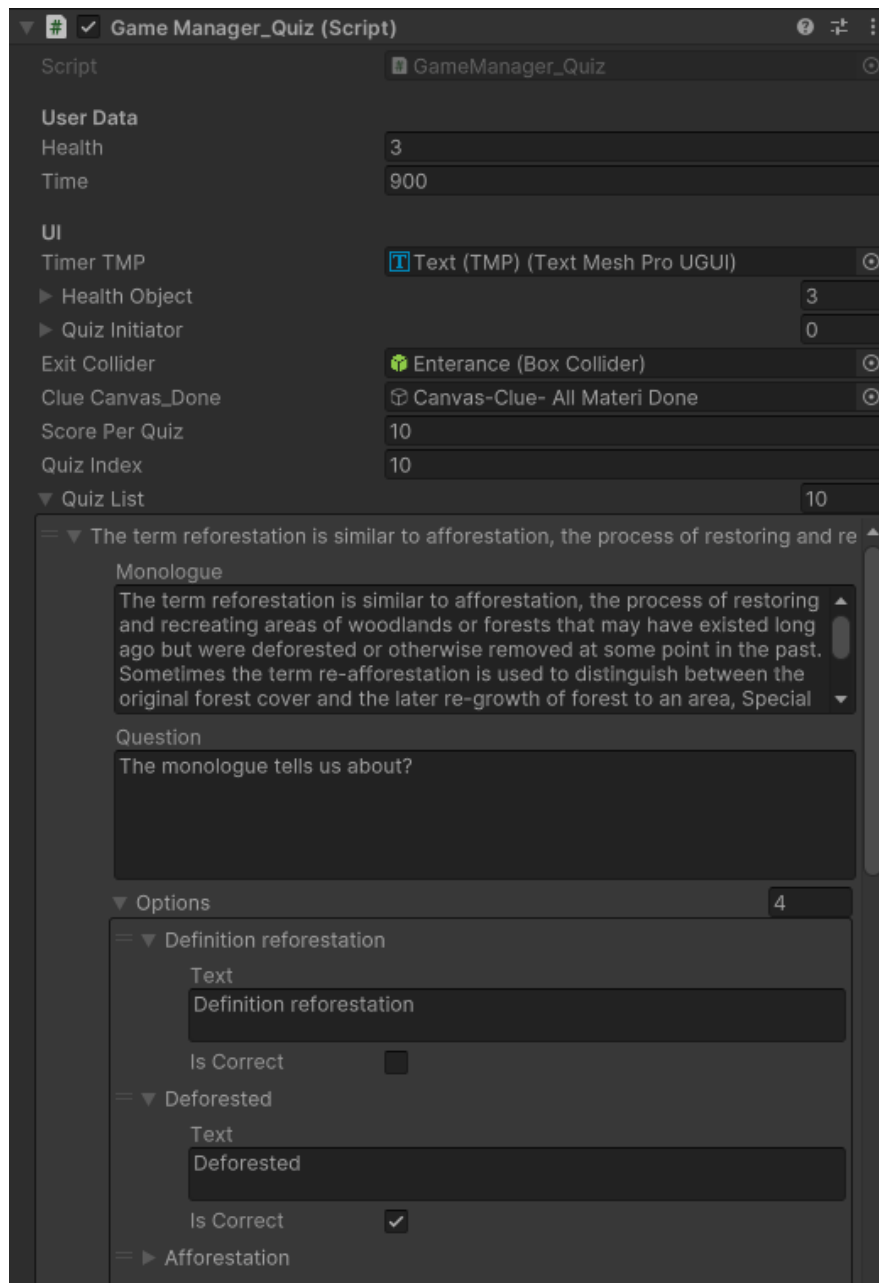
Gambar 4.16 Tampilan Menjawab Soal Dengan Benar

Pada gambar 4.15, menampilkan *feedback* yang muncul setelah pemain menjawab soal dengan benar. Feedback ini berupa notifikasi skor tambahan yang diperoleh pemain.



Gambar 4.17 Tampilan Menjawab Soal Dengan Salah

Pada gambar 4.16, menampilkan *feedback* ketika pemain menjawab soal dengan salah. Pada kondisi ini, pemain tidak mendapatkan skor dan nyawa berkurang, memberikan konsekuensi langsung atas kesalahan agar pemain lebih berhati-hati dalam menjawab soal berikutnya.

Gambar 4.18 *GameManager Quiz*

Pada gambar 4.17, merupakan inspector GameManager_Quiz ditampilkan berbagai properti yang mengatur jalannya kuis. Mulai dari pengaturan nyawa, dan waktu pengerjaan kuis, Komponen UI seperti timer, komponen Collider, canvas ketika selesai menjawab soal, score per quiz, quiz index yang merupakan kuis yang telah dikerjakan, serta quiz list yang berisi soal dan jawaban.

4.5.10 Quiz Initiator

Script QuizInitiator bertugas sebagai penghubung antara objek interaktif di scene dengan sistem kuis. Script ini menyimpan referensi ke sebuah soal kuis dan akan

memicu tampilan kuis ketika objek tersebut diinteraksikan oleh pemain, yang dimana soal tersebut belum pernah dikerjakan.

Berikut adalah kode program QuizInitiator :

1. Quiz

Menyimpan data kuis yang akan dipicu oleh objek ini. Data ini merupakan satu soal dari GameManager_Quiz.

2. Start()

Fungsi ini mengambil komponen InteractableObjectController yang bertugas mengatur interaksi pemain dengan objek. Kemudian menambahkan event listener InitiateQuiz pada event OnInteracted, sehingga ketika pemain berinteraksi, fungsi InitiateQuiz() akan dipanggil.

3. InitiateQuiz()

Fungsi ini mengecek apakah soal kuis sudah pernah dikerjakan dengan ditandai properti done. Jika sudah, fungsi akan berhenti dan tidak memulai kuis lagi. Jika belum, maka fungsi akan memanggil *method* InitiateQuiz() dari GameManager_Quiz dan mengirim data soal yang terkait untuk ditampilkan.

BAB V

PENGUJIAN

Bab ini akan membahas pengujian yang dilakukan terhadap *game* edukasi Kuro Game yang telah dikembangkan menggunakan metode *Digital Game Based Learning* (DGBL). Pengujian dilakukan untuk memastikan bahwa *game* berjalan sesuai dengan fungsi yang telah dirancang dalam *Game Design Document* (GDD). Dalam bab ini dijelaskan tujuan pengujian, metode pengujian dan hasil dari pengujian.

5.1 Tujuan Pengujian

Pengujian dilakukan untuk memastikan bahwa seluruh fitur dan sistem yang telah dirancang pada Kuro Game berjalan sesuai dengan fungsi dan skenario yang diharapkan. Pengujian ini penting untuk mengidentifikasi apakah sistem interaksi, alur permainan, serta penyajian materi dan soal bekerja secara konsisten dan responsif terhadap aksi pemain.

Secara lebih spesifik, tujuan pengujian ini adalah :

1. Memastikan bahwa setiap komponen sistem, seperti interaksi dengan NPC, objek edukatif, sistem kuis, dan sistem penyimpanan progres, berfungsi sesuai rancangan awal.
2. Mengetahui apakah alur interaksi antara pemain dan lingkungan *game* dapat berjalan secara natural dan tidak menimbulkan *error* atau gangguan pada saat dimainkan.
3. Menilai Tingkat kestabilan dan keandalan aplikasi pada berbagai skenario umum yang akan dihadapi pemain saat bermain.
4. Mendeteksi potensi bug atau kekeliruan logika dalam *gameplay* sebelum *game* dipublikasikan lebih lanjut atau diuji oleh pengguna sebenarnya.
5. Menjamin bahwa pengalaman bermain tetap intuitif, terutama karena target pengguna adalah pemain mobile yang membutuhkan kemudahan dan kenyamanan navigasi.

5.2 Metode Pengujian

Metode yang digunakan dalam proses ini adalah pengujian *Black Box* (*Black Box Testing*), dimana pengujian dilakukan berdasarkan masukan (*input*) dan keluaran (*Output*) dari sistem tanpa melihat atau mengevaluasi struktur kode program secara langsung.

Black box testing digunakan karena fokus utama pengujian ini adalah pada fungsionalitas sistem dari sudut pandang pengguna. Dalam konteks Kuro Game, pengujian dilakukan dengan memainkan *game* secara langsung, kemudian mencatat setiap hasil dari aksi tertentu, seperti :

1. Apakah dialog muncul saat mendekati NPC?
2. Apakah tombol interaksi muncul saat mengarahkan *crosshair* ke objek?
3. Apakah materi atau kuis ditampilkan dengan benar saat tombol ditekan?
4. Apakah progres pemain disimpan dengan baik saat berpindah scene?

Pengujian dilakukan secara manual, tanpa bantuan *tools* otomatis, karena kompleksitas *gamemasih* dalam skala menengah dan dapat diuji secara menyeluruh oleh pengembang.

5.3 Skenario dan Hasil Pengujian

Pengujian dilakukan terhadap beberapa fitur utama dalam Kuro Game. Berikut ini adalah daftar fitur yang diuji, skenario pengujian, hasil yang diharapkan, dan status hasil pengujian berdasarkan uji internal pengembang:

Tabel 5.1 Hasil Pengujian

No	Fitur yang Diuji	Skenario Pengujian	Hasil yang Diharapkan	Status
1	Main Menu	Pemain membuka <i>game</i> dan menekan tombol <i>Play</i>	Tombol <i>Play</i> berpindah ke scene permainan	Berhasil
2	Interaksi dengan NPC	Pemain mendekati NPC (Guru/Satpam) dalam jarak tertentu	Dialog muncul secara otomatis	Berhasil
3	Perpindahan Scene	Pemain mendekati pintu atau area akses ruangan	Sistem berpindah secara otomatis ke ruangan yang sesuai	Berhasil
4	Interaksi dengan Objek (Materi)	Pemain mengarahkan <i>crosshair</i> ke objek materi (Buku, pensil, dll)	Materi pembelajaran ditampilkan dengan menekan tombol interaksi	Berhasil
5	Interaksi dengan Objek (Kuis)	Pemain mengarahkan <i>crosshair</i> ke objek kuis (Buku, pensil, dll)	Kuis dimulai dan soal ditampilkan	Berhasil
6	Sistem skor kuis	Pemain menjawab soal kuis dengan benar dan salah	+10 point untuk setiap jawaban benar, dan jika 3 kali salah menjawab, kuis selesai	Berhasil
7	Timer Kuis	Pemain memasuki ruangan kuis dan waktu dimulai dalam 15 menit untuk menjawab soal	Waktu mulai berkurang ketika pemain memasuki ruangan	Berhasil
8	Penyimpanan progres pemain	Pemain menyelesaikan materi atau kuis, lalu berpindah ke scene lain	Progres tetap tersimpan dan daapt diakses Kembali setelah pindah scene	Berhasil
9	Akses scene berdasarkan progress	Pemain mencoba masuk ke ruangan yang belum diizinkan sebelum menyelesaikan materi/kuis	Tidak bisa memasuki ruangan	Berhasil

5.4 Evaluasi Hasil Pengujian

Dari hasil pengujian internal yang dilakukan oleh pengembang, dapat disimpulkan bahwa seluruh fitur utama telah berjalan sesuai rencana dan tidak ditemukan bug yang mengganggu fungsionalitas.

1. Dialog NPC muncul secara otomatis saat pemain berada dalam jarak interaksi.
2. Tombol interaksi muncul saat pemain mengarahkan crosshair ke objek edukatif, dan sistem merespons dengan menampilkan materi atau soal sesuai alurnya.
3. Transisi antar scene berjalan lancar tanpa error, dan data progres tersimpan dengan baik.
4. Antarmuka pada perangkat mobile berfungsi dengan baik dan elemen mudah diakses.

Namun, karena pengujian belum dilakukan kepada pengguna sebenarnya, maka hasil ini bersifat sementara dan memerlukan tahap uji coba lebih lanjut untuk menilai kenyamanan, pengalaman pengguna, dan efektivitas edukasi dari sisi pemain.

BAB VI

PENUTUP

Bab ini merupakan penutup dari laporan tugas akhir yang berisi kesimpulan dari hasil analisis, pembangunan, dan pengujian *game* edukasi Kuro Game. Selain itu, bab ini juga memuat saran untuk pengembangan lebih lanjut agar *game* dapat lebih bermanfaat dan efektif dalam proses pembelajaran.

6.1 Kesimpulan

Berdasarkan hasil perancangan, implementasi, dan pengujian yang telah dilakukan terhadap *game* edukasi Kuro Game, maka dapat disimpulkan bahwa :

1. Pengembangan Kuro Game berhasil mengintegrasikan berbagai elemen penting dalam *game* seperti interaksi dengan NPC, navigasi antar scene, sistem pembelajaran, dan kuis ke dalam satu sistem yang mendukung proses pembelajaran Bahasa Inggris secara interaktif. *Game* ini dirancang agar mendorong pemain untuk aktif belajar melalui media yang menyenangkan.
2. Struktur pemrograman disusun secara modular dengan pemisahan fungsi dalam komponen-komponen yang jelas, seperti *GameManager*, *MateriController*, *QuizInitiator*, dan lainnya. Hal ini membantu dalam pengelolaan alur eksekusi kode agar setiap fitur dapat berjalan sesuai dengan permainan dan tidak saling bertabrakan.
3. Pengujian internal menunjukkan bahwa seluruh fitur utama dapat berjalan dengan baik pada perangkat *mobile*. Tidak ditemukan error kritis dan antarmuka *game* berjalan dengan responsive, baik dalam interaksi dengan objek maupun saat menampilkan konten pembelajaran dan soal.

Dengan demikian, pengenmangan Kuro Game telah menjawab seluruh identifikasi masalah yang diajukan dalam tugas akhir ini. *Game* ini dinilai layak untuk dijadikan sebagai salah satu alternatif media pembelajaran berbasis digital.

6.2 Saran

Untuk pengembangan lebih lanjut, terdapat beberapa saran yang dapat dipertimbangkan:

1. Melakukan pengujian langsung kepada pengguna
Uji coba dengan melibatkan siswa atau target pengguna langsung akan memberikan masukan yang lebih objektif terhadap kenyamanan, efektivitas, dan daya tarik *game* sebagai media belajar.
2. Mengoptimalkan performa *game*
Pengoptimalan diperlukan agar *game* tetap berjalan lancar di perangkat dengan spesifikasi rendah. Ini dapat dilakukan dengan pengurangan asset tidak penting, kompresi pada gambar, dan pengelolaan memori yang efisien.

3. Membuat struktur kode yang lebih terorganisir

Penulisan kode sebaiknya dibuat lebih terstruktur dengan memisahkan fungsi-fungsi ke dalam *script* yang berbeda sesuai tugasnya. Hal ini bertujuan agar sistem lebih mudah dalam pemeliharaan, dipahami, dan dikembangkan jika di masa depan ingin menambahkan fitur baru.

DAFTAR PUSTAKA

- [AHM19] Ahmad, I., Abdullasim, N., “Game Design Issues, Trend and Challenges”, Malaysia, Universiti
- [BAP15] Pranata, B, A., 2015., “Mudah Membuat Game dan Potensi Finansialnya dengan Unity 3D”, Jakarta, PT Elex Media Komputindo
- [CIP20] Cipta, M, R, E., 2020., “Membuat Game Edukasi Matematika Berbasis Android menggunakan Unity”, Skripsi, Ilmu Komputer – Universitas Hasanudin
- [KUS18] Yudha, C, B., 2018., “Penerapan Game Edukasi Berbasis Android Dan Gambar Bagi Siswa Sekolah Dasar”, Jurnal, Tunas Bangsa
- [PRA21] Prasetyo, R, M, M., Syaputra, H., Cholil, W., Sauda, S., 2021., “Rancang dan Bangun Game Edukasi Anak-Anak Berbasis Android Dengan Unity Menggunakan Metode Game Development Life Cycle”, Jurnal, Ilmu Komputer – Universitas Bina Darma
- [SIT17] Asmiatun, S., Putri, A, N., 2017., “Belajar Membuat Game 2D dan 3D Menggunakan Unity”, Sleman, Deepublish
- [WAH22] Wahyu, S., 2022., “Penerapan Metode Game Development Life Cycle Pada Pengembangan Aplikasi Game Pembelajaran Budi Pekerti”, Jurnal, Teknik Informatika – Universitas Esa Unggul
- [WAJ20] Wajid, A., 2020., “Pembangunan Game Edukasi Pembelajaran Algoritma Pengulangan Berbasis Web Menggunakan Metode *Digital Game Based Learning-Instructional*”, Tugas Akhir, Teknik Informatika – Universitas Pasundan
- [RAM16] Ramdhani, F, B., 2016., “Pembangunan Game Edukasi Anak Usia Dini Menggunakan Metode Game Development Life Cycle (Studi Kasus : TK BHAYANGKARI 31 KANDANGHAUR)”, Tugas Akhir, Teknik Informatika – Universitas Pasundan
- [DEB19] Debbie, N, S., 2017., “Perancangan Game Edukasi “Kiding” Sebagai Media Pembelajaran Menggunakan Unity Engine Berbasis Multimedia”, Skripsi, Teknik Informatika – Universitas Islam Negeri Syarif Hidayatullah
- [ERI16] Ericksoon, H, A., Kuswardayan, I., Dr. Suciati, E, N., 2016., “Rancang Bangun Game Berhitung *Spaceship* dengan Pengendali Suara Menggunakan *Speech Recognition Plugin* pada *Unity*”, Jurnal, Teknik Informatika – Institut Teknologi Sepuluh Nopember
- [MUS21] Mustofa., Putra, J, L., Kesuma, C., 2021., “Penerapan Game Development Life Cycle Untuk Video Game Dengan Model Role Playing Game”, Jurnal, Teknik Informatika – Sekolah Tinggi Manajemen Informatika dan Komputer Nusa Mandiri
- [RIA19] Rianingtias, O., 2019., “Pengembangan Game Edukasi Berbasis Android Sebagai Media Pembelajaran Biologi Bernuansa Motivasi Siswa Kelas XI di SMA/MA”, Skripsi, Pendidikan Biologi – Universitas Islam Negeri Raden Intan Lampung
- [RIF15] Rifai, W, A., 2015., “Pengembangan Game Edukasi Lingkungan Berbasis Android”, Tugas Akhir, Pendidikan Teknik Informatika – Universitas Negeri Yogyakarta
- [PUT21] Putra, M, C, T., 2021., “Game Edukasi Penyelamatan Diri Dari Virus Covid-19 Berbasis 3D Menggunakan Game Engine Unity”, Tugas Akhir, Teknik Informatika – Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Palangkaraya

- [NIN20] Ningsih, G, W., 2021., “Game Edukasi Sejarah di Indonesia Sebagai Media Pembelajaran Berbasis Android”, Skripsi, Teknik Informatika – Universitas Islam Negeri Alauddin Makassar
- [SAK] Sakdiah, H., Fuadi, W., Risawandi., “Aplikasi Game Edukasi Tebak Gambar Ulama Se-Aceh Menggunakan Metode Rolling Hash Berbasis Android”, Jurnal, Teknik Informatika – Universitas Malikussaleh
- [SIR20] Siregar, M, R., Nelmiawati. B.CS., M.Comp.Sc., “*Game 3D “Lawan Narkoba” Menggunakan Metode Game Development Life Cycle (GDLC)*”, Jurnal, Teknik Informatika – Politeknik Negeri Batam
- [SOE16] Soeheri, 2016., “DGBL-ID (*Digital Game Based Learning*) Sebagai Arsitektur Perancangan *Game* Edukasi”, Jurnal, Teknik Informatika – Universitas Potensi Utama, Medan
- [ANG21] Anggraini H, I., Nurhayati., Kusumaningrum, S, R., “Penerapan Media Pembelajaran *Game* Matematika Berbasis *HOTs* Dengan Metode *Digital Game Based Learning* (DGBL) di Sekolah Dasar”, Jurnal, Universitas Negeri Malang
- [SUN] Sundayana, W., “Pedoman Umum Pembelajaran Bahasa Inggris di SMA”, Direktori FPBS Jurusan Pendidikan Bahasa Inggris – Universitas Pendidikan Indonesia
- [RAM16] Ramdhani, F, B., 2016., “Pembangunan *Game* Edukasi Anak Usia Dini Menggunakan Metode *Game Development Life Cycle*”, *Tugas Akhir*, Teknik Informatika – Universitas Pasundan, Bandung
- [APR23] Aprian, W., 2023., “Perancangan Game Edukasi Bahasa Inggris Menggunakan Digital Game Based Learning (DGBL)”, *Tugas Akhir*, Teknik Informatika – Universitas Pasundan, Bandung
- [PAR20] Parluka, R., Nisaa, T, A., Ningrum, S., M., Haque, B, A., “Studi Literatur Kekurangan dan Kelebihan Pengujian Black Box”, Jurnal, Informatika – Universitas Pembangunan Nasional, Jawa Timur

LAMPIRAN

LAMPIRAN A PERIZINAN DAN PEMBAHASAN KURIKULUM



UNIVERSITAS PASUNDAN
Fakultas Teknik

Teknik Industri ☐ 022 - 2019335
Teknologi Pangan ☐ 022 - 2019339
Teknik Mesin ☐ 022 - 2019352
Teknik Informatika ☐ 022 - 2019371
Teknik Lingkungan ☐ 022 - 2009574
Perencanaan Wilayah dan Kota ☐ 022 - 2006466

Bandung, 19 Mei 2023

No : 0037/Unpas.FT.IF/Q/ V /2023
Hal : Penelitian Tugas Akhir
Lamp : -

Kepada Yth,
Bapak/Ibu Kepala
Sman 1 LEMAHABANG
Jl. Syech Quro, Kedawung, Kec. Lemahabang, Kec. Karawang, Kab. Karawang

Dengan hormat,
Bersama ini kami perkenalkan 1 (satu) orang mahasiswa Program Studi Teknik Informatika
Fakultas Teknik Universitas Pasundan.

Nama : Wahyu Aprian - 193040141
Ferri Syamsul M. - 19304045
Hikmat Fadrial R - 193040138
Chandra Dwi R. - 193040133

Program Studi : Teknik Informatika.

Untuk menyelesaikan studinya, mahasiswa tersebut di atas bermaksud untuk
Wawancara/survey, Penelitian Tugas Akhir, untuk mengumpulkan data di Perusahaan yang
Bapak / Ibu Pimpin.

Sehubungan dengan itu kami mohon dengan hormat agar mahasiswa tersebut di atas
di beri izin mengumpulkan data yang diperlukan untuk penelitian tersebut.

Atas perhatian, bantuan dan kerjasama yang baik, kami ucapkan terima kasih.

Program Studi Teknik Informatika
Universitas Pasundan.
Ketua,


(Dr. Ayt Purnasari, ST., MT.)

Tembusan

Jl. Arsin
Jl. Setiabudi No. 193 Bandung - 40153 DEKANAT ☎ 022 - 2019435, (HUNTING) 2019433, 2019407 Fax. 022 - 2019329

 Dipindai dengan CamScanner

Surat Pengantar Penelitian Tugas Akhir



Izin untuk Wawancara dan Quisioner Siswa

LAMPIRAN B BERITA ACARA



UNIVERSITAS PASUNDAN
Fakultas Teknik

Teknik Industri ☐ 022 - 2019335
Teknologi Pangan ☐ 022 - 2019339
Teknik Mesin ☐ 022 - 2019352
Teknik Informatika ☐ 022 - 2019371
Teknik Lingkungan ☐ 022 - 2009574
Perencanaan Wilayah dan Kota ☐ 022 - 2006466

BERITA ACARA PENYEBARAN KUISIONER

Pada hari Kamis, 25 Mei 2023.

Telah dilaksanakan penyebaran kuisisioner yang berkaitan dengan penelitian yang akan dilakukan untuk memenuhi Tugas Akhir Strata Satu (S1).

Tempat : SMA Negeri 1 Lemahabang Karawang

Responden : Siswa siswi kelas XI IPA 1

Peneliti : Wahyu Aprian

Prodi / Fakultas : Teknik Informatika / Fakultas Teknik

Pihak pewawancara melakukan wawancara dengan pihak narasumber yang berkaitan dengan penelitian yang dilakukan di SMA Negeri 1 Lemahabang Karawang tentang "Perancangan Game Edukasi Bahasa Inggris Menggunakan Digital Game Based Learning (DGBL)", kemudian narasumber memberikan jawaban terkait pertanyaan yang diajukan oleh pewawancara. Adapun pertanyaan yang diajukan serta hasil terlampir.

Karawang, 25 Mei 2023.....

Mengetahui,

Peneliti

Wahyu Aprian
NRP. 193040141

Wakasek Kesiswaan

(.....Teja Hidayat.....)

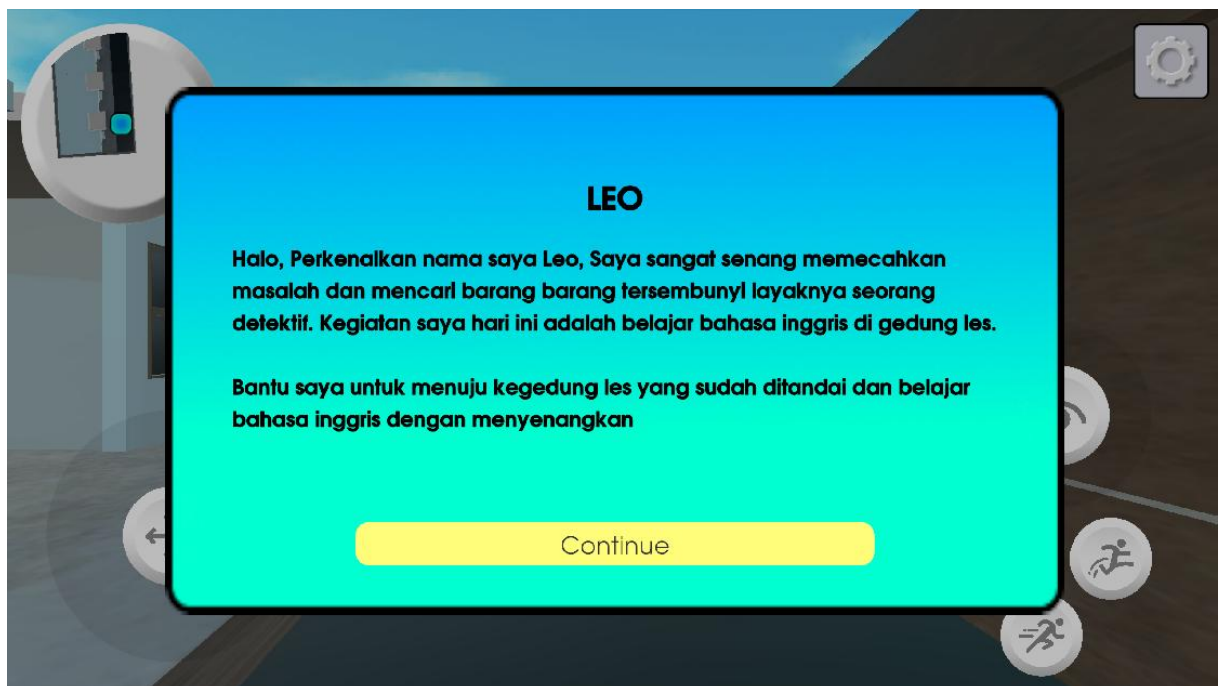


Quisioner Siswa

LAMPIRAN C TAMPILAN GAME



Lampiran Main Menu



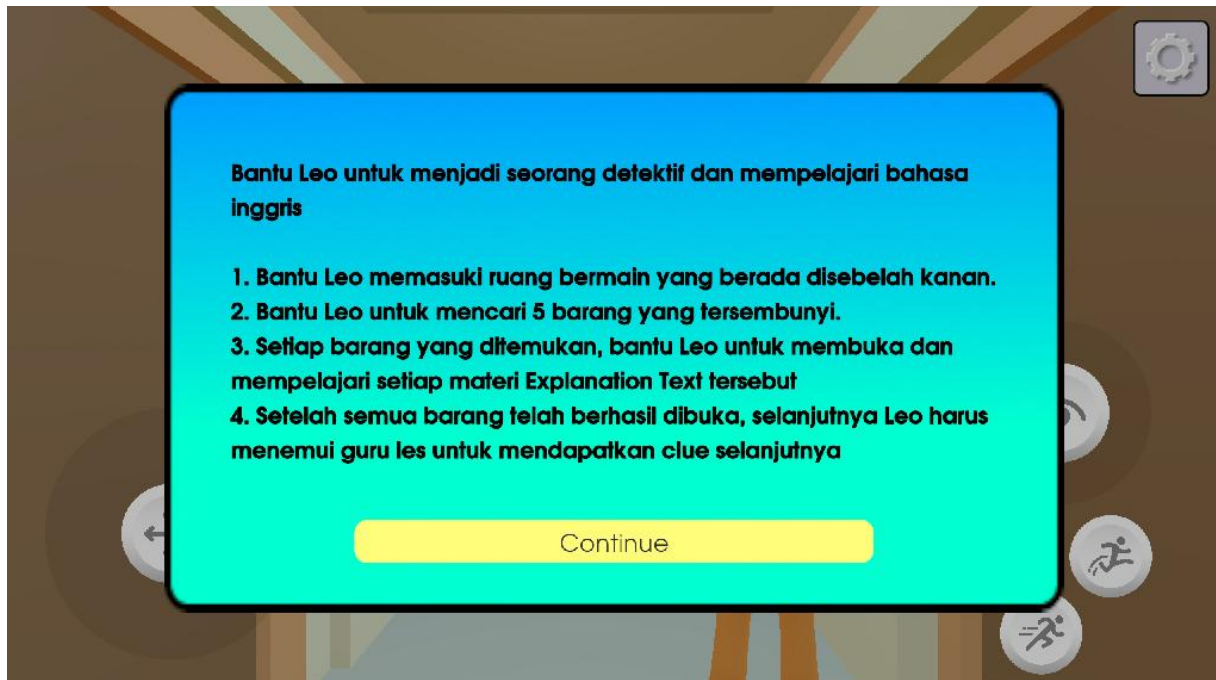
Lampiran Intro Pemain



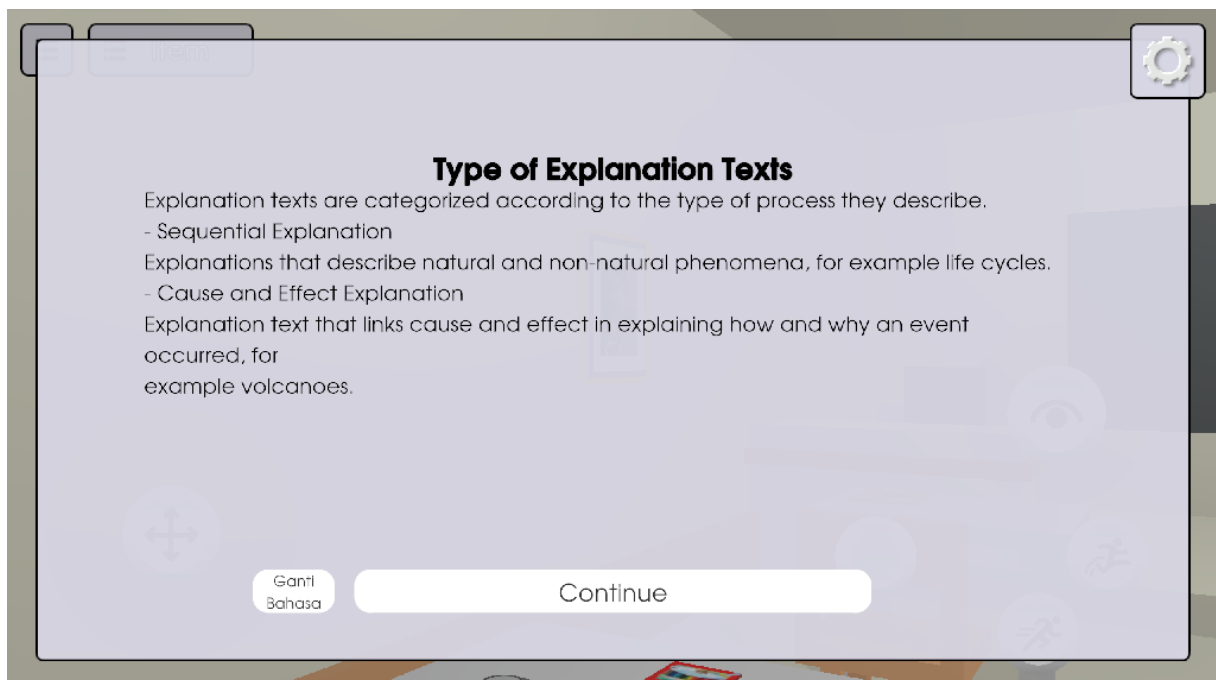
Lampiran Interaksi NPC Satpam



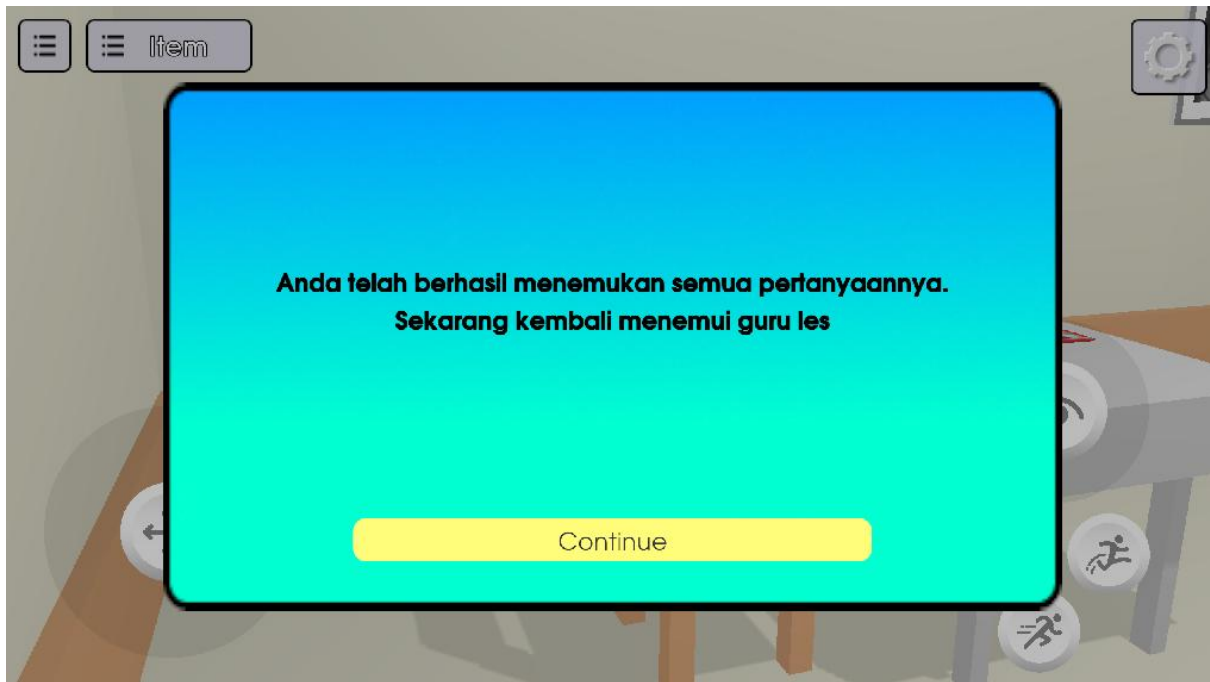
Lampiran Interaksi NPC Guru



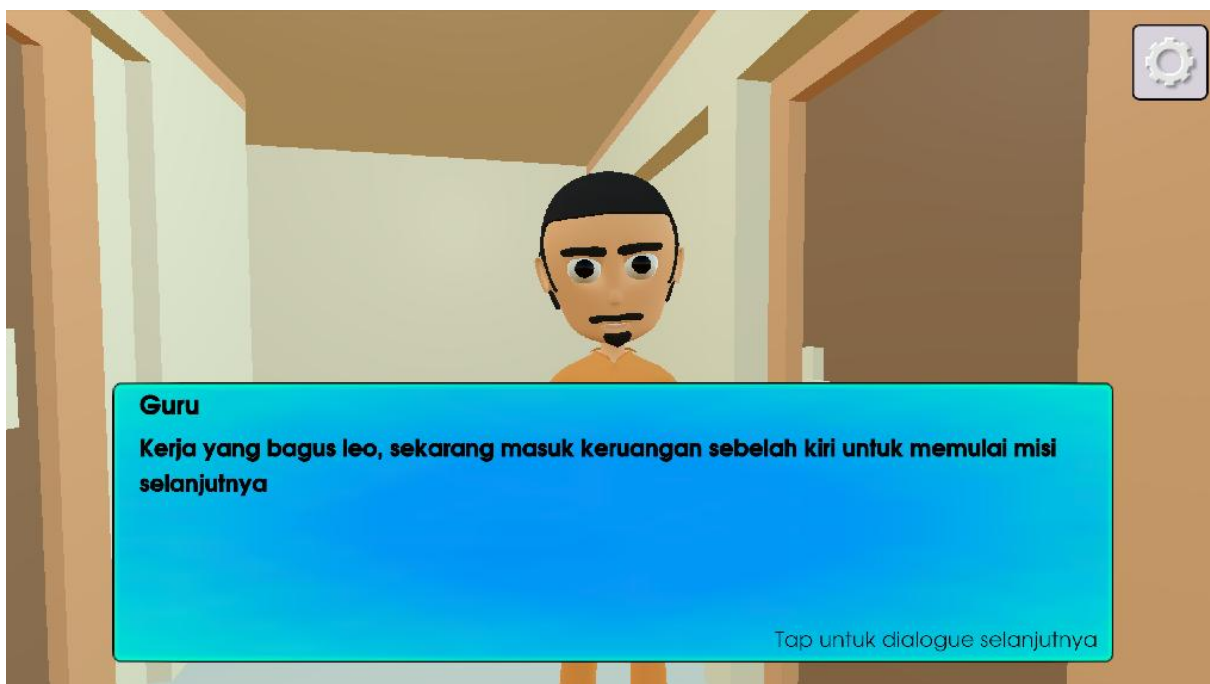
Lampiran Clue Untuk Pemain Mencari Materi



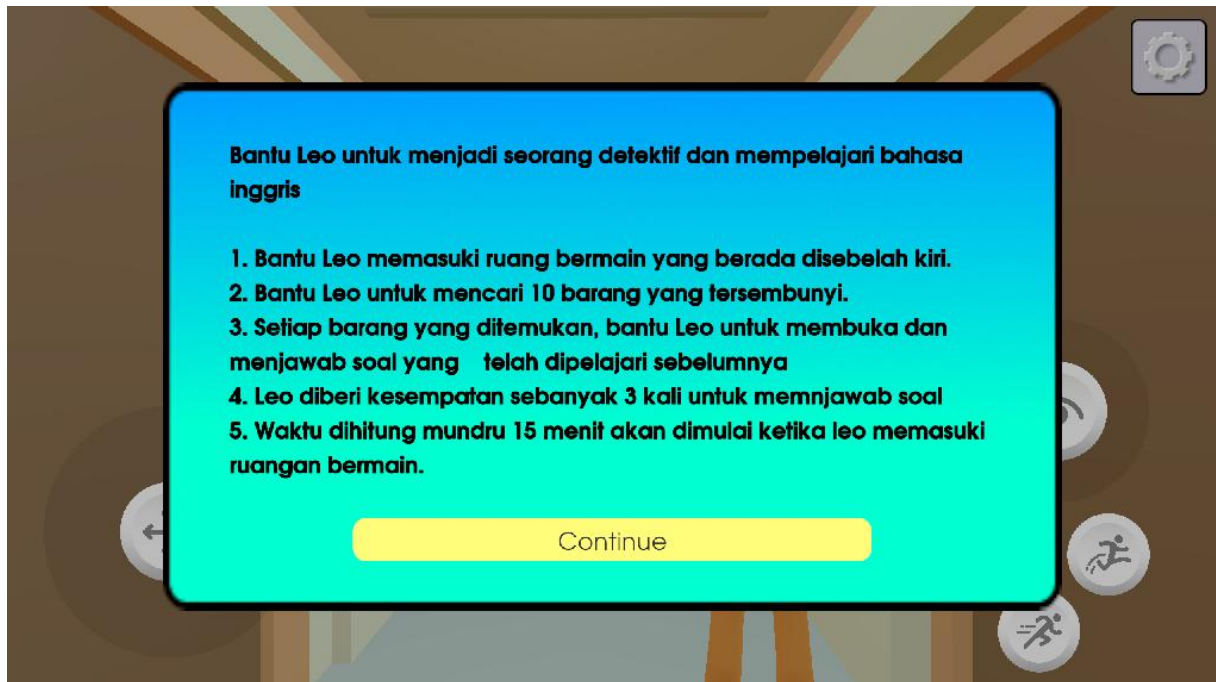
Lampiran Tampilan Materi



Lampiran Selesai Menyelesaikan Materi



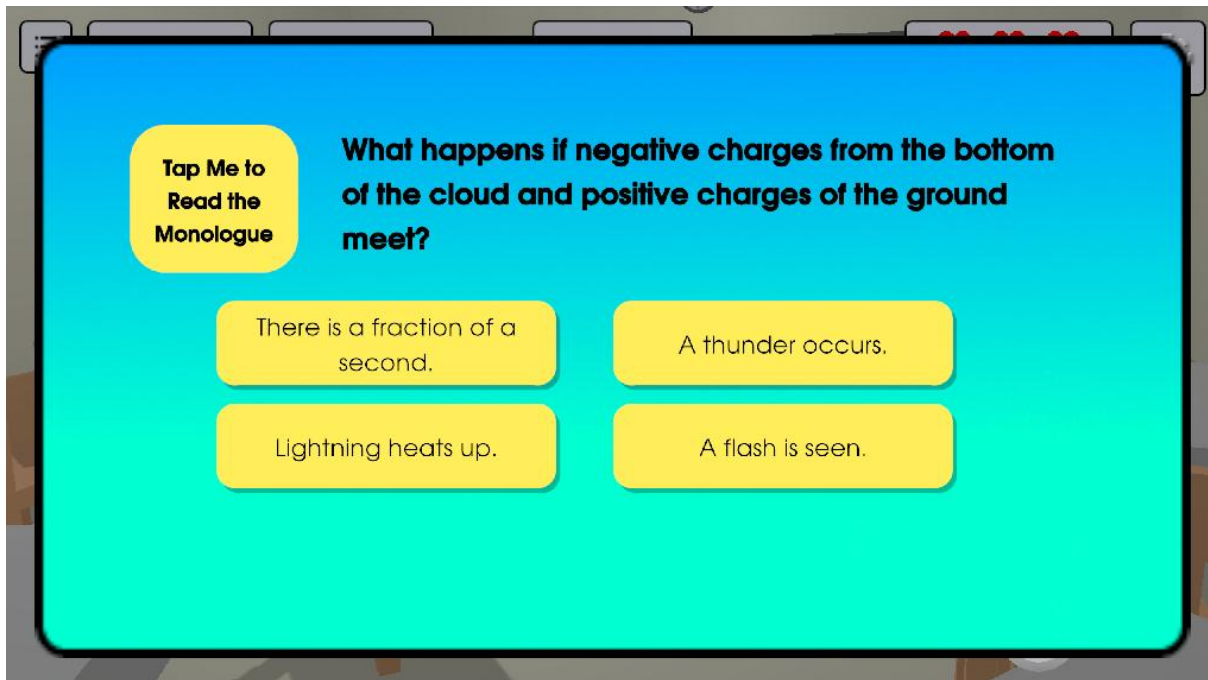
Lampiran NPC Setelah Menyelesaikan Materi



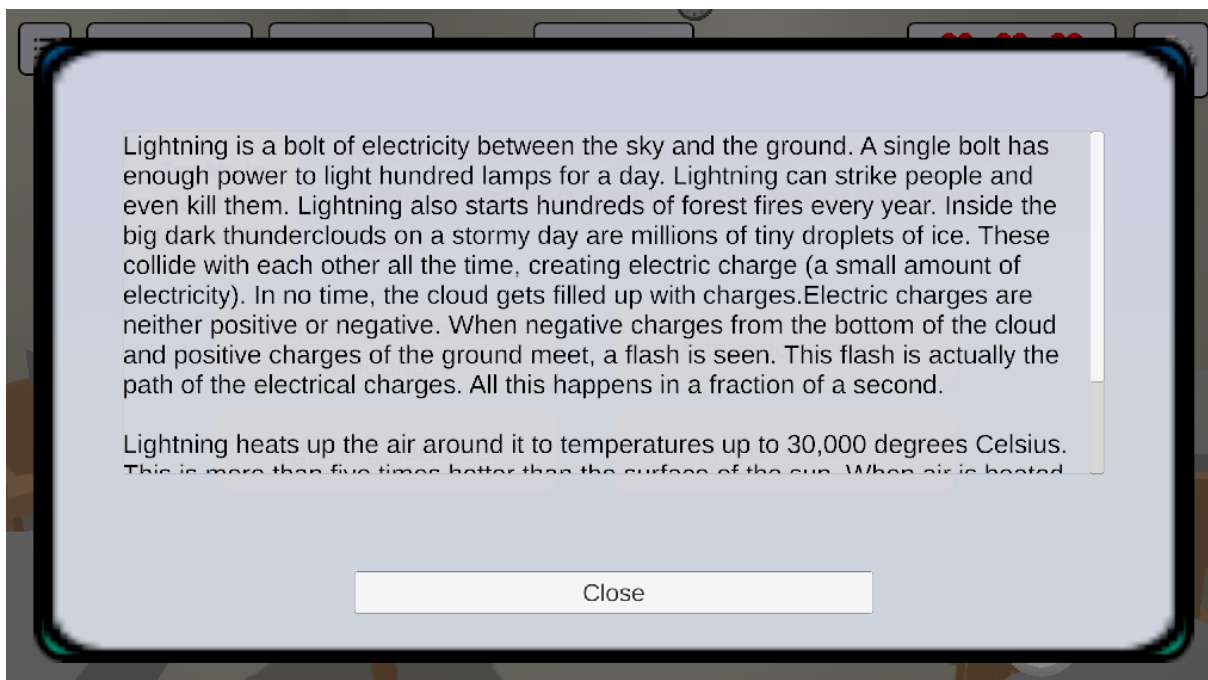
Lampiran Clue Untuk Memasuki Ruang Soal



Lampiran Ruangan Soal



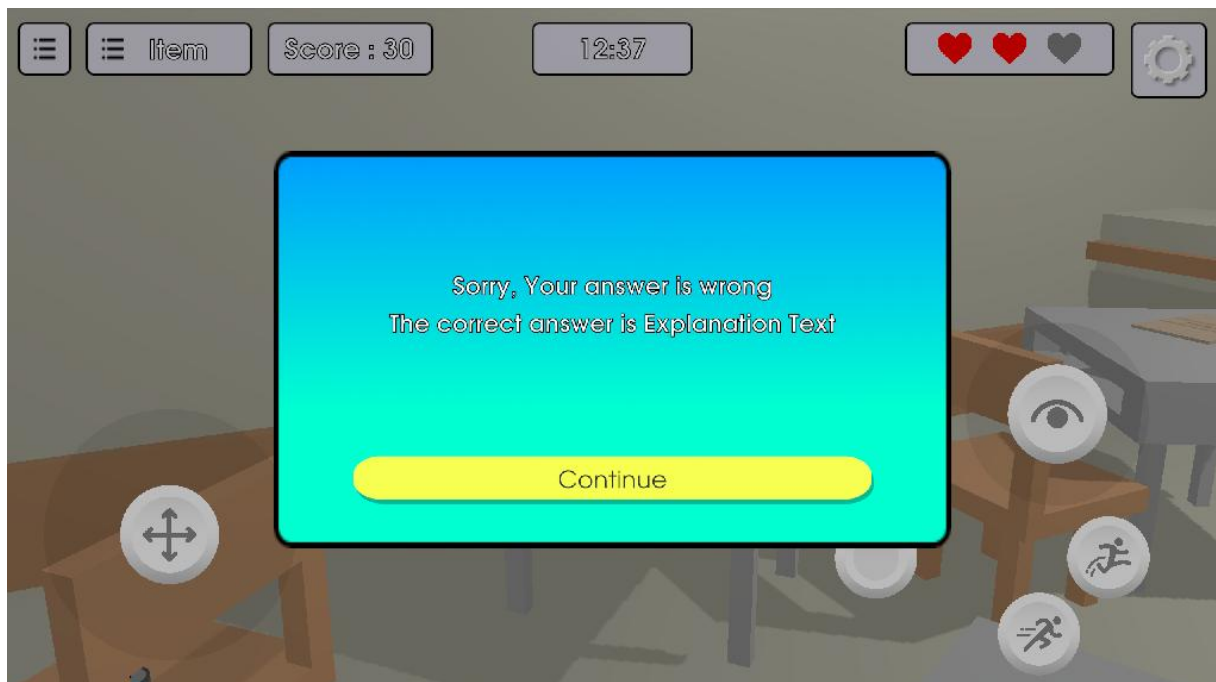
Lampiran Tampilan Soal



Lampiran Monolog Pada Soal



Lampiran Menjawab Soal Benar



Lampiran Menjawab Soal Salah



Lampiran Setelah Menjawab Semua Soal

LAMPIRAN D GAME DESIGN DOCUMENT (GDD)

GDD

KURO GAME

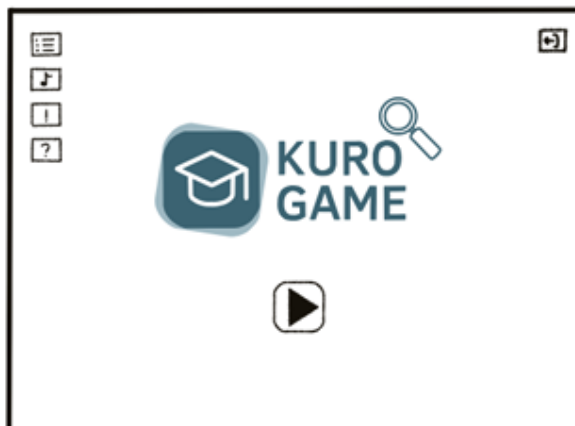
Final Version

Pada setiap item yang terdapat pada ruangan bermain level 2 ini dilengkapi dengan sketsa rancangan collider yang berfungsi untuk pemantik agar pemain dapat mengklik item yang ditersembunyi.

Gameplay



Tampilan ketika pertama kali pemain membuka game. Pada bagian halaman utama ini terdapat tombol play, tombol ikon option di sisi kiri atas, dan tombol ikon keluar di sisi kanan layar.

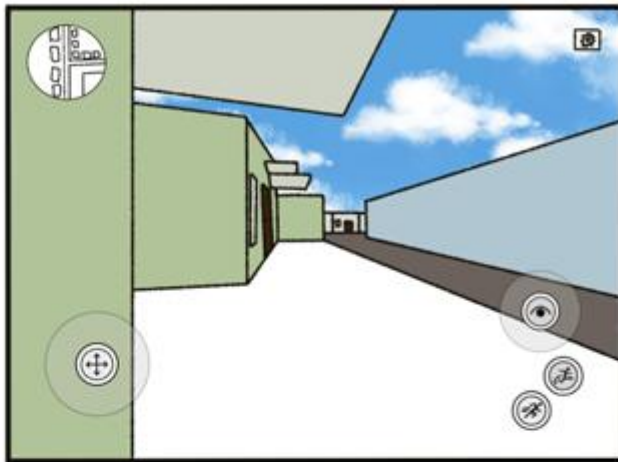


Sketsa tampilan menu utama

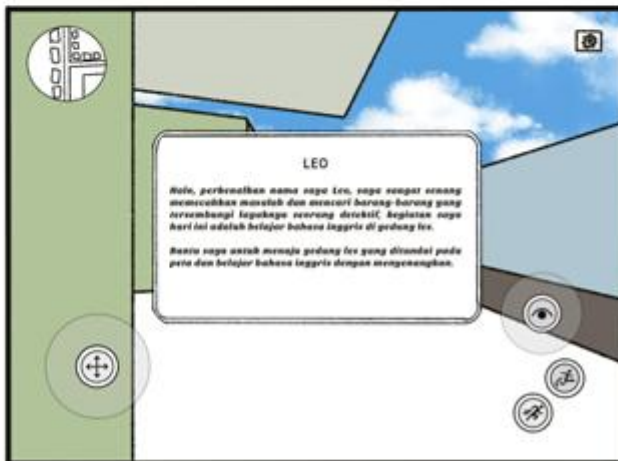
GDD

KURO GAME

Final Version



Gameplay dimulai ketika pemain berperan sebagai seorang karakter Leo yang berada dirumahnya sedang Bersiap-siap untuk berangkat les Bahasa Inggris, pemain menerima notifikasi mengenai penjelasan seorang karakter Leo dan pemain menerima titik lokasi gedung les yang harus dituju oleh pemain, pemain menyusuri kota untuk menuju gedung les. Pada tampilan luar ruangan akan terdapat sebuah peta kecil dibagian kiri atas layar dan terdapat tombol pengaturan dibagian kanan atas.



Berikut adalah interaksi dari karakter utama Bernama Leo yang ingin memperkenalkan diri pada orang yang memainkan dirinya.

Sketsa Saat Permainan Pertama Kali Dimulai

GDD

KURO GAME

Final Version

ruangan akan terdapat sebuah peta kecil dibagian kiri atas layar dan terdapat tombol pengaturan dibagian kanan atas.



Sebelum pemain memasuki gedung, pemain akan bertemu dengan seorang satpam yang menjaga gedung les. NPC satpam dapat berinteraksi dengan pemain dengan menyapa pemain dan mempersilahkan memasuki gedung les. Pada tampilan luar ruangan akan terdapat sebuah peta kecil dibagian kiri atas layar dan terdapat tombol pengaturan dibagian kanan atas.



Sketsa Interaksi Pemain dengan NPC Satpam



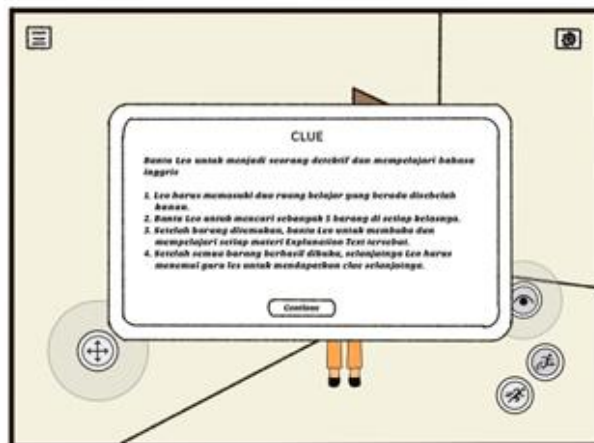
Setelah memasuki gedung, pemain akan melihat seorang guru les yang akan menyambutnya. Pada tampilan dalam ruangan akan terdapat tombol disisi kanan atas layar yang akan memuat informasi clue yang diberikan oleh guru les dan terdapat tombol pengaturan dibagian kanan atas layar.

Sketsa Interaksi Pemain dengan NPC Guru Les

GDD

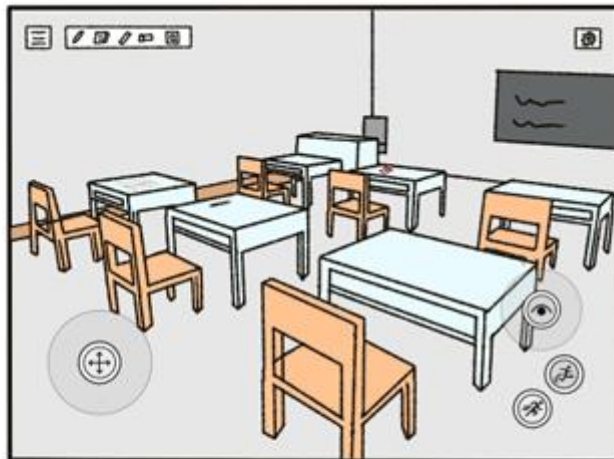
KURO GAME

Final Version



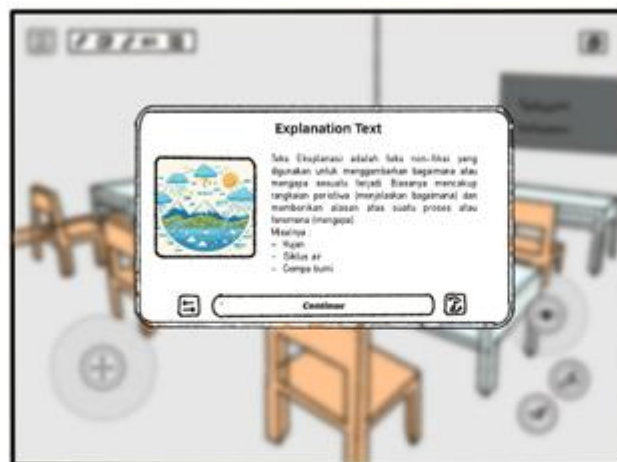
Setelah clue diberikan dan setelah pemain membaca dengan teliti clue tersebut, pemain harus memasuki dua ruang belajar yang berada di sebelah kanan pemain. Pada tampilan dalam ruangan akan terdapat tombol disisi kanan atas layar yang telah memuat informasi clue yang diberikan oleh guru les dan terdapat tombol pengaturan dibagian kanan atas layar.

Sketsa Guru Memberi Clue untuk Pemain



Pemain memasuki ruang belajar satu persatu dan sesuai clue yang telah diberikan, pemain harus mencari barang-barang yang dimana didalamnya berisi materi pembelajaran explanation text. Pada tampilan dalam ruangan akan terdapat tombol disisi

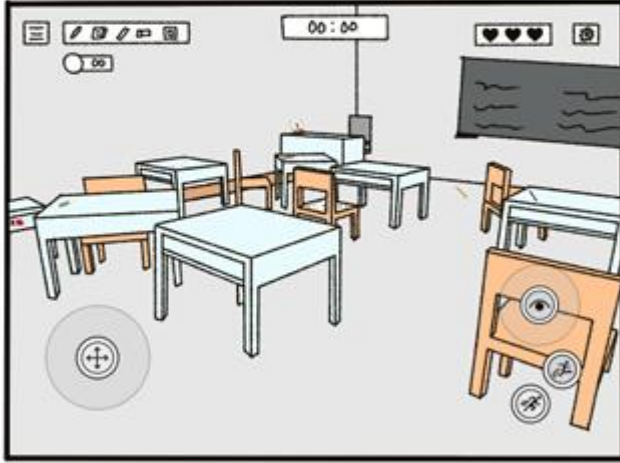
Sketsa Ruang Materi



Setelah pemain berhasil menemukan salah satu barang lalu membukanya maka akan muncul popup. pada popup ini akan menampilkan materi pembelajaran serta terdapat tombol ubah Bahasa, tombol mengulang suara, dan tombol lanjutkan untuk menutup popup dan melanjutkan pencarian. Pemain yang telah berhasil menemukan dan membuka semua materi yang terdapat di kedua ruang belajar tersebut maka pemain harus bertemu dengan guru les yang berada diluar ruangan.

Sketsa Tampilan Materi

Berdasarkan clue tersebut pemain harus memasuki ruang bermain yang berada di sebelah kiri pemain. Pada popup ini menampilkan clue yang diberikan oleh guru dan terdapat tombol kembali untuk keluar dari popup clue.



Sketsa Ruang Bermain / Ruang yang Berisi Soal

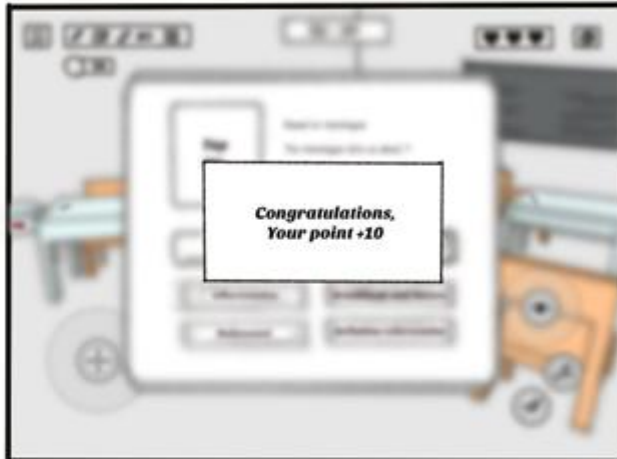
Pemain memasuki ruang bermain yang dimana Susana ruang kelas tersebut berantakan, pemain harus mencari barang-barang yang penempatannya teracak. Pada tampilan ruang bermain ini terdapat tombol untuk membuka clue disebelah kiri atas layar, disampingnya terdapat progress benda yang telah ditemukan pemain, dibawahnya terdapat jumlah poin yang telah didapat, ikon tiga hati sebagai tiga kali kesempatan untuk pemain yang berada disebelah kanan atas, serta tombol pengaturan.



Pemain harus mencari dan menjawab soal dengan benar dalam estimasi waktu sebanyak 15 menit dengan jenis soal pilihan ganda. Pada popup soal ini terdapat soal,

Sketsa Pemain Tampilan Soal

gambar soal, 4 pilihan kata untuk disusun, tombol x untuk menghapus jawaban dan tombol ceklis untuk mengecek jawaban.



Setelah pemain menjawab soal dengan benar, maka pemain akan mendapatkan nilai sesuai dengan berapa kali pemain berusaha menjawab pertanyaan. Popup ini akan menampilkan nilai yang didapat setelah menjawab 1 soal.



Setelah semua pertanyaan terjawab dengan benar, pemain akan mendapatkan total nilai dari keseluruhan soal yang telah dijawab oleh pemain dan pemain akan

Sketsa Pemain Menjawab Soal Benar

mendapatkan bintang sebagai tingkatan nilai. Pada popup ini menampilkan hasil akumulasi nilai yang berhasil didapat pada level 1 dan nilai tersebut direpresentasikan oleh ketiga Bintang yang menyala.



Pemain akan mendapatkan kartu ucapan bagi pemain yang berhasil mencapai nilai maksimal, dan bagi pemain yang tidak mencapai nilai maksimal akan mendapatkan kartu ucapan penyemangat. Pada popup ini akan menampilkan kartu ucapan beserta nama pemain atas pencapaiannya dan terdapat tombol kembali untuk kembali ke tampilan selanjutnya.

Sketsa Pemain telah Menyelesaikan Soal

LAMPIRAN E QUISIONER

Nama : _____ Jenis Kelamin : _____
Kelas : _____ Usia : _____

Harap isi quisioner ini dengan bersungguh-sungguh, quisioner ini akan menjadi acuan dalam pengembangan sebuah game

1. Suka belajar Bahasa Inggris?

☐ Ya☐ Tidak☐ Alasan lain : _____

2. Apa saja yang kalian sukai dalam belajar Bahasa Inggris di tingkat SMA?

(contoh : belajar di kelas, tugas di kelas, PR, dll)

3. Apa saja yang kalian tidak suka dalam belajar Bahasa Inggris di tingkat SMA?

(contoh : belajar di kelas, tugas di kelas, PR, dll)

4. Seberapa banyak tugas kalian di kelas 11? (Pilih salah satu)

☐ Tidak ada,☐ Sedikit,☐ Banyak,☐ Banyak Sekali

5. Bagaimana tanggapannya ketika diberi tugas? (boleh pilih lebih dari satu)

☐ Suka☐ Bersemangat☐ Malas☐ Ngantuk☐ Bosan☐ Alasan lain : _____

6. Apakah kalian suka bermain game?

☐ Suka☐ Kurang Suka☐ Tidak Suka

Quisioner Siswa

7. Game apa yang sering dimainkan?

8. Game seperti apa yang disukai? (Contoh : petualangan, tantangan, dll)

9. Perangkat yang sering digunakan untuk bermain game

☐ Komputer

☐ Smartphone

10. Jenis ponsel yang digunakan

☐ Android

☐ IOS (Apple OS)

☐ Handphone jadul

☐ Microsoft Windows Mobile

11. Tipe ponsel yang digunakan (Contoh : Poco X3, Infinix Zero 5G)

Quisioner Siswa

Nama : _____ Jenis Kelamin : _____
 Kelas : _____ Usia : _____

Harap isi quisioner ini dengan bersungguh-sungguh, quisioner ini akan menjadi acuan dalam pengembangan sebuah game

1. Seberapa sering Anda menggunakan perangkat teknologi, seperti komputer atau ponsel cerdas, dalam kegiatan belajar Anda?
☐ Sering ☐ Jarang ☐ Tidak Pernah

2. Apakah Anda lebih suka belajar melalui permainan atau metode tradisional, seperti membaca buku atau mendengarkan ceramah?
☐ Tradisional ☐ Permainan ☐ Tidak Keduanya

3. Apa jenis materi pelajaran Bahasa Inggris yang menurut Anda paling sulit dipahami atau menarik minat Anda untuk dipelajari?

4. Dalam game edukasi, bagaimana Anda ingin melihat informasi atau konsep disampaikan kepada Anda? (Contoh: melalui teks, gambar, video, animasi)

5. Apakah ada fitur khusus yang ingin Anda lihat dalam game edukasi, seperti sistem poin, tantangan, atau tugas tambahan?
☐ Ya ☐ Tidak

6. kemampuan ekstra sebagai pemain seperti apa yang anda harapkan di dalam sebuah game? (pilih lebih dari satu)

☐ Berinteraksi dengan karakter game	☐ Fitur Kesehatan
☐ Alat khusus untuk membantu player	☐ Penambahan waktu bermain
☐ Poin tambahan	☐ Fitur lainnya :

Quisioner Siswa

7. Suasana game yang anda harapkan
- | | |
|--|--|
| ☐ Menegangkan | ☐ Menyeramkan |
| ☐ Penuh Tantangan | ☐ Lainnya : |
8. Apakah Anda lebih tertarik dengan game edukasi yang berfokus pada pemecahan masalah, keterampilan kreatif, atau pengetahuan faktual?
- | | |
|-----------------------------|--------------------------------|
| ☐ Ya | ☐ Tidak |
|-----------------------------|--------------------------------|
9. Apakah Anda lebih suka game edukasi dengan tampilan grafis yang realistis atau yang lebih berfokus pada kesederhanaan dan kebersamaan?
- | | |
|-----------------------------|--------------------------------|
| ☐ Ya | ☐ Tidak |
|-----------------------------|--------------------------------|
10. Apakah Anda lebih suka bermain game dengan banyak kontrol pemain atau game dengan kontrol sederhana?
- | | |
|--|--|
| ☐ Banyak kontrol pemain | ☐ Kontrol sederhana |
|--|--|

Quisioner Siswa

Form 1 Total 23 Orang

1. Suka belajar Bahasa Inggris?
 - Ya : 19 Orang
 - Tidak : 2 Orang
 - Alasan lain : 2 Orang
2. Apa saja yang kalian sukai dalam belajar Bahasa Inggris di tingkat SMA?
 - Berbicara (berkomunikasi – kosa kata) : 11 Orang
 - Belajar di kelas : 10 Orang
 - Pemahaman tugas : 2 Orang
3. Apa saja yang kalian tidak suka dalam belajar Bahasa Inggris di tingkat SMA?
 - Kosa kata (vocabulary - komunikasi) : 10
 - Test : 1
 - Tugas – pr : 11
 - Pemahaman materi : 1
4. Seberapa banyak tugas kalian di kelas 11?
 - Banyak : 14 Orang
 - Banyak sekali : 9 Orang
5. Bagaimana tanggapannya ketika diberi tugas? (1 orang lebih dari 1 jawaban)
 - Suka : 10
 - Bersemangat : 4
 - Ngantuk : 6
 - Malas : 4
 - Bosan : 3
 - Cape, mudah pusing : 3

$10 + 4 = 14$
 $6 + 4 + 3 + 3 = 16$
6. Apakah kalian suka bermain game?
 - Suka : 8 Orang
 - Kurang suka : 15 Orang
7. Game apa yang sering dimainkan?

Hasil Olah Data Quisioner Siswa

- Org, real drum : 1
- Ludo : 6
- Mobile legends : 8
- Game mobile genos : 1
- Buss id : 1
- Clash of clan : 1
- Free fire : 1
- Pubg : 1
- Arena of valor : 1
- Minecraft : 1
- Wow (teka-teki kata) : 1
- Talking tom gold run : 1
- Stickman : 1
- Tidak bermain game : 1
- Golput : 3

7. ✓ 8. Game seperti apa yang disukai?
9

- Perang : 2
- Tantangan : 15
- Musik : 1
- Menegangkan : 1
- Petualangan : 4
- Teka-teki : 1
- Fantasi : 1
- Merawat hewan (pou) : 1
- Golput : 1

5. ✓ 9. Perangkat yang sering digunakan untuk bermain game
5

- Smartphone : 21 Orang
- Golput : 2 Orang

6. ✓ 10. Jenis ponsel yang digunakan
76

- Android : 23 orang

Hasil Olah Data Quisioner Siswa

11. Tipe ponsel yang digunakan

- Realme c33 : 1
- Oppo a5s : 1
- Oppo a31 : 1
- Infinix note 10 : 1
- Realme narzo 30a : 1
- Samsung galaxy j7 : 1
- Samsung a 13 : 1
- Infinix smart 6 : 1
- Vivo y15s : 1
- Oppo a9 2020 : 1
- Vivo y15 : 1
- Oppo a17 : 1
- Infinix hot 9 play : 2
- Redmi 6a : 1
- Vivo y 12 : 2
- Infinix hot 11 play : 1
- Oppo 1724 : 1 ✓
- Oppo a16 : 1
- Redmi 9a : 1
- Oppo a57 : 1
- Vivo y71 : 1

Hasil Olah Data Quisioner Siswa

Form 2 Total 23 Orang

4. ✓ 1. Seberapa sering Anda menggunakan perangkat teknologi, seperti komputer atau ponsel cerdas, dalam kegiatan belajar Anda?
- Sering : 16 Orang
 - Jarang : 7 Orang
2. ✓ 2. Apakah Anda lebih suka belajar melalui permainan atau metode tradisional, seperti membaca buku atau mendengarkan ceramah?
- Permainan : 15 Orang
 - Tradisional : 6 Orang
 - Tradisional & permainan : 2 Orang
3. ✓ 3. Apa jenis materi pelajaran Bahasa Inggris yang menurut Anda paling sulit dipahami atau menarik minat Anda untuk dipelajari?
- Materi yang sulit dipahami
- Semua : 2
 - Dialog : 2
 - Grammar : 7
 - Kata kerja : 6
 - verb : 4
 - Kosa kata : 3
 - Komunikasi : 2
 - Text : 1
- Materi yang mudah dipahami
- Materi yang mudah dipelajari : 2
 - Tertarik, berguna untuk kehidupan : 1
- Tidak tahu : 1
4. ✓ 4. Dalam game edukasi, bagaimana Anda ingin melihat informasi atau konsep disampaikan kepada Anda?
- Video : 9
 - Animasi : 9
 - Suara : 2

Hasil Olah Data Quisioner Siswa

- Gambar : 5
 - Teks : 6
- ✓ 5. Apakah ada fitur khusus yang ingin Anda lihat dalam game edukasi, seperti sistem poin, tantangan, atau tugas tambahan?
- Ya : 19 Orang
 - Tidak : 4 Orang
- ✓ 6. kemampuan ekstra sebagai pemain seperti apa yang anda harapkan di dalam sebuah game?
- Berinteraksi dengan npc : 5
 - Alat khusus untuk membantu player : 7
 - Poin tambahan : 13
 - Fitur Kesehatan : 2
 - Penambahan waktu bermain : 11
 - Skin gratis : 1
7. Suasana game yang anda harapkan
- Penuh tantangan : 19
 - Menegangkan : 1
 - Banyak fitur : 1
 - Santai, tenang : 2
 - Multiplayer : 1
 - Seru : 1
- ✓ 8. Apakah Anda lebih tertarik dengan game edukasi yang berfokus pada pemecahan masalah, keterampilan kreatif, atau pengetahuan faktual?
- Ya : 22 Orang
 - Tidak : 1 Orang
- ✓ 9. Apakah Anda lebih suka game edukasi dengan tampilan grafis yang realistis atau yang lebih berfokus pada kesederhanaan dan kebersamaan?
- Ya : 20 Orang
 - Tidak : 3 Orang

Hasil Olah Data Quisioner Siswa

12. ✓ 10. Apakah Anda lebih suka bermain game dengan banyak kontrol pemain atau game dengan kontrol sederhana?

- Kontrol sederhana : 17 Orang
- Banyak kontrol : 6 Orang

Kelas : XI IPA 1

Usia

- 16 Tahun : 3 Orang
- 17 tahun : 20 Orang

Gender

- Laki-laki : 6 Orang
- Perempuan : 17 Orang

Rincian

- Laki-laki usia 17 tahun : 6 Orang
- Laki-laki usia 16 tahun : 0 Orang
- Perempuan usia 17 tahun : 14 Orang
- Perempuan usia 16 tahun : 3 Orang

Hasil Olah Data Quisioner Siswa