

**IMPLEMENTASI *AUGMENTED REALITY* PADA
PEMBUATAN *GAME MOBILE* DALAM
MEMPERKENALKAN SEJARAH KEMERDEKAAN
REPUBLIK INDONESIA**

SKRIPSI

Diajukan Sebagai Salah Satu Syarat untuk Mendapatkan
Gelar Sarjana Komputer (S.Kom) Program Studi Informatika



LA ODE TAUFIK ISMAIL

105841102718

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAKASSAR**

2022

**IMPLEMENTASI *AUGMENTED REALITY* PADA
PEMBUATAN *GAME MOBILE* DALAM
MEMPERKENALKAN SEJARAH KEMERDEKAAN
REPUBLIK INDONESIA**

**Diajukan untuk Memenuhi Salah Satu Syarat Guna Memperoleh
Gelar Sarjana Komputer Prodi Informatika Fakultas Teknik
Universitas Muhammadiyah Makassar**

Disusun Dan Diajukan Oleh:

**LA ODE TAUFIK ISMAIL
105 84 11027 18**

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MUHAMMADIYAH MAKASSAR**

2022

14/09/2022

1209
Smb. Alumni

R/0064/IFM/2200
ISM
E



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

PENGESAHAN

Skripsi atas nama La Ode Taufik Ismail dengan nomor induk Mahasiswa 105 84 11027 18, dinyatakan diterima dan disahkan oleh Panitia Ujian Tugas Akhir/Skripsi sesuai dengan Surat Keputusan Dekan Fakultas Teknik Universitas Muhammadiyah Makassar Nomor : 453/05/A.5-VI/VI/43/2022, sebagai salah satu syarat guna memperoleh gelar Sarjana Komputer pada Program Studi Informatika Teknik Universitas Muhammadiyah Makassar pada hari Selasa tanggal 30 Agustus 2022.

Panitia Ujian :

Makassar, 09 Safar 1444 H

06 September 2022 M

1. Pengawas Umum

a. Rektor Universitas Muhammadiyah Makassar

Prof. Dr. H. Ambo Asse, M.Ag

b. Dekan Fakultas Teknik Universitas Hasanuddin

Prof. Dr. Eng. Muhammad Isran Ramli, ST., MT.

2. Penguji

a. Ketua : Dr. Ir. Zahir Zainuddin, M.Sc

b. Sekretaris : Chyquitha Danuputri, S.Kom., M.Kom

3. Anggota

1. Muhyiddin A. M. Hayat, S.Kom., MT.

2. Lukman Anas S.Kom., MT.

3. Titin Wahyuni, S.Pd., MT.

Mengetahui :

Pembimbing I

Fahrim Irhamna Rahman S.Kom., MT.

Pembimbing II

Rizki Yusliana Bakti ST.,MT.

Dekan



Dr. Ir. Hj. Nurnawaty, ST., MT., IPM.

NBM : 795 108



بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

HALAMAN PENGESAHAN

Tugas Akhir ini diajukan untuk memenuhi syarat ujian guna memperoleh gelar Sarjana Komputer (S.Kom) Program Studi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar.

Judul Skripsi : **IMPLEMENTASI AUGMENTED REALITY PADA PEMBUATAN GAME MOBILE DALAM MEMPERKENALKAN SEJARAH KEMERDEKAAN REPUBLIK INDONESIA**

Nama : La Ode Taufik Ismail

Stambuk : 105 84 11027 18.

Makassar, 6 September 2022

Telah Diperiksa dan Disetujui
Oleh Dosen Pembimbing;

Pembimbing I

Pembimbing II

Fahrir Irhamna Rahman, S.Kom.MT.

Rizki Yusliana Bakti ST.,MT.

Mengetahui,

Ketua Program Studi Informatika

Muhyiddin A M Hayat, S.Kom.,MT.

NBM : -

ABSTRAK

LA ODE TAUFIK ISMAIL. Implementasi *Augmented Reality* Pada Pembuatan *Game Mobile* Dalam Memperkenalkan Sejarah Kemerdekaan Republik Indonesia (dibimbing oleh Fahrir Irhamna Rahman S.Kom., MT. dan Rizki Yusliana Bakti ST.,MT.)

Penelitian ini bertujuan agar menghasilkan aplikasi *game* yang menggunakan teknologi *augmented reality* sebagai media untuk menarik minat masyarakat khususnya pelajar dalam mengenalkan sejarah perjuangan kemerdekaan khususnya peristiwa 10 november 1945.

Dalam pembuatan *game* ini melalui beberapa tahap, antara lain pembuatan *usecase diagram*, *activity diagram*, *sequence diagram*, perancangan komponen permainan dan pengujian aplikasi yang dimana pengujian aplikasi ini terdiri dari pengujian *blackbox*, pengujian intensitas cahaya dan pengujian respon siswa terhadap *game* edukasi sejarah ini. Di dalam *game* ini terdapat *tank* sebagai pemain serta *turret* dan *tank*-musuh sebagai lawan dari pemain dan juga terdapat *cutscene intro* dan *outro* yang digunakan sebagai media yang akan memberikan informasi pembelajaran didalam *game* ini.

Kesimpulan yang didapatkan dari hasil pengujian aplikasi *game* ini yaitu setiap komponen *game* berjalan dengan baik dan berfungsi sebagaimana harapan *developer* dengan syarat *game* dapat dimainkan ketika intensitas cahaya berada diatas 51 *lux* keatas dan hasil respon para siswa dan siswi menikmati serta menanggapi bahwasanya *game augmented reality* ini dianggap menarik oleh mereka dalam mempelajari sejarah perjuangan kemerdekaan republik Indonesia sehingga mereka ingin kembali bermain dan mencoba jenis *game* edukasi yang serupa dan sejenis.

Kata Kunci : *Augmented Reality*; Edukasi; *Game*; Kemerdekaan; Sejarah

ABSTRACT

LA ODE TAUFIK ISMAIL. *Implementation of Augmented Reality in Mobile Game Development in Introducing the History of the Independence of the Republic of Indonesia (supervised by Fahrim Irahma Rahman S.Kom., MT. and Rizki Yusliana Bakti ST.,MT.)*

This study aims to produce a game application that uses augmented reality technology as a medium to attract public interest, especially students, in introducing the history of the struggle for independence, especially the events of 10 November 1945.

In making this game through several stages, including making use case diagrams, activity diagrams, sequence diagrams, designing game components and testing applications where testing this application consists of blackbox testing, light intensity testing and testing student responses to this historical educational game. In this game there are tanks as players as well as turrets and enemy tanks as opposed to players and there are also intro and outro cutscenes that are used as media that will provide learning information in this game.

The conclusion obtained from the results of testing this game application is that each game component runs well and functions as the developer hopes with the condition that the game can be played when the light intensity is above 51 lux and above and the results of the responses of students enjoying and responding that this augmented reality game is considered interesting by them in studying the history of the struggle for independence of the republic of Indonesia so that they want to return to play and try similar and similar types of educational games.

Keywords: *Augmented Reality; Education; Game; Independence; History*

KATA PENGANTAR

Segala Puji syukur kehadirat Allah SWT atas segala rahmat dan karunia-Nya. Dan tidak lupa shalawat serta salam kepada Rasulullah Muhammad SAW, keluarga, sahabat dan para pengikutnya hingga akhir zaman nanti sehingga penulis dapat menyelesaikan skripsi dengan judul "**IMPLEMENTASI AUGMENTED REALITY PADA PEMBUATAN GAME MOBILE DALAM MEMPERKENALKAN SEJARAH KEMERDEKAAN REPUBLIK INDONESIA**" sebagaimana mestinya.

Tugas akhir ini tidak akan terwujud tanpa adanya bantuan, arahan, bimbingan dan dorongan dari berbagai pihak. Maka dari itu penghargaan yang setinggi-tingginya dan terimakasih banyak kami haturkan dengan hormat kepada:

1. Kedua orang tua kami tercinta, penulis mengucapkan terima kasih yang sebesar-besarnya atas segala limpahan kasih sayang, Doa dan dukungan baik secara moral maupun materi.
2. Bapak Prof. Dr. H. Ambo Asse, Mag. Sebagai Rektor Universitas Muhammadiyah Makassar.
3. Ibu Dr. Ir. Hj. Nurnawaty, ST., MT., IPM. Sebagai Dekan Fakultas Teknik Universitas Muhammadiyah Makassar.
4. Bapak Muhyiddin AM Hayat, S.Kom, M.T. Sebagai Ketua Prodi Informatika, Fakultas Teknik Universitas Muhammadiyah Makassar.
5. Fahrin Irhamna Rahman S.Kom.,MT. Selaku Pembimbing I dan Ibu Rizki Yusliana Bakti, S.T., MT. Selaku Pembimbing II yang senantiasa meluangkan waktunya membimbing dan mengarahkan penulis dalam penyusunan tugas Skripsi ini.
6. Bapak/Ibu Dosen dan Staff Administrasi Prodi Informatika Fakultas Teknik Universitas Muhammadiyah Makassar.
7. Saudara/saudari kami di Fakultas Teknik, Mekanika 2018 yang selalu belajar dan berjuang bersama dengan rasa persaudaraan yang tinggi banyak membantu serta memberi dukungan dalam menyelesaikan tugas akhir ini. Akhir kata,

penulis mengharapkan tugas akhir ini dapat bermanfaat bagi pengembangan ilmu pengetahuan khususnya dibidang Keinformatikaan. Aamiin.

Dalam penulisan skripsi ini peneliti menyadari bahwa penulisan skripsi ini masih belum mencapai kesempurnaan baik dari segi materi maupun dari segi penyajian, untuk itu peneliti mengharapkan kritik dan saran untuk membangun. Semoga dengan adanya skripsi ini dapat berguna bagi pembaca dan memberikan manfaat bagi pihak yang membutuhkannya.

Semoga segala bantuan yang telah diberikan kepada penulisakan mendapatkan balasan yang setimpal dan Allah SWT. Aamiin

"Billahi Fii Sabilil Haq Fastabiquil Khaerat"

Makassar, 09 Semptember 2022

Penulis

DAFTAR ISI

ABSTRAK.....	v
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xi
DAFTAR TABEL.....	xiii
DAFTAR LAMPIRAN.....	xiv
DAFTAR ISTILAH.....	xv
BAB I PENDAHULUAN.....	1
A. Latar Belakang Masalah.....	1
B. Rumusan Masalah.....	2
C. Tujuan Penelitian.....	2
D. Manfaat Penelitian.....	2
E. Ruang Lingkup Penelitian.....	3
F. Sistematika Penulisan.....	3
BAB II TINJAUAN PUSTAKA.....	4
A. Landasan Teori.....	4
1. Sejarah.....	4
2. <i>Game</i>	7
3. <i>Augmented Reality</i>	11
4. Perangkat Pengembang.....	12
B. Penelitian Terkait.....	17
C. Kerangka Pikir.....	19
BAB III METODE PENELITIAN.....	21
A. Tempat dan Waktu Penelitian.....	21

B. Alat dan Bahan	21
C. Perancangan Sistem	21
1. Rancangan Penelitian	22
2. Konsep Pembangunan <i>Game</i>	23
3. <i>Design Treatment</i>	24
4. <i>Storyline</i>	24
5. Analisa Kebutuhan Fungsional	25
6. Analisa Kebutuhan Non-Fungsional	31
D. Teknik Pengujian Sistem	33
E. Teknik Analisis Data.....	34
BAB IV HASIL DAN PEMBAHASAN	36
A. Hasil	36
1. <i>Sequence Diagram</i>	36
2. <i>Game Rules</i>	38
3. <i>Storyboard</i>	39
4. Perancangan Komponen Permainan.....	43
5. Perancangan Antarmuka.....	52
6. Perancangan <i>CutScene</i>	55
B. Pengujian.....	61
1. Pengujian Aplikasi	61
BAB V KESIMPULAN DAN SARAN	78
A. Kesimpulan	78
B. Saran.....	78
DAFTAR PUSTAKA	80
LAMPIRAN	83

DAFTAR GAMBAR

Gambar 1. Simbol <i>Use Case Diagram</i>	15
Gambar 2. Simbol <i>Activity Diagram</i>	16
Gambar 3. Simbol <i>Sequence Diagram</i>	17
Gambar 4. Kerangka Pikir.....	19
Gambar 5. Rancangan Penelitian.....	22
Gambar 6. <i>Use Case Diagram</i>	26
Gambar 7. <i>Activity Diagram</i> Menu Mulai.....	27
Gambar 8. <i>Activity Diagram</i> Menu Keluar.....	28
Gambar 9. <i>Activity Diagram</i> Ketika <i>Game</i> Dimulai.....	29
Gambar 10. <i>Activity Diagram</i> Ketika Pemilihan <i>Game</i>	30
Gambar 11. <i>Sequence Diagram</i> Dalam <i>Game</i>	36
Gambar 12. <i>Sequence Diagram</i> Pemilihan <i>Game</i>	38
Gambar 13. <i>Scene</i> Menu Utama.....	40
Gambar 14. <i>Scene</i> Pemilihan Tingkat Kesulitan.....	40
Gambar 15. <i>Scene</i> Menampilkan <i>CutScene Intro</i>	41
Gambar 16. <i>Scene Scan</i> Lingkungan.....	41
Gambar 17. <i>Scene</i> Utama <i>Game</i>	42
Gambar 18. <i>Scene</i> Menang.....	42
Gambar 19. <i>Scene</i> Menampilkan <i>CutScene Outro</i>	43
Gambar 20. <i>Scene</i> Kalah.....	43
Gambar 21. Desain <i>Game</i> -Pemain.....	44
Gambar 22. Desain <i>Game</i> - <i>Enemy</i>	44
Gambar 23. Desain <i>Turret</i>	45
Gambar 24. Gambar <i>Shell</i>	45
Gambar 25. Desain <i>JoyStick</i>	46
Gambar 26. Desain <i>Joybutton</i>	46
Gambar 27. Desain UI HP.....	47
Gambar 28. Desain HP UI Ketika HP Penuh.....	47

Gambar 29. Desain HP UI Ketika HP Berkurang.....	48
Gambar 30. Letak Jatuh Awal Rudal.....	48
Gambar 31. Letak Jatuh Akhir Rudal.....	49
Gambar 32. Antarmuka Menu Utama.....	53
Gambar 33. Antarmuka Menu Pemilihan Tingkat Kesulitan.....	53
Gambar 34. Antarmuka Pemilihan <i>Tank</i>	54
Gambar 35. Antarmuka Dalam <i>Game</i>	54
Gambar 36. Antarmuka Menang.....	55
Gambar 37. Antarmuka Kalah.....	55
Gambar 38. <i>CutScene Intro I</i>	56
Gambar 39. <i>CutScene Intro II</i>	56
Gambar 40. <i>CutScene Intro III</i>	57
Gambar 41. <i>CutScene Intro IV</i>	57
Gambar 42. <i>CutScene Intro V</i>	57
Gambar 43. <i>CutScene Intro VI</i>	58
Gambar 44. <i>CutScene Intro VII</i>	58
Gambar 45. <i>CutScene Intro VIII</i>	59
Gambar 46. <i>CutScene Outro I</i>	59
Gambar 47. <i>CutScene Outro II</i>	60
Gambar 48. <i>CutScene Outro III</i>	60
Gambar 49. <i>CutScene Outro IV</i>	61
Gambar 50. <i>CutScene Outro V</i>	61
Gambar 51. Hasil Responden Dari Pertanyaan I.....	71
Gambar 52. Hasil Responden Dari Pertanyaan II.....	72
Gambar 53. Hasil Responden Dari Pertanyaan III.....	73
Gambar 54. Hasil Responden Dari Pertanyaan IV.....	74
Gambar 55. Hasil Responden Dari Pertanyaan V.....	75
Gambar 56. Hasil Responden Dari Pertanyaan VI.....	76
Gambar 57. Hasil Responden Dari Pertanyaan VII.....	77

DAFTAR TABEL

Tabel 1. Perangkat Laptop <i>Developer</i>	31
Tabel 2. Perangkat <i>Smartphone Developer</i>	32
Tabel 3. Perangkat <i>Smartphone User</i>	32
Tabel 4. Perangkat Pengujian I Untuk Intensitas Cahaya.....	32
Tabel 5. Perangkat Pengujian II Untuk Intensitas Cahaya.....	33
Tabel 6. Rancangan Tabel Pengujian <i>BlackBox</i>	34
Tabel 7. Rincian Tingkat Kesulitan.....	38
Tabel 8. Pengujian Komponen Mulai.....	62
Tabel 9. Pengujian Komponen Keluar.....	62
Tabel 10. Pengujian Komponen Tingkat Kesulitan.....	63
Tabel 11. Pengujian Komponen Scan Lingkungan.....	64
Tabel 12. Pengujian Komponen <i>CutScene</i>	65
Tabel 13. Pengujian Komponen Pemilihan <i>Tank</i>	66
Tabel 14. Pengujian Komponen Bergerak.....	66
Tabel 15. Pengujian Komponen Menembak.....	67
Tabel 16. Tabel Pengujian Intensitas Cahaya <i>Indoor Redmi Note 10 5G</i>	67
Tabel 17. Tabel Pengujian Intensitas Cahaya <i>Outdoor Redmi Note 10 5G</i>	68
Tabel 18. Tabel Pengujian Intensitas Cahaya <i>Indoor Poco M3 Pro 5G</i>	69
Tabel 19. Tabel Pengujian Intensitas Cahaya <i>Outdoor Poco M3 Pro 5G</i>	70
Tabel 20. Hasil Responden Dari Pertanyaan I.....	71
Tabel 21. Hasil Responden Dari Pertanyaan II.....	72
Tabel 22. Hasil Responden Dari Pertanyaan III.....	72
Tabel 23. Hasil Responden Dari Pertanyaan IV.....	73
Tabel 24. Hasil Responden Dari Pertanyaan V.....	74
Tabel 25. Hasil Responden Dari Pertanyaan VI.....	75
Tabel 26. Hasil Responden Dari Pertanyaan VII.....	76

DAFTAR LAMPIRAN

Lampiran 1 : Daftar Responden Angket	83
Lampiran 2 : <i>Source Code</i>	85
Lampiran 3 : Hasil Pengujian Aplikasi	102
Lampiran 4 : Hasil Pengujian Intensitas Cahaya <i>Indoor Redmi Note 10 5G</i>	104
Lampiran 5 : Pengujian Intensitas Cahaya <i>Outdoor Redmi Note 10 5G</i>	106
Lampiran 6 : Pengujian Intensitas Cahaya <i>Indoor Poco M3 Pro 5G</i>	108
Lampiran 7 : Pengujian Intensitas Cahaya <i>Outdoor Poco M3 Pro 5G</i>	110
Lampiran 8 : Surat Keterangan Selesai Meneliti	112



DAFTAR ISTILAH



2D	Suatu bidang yang memiliki panjang dan lebar.
3D	Suatu bidang yang memiliki panjang dan lebar dan volume (isi).
Adobe Photoshop	Adalah sebuah aplikasi pengedit gambar.
Animasi	Perubahan cepat dari gambar yang berurutan untuk menciptakan suatu ilusi gerakan.
AR	Singkatan dari <i>Augmented Reality</i> yaitu realitas tabahan dengan pengabungan benda virtual kedalam dunia nyata.
Background	Merupakan bidang atau celah yang terlihat paling jauh dari sebuah objek.
Basic Goal	Deskripsi objek permainan. Atau lebih tepatnya, apa yang perlu di lakukan untuk memenangkan permainan.
Basic Mechanic	Mekanik dasar dari <i>Game</i> yang mengatur dan memandu Tindakan pemain serta respon terhadap <i>Game</i> .
Blender 3D	Adalah aplikasi pembuatan dan pemodelan 3D.
Character Development	Pengembangan karakter yang diiringi seraha dengan jalan cerita suatu <i>Game</i> .
Compositing	Proses penggabungan objek 3D dalam belender.

<i>CutScene</i>	Adegan sisipan yang berada pada awal <i>Game</i> Ketika dimainkan.
<i>CutScene Intro</i>	Adegan sisipan yang berada pada awal <i>Game</i> Ketika dimainkan.
<i>CutScene Outro</i>	Adegan sisipan yang berada pada akhir <i>Game</i> atau setelah basic goal <i>Game</i> tercapai.
<i>Collider</i>	Merupakan komponen bagian dari <i>physics</i> yang dibutuhkan oleh <i>Gameobject</i> untuk memberikan fungsi tertentu ketika <i>objek</i> tersebut menerima <i>action</i> baik dari <i>User</i> maupun dari objek lainnya yang berada dalam <i>Scene</i> .
<i>Developer</i>	Seseorang yang bertugas untuk mewujudkan desain sebuah produk atau layanan, biasanya berupa software atau aplikasi.
<i>Enemies</i>	Musuh dari pemain.
<i>Easy</i>	Mudah.
<i>Field</i>	Suatu bidang yang yang luas, rata, dan terbuka.
<i>Field Game</i>	Suatu bidang yang terdapat pada <i>Game</i> yang menjadi titik berdiri dari objek <i>Game</i> .
<i>Game</i>	Permainan.
<i>Game manager</i>	Objek dalam <i>Game</i> yang mengatur jalannya <i>Game</i> .
<i>Game Object</i>	Objek dalam <i>Game</i> .

<i>Game Puzzle</i>	Permainan teka-teki
<i>Game Rule</i>	Aturan yang ada pada <i>Game</i> .
<i>Game Simulasi</i>	Permainan simulasi.
<i>Gameplay</i>	Bagaimana pemain berinteraksi dengan <i>Game</i> atau permainan
<i>Genre</i>	Jenis atau kategori dari sesuatu film atau <i>Game</i> .
<i>Hard</i>	Sulit.
<i>HP</i>	Singkatan dari Health Point yang merupakan status darah atau kesehatan dari karakter <i>Game</i> yang sedang dimainkan.
<i>Intellectual Playbility</i>	Kelincahan Intelektual.
<i>Joybutton</i>	Sebuah tombol yang ada pada <i>Game</i> yang bisa berupa UI atau tombol fisik.
<i>JoyStick</i>	Sebuah tuas kendali yang ada pada <i>Game</i> yang bisa berupa UI atau tombol fisik yang berfungsi untuk menggerakkan karakter atau objek dalam <i>Game</i> .
<i>Lux</i>	didefinisikan sebagai satuan metrik ukuran cahaya pada suatu permukaan. Lux meter memiliki range intensitas cahaya antara 1 – 100.000 Lux.
<i>Marker</i>	Penanda.
<i>Markerless</i>	Tanpa Penanda.

Modeling	Proses mengembangkan representasi matematis dari setiap permukaan suatu objek ke dalam ruang tiga dimensi melalui perangkat lunak khusus.
Platform	Tempat untuk menjalankan perangkat lunak.
Platform Game	Tempat untuk menjalankan aplikasi <i>Game</i> .
Player	Pemain, orang yang memainkan <i>Game</i> .
Play	Bermain, mulai main.
Player Interaction	Interaksi dari pemain dalam memainkan <i>Game</i> .
Plot	Urutan beberapa peristiwa, dan setiap peristiwa tersebut memengaruhi peristiwa berikutnya melalui prinsip sebab-akibat.
Rendering	proses menghasilkan citra dari sebuah model 2 dimensi atau model 3 dimensi, melalui program computer.
Rigging	Animasi kerangka atau tali-temali adalah teknik dalam animasi computer.
Rigidbody	Sebuah fitur pada <i>Unity 3D</i> yang mampu memberikan efek gravitasi pada suatu objek atau suatu tools yang menawarkan dampak gravitasi untuk <i>character Game</i> kita. Sehingga <i>character</i> tidak hanya akan mengambang ditempatnya.

RPG	<i>Role Play Game</i> , yaitu permainan yang dimana pemain berperan sebagai tokoh dalam <i>Game</i> tersebut.
Scene	Adegan dari jalan cerita atau jalanya <i>Game</i> .
Scan	Memindai
Shell	Rudal, atau peluru yang akan ditembakkan.
Simulasi	Proses peniruan dari sesuatu yang nyata beserta keadaan sekelilingnya.
Single Player	Permainan yang dimainkan dengan 1 pemain saja.
Sports Games	Permainan Olahraga.
Storyline	Rangkaian peristiwa yang direka dan dijalin sedemikian rupa sehingga menggerakkan jalan cerita.
Target Player	Target pemain yang akan memakai aplikasi <i>Game</i> .
Test case	Kasus yang ingin diujikan.
Turret	senjata yang biasanya berupa alat penopang senjata yang mampu berputar, sehingga memungkinkan senjata ditembakkan ke segala arah
UI	Tampilan visual sebuah produk yang menjembatani sistem dengan pengguna.
Unity	Aplikasi pembuatan <i>Game</i> /permainan

Visual Studio

Merupakan salah satu *code editor* untuk mengembangkan aplikasi.

Quit

Keluar.



BAB I PENDAHULUAN

A. Latar Belakang Masalah

Indonesia adalah negara multikultural dengan sejarah panjang dan budaya yang kaya. Pelajaran sejarah terhadap sebagian orang dapat sangat membosankan. Kita mengingat nama dan kapan peristiwa itu terjadi dan belajar melalui buku. Hal ini membuat generasi mudah bosan mempelajari sejarah. Belajar melalui bermain merupakan terobosan dalam sistem pembelajaran. Sistem *experiential learning* dapat digunakan untuk pelatihan bagi para pelajar.

Game merupakan solusi media interaktif untuk belajar sambil bermain. Salah satu elemen kunci dari permainan ini adalah imersif. Hal ini memungkinkan pemain untuk merasa betah dalam permainan dan tidak peduli dengan lingkungan sekitar mereka. Sifat imersif juga dapat memengaruhi penerimaan pemain terhadap konten dan pesan *Game*. Immersiveness ini juga memungkinkan pemain agar seolah-olah mereka adalah bagian dari dalam *Game* tersebut.

Augmented Reality (AR) merupakan sebuah teknologi yang menggabungkan sebuah objek virtual 2D atau 3D dalam lingkungan nyata dan memproyeksikannya ke dalam objek virtual secara real time. *Augmented Reality* dapat diterapkan pada semua indera, termasuk pendengaran, sentuhan, dan penciuman. *Augmented Reality* sudah digunakan di berbagai bidang seperti pendidikan, militer, kesehatan dan manufaktur. *Augmented Reality* ini memungkinkan kita untuk memasukkan sebuah informasi tertentu ke dunia maya dan menampilkannya di dunia nyata menggunakan perangkat seperti *webcam*, komputer, *Smartphone Android*, dan kacamata khusus (Rizqi Mauludin, 2017).

Game edukasi terutama edukasi yang memperkenalkan sejarah sudah ada sebelumnya tapi belum ada yang mengimplementasikan *Augmented Reality* kedalam *Gamenya*. contohnya, yaitu *Game* "P10NER" yang menceritakan tentang perjuangan kemerdekaan pada pertempuran 10 November 1945 yang dibuat oleh mahasiswa Teknik Informatika Institut Teknologi Sepuluh November Surabaya (ITS). Kekurangan *Game* ini ialah untuk memainkannya pengguna atau pemain

harus menggunakan PC (*personal computer*) dikarenakan tidak semua orang mempunyai PC (Tempo.co, 2010).

Tidak banyak *Game* edukasi yang tersedia saat ini yang mencakup sejarah kemerdekaan Republik Indonesia. Isu-isu tersebut memunculkan konsep pemanfaatan kemajuan teknologi informasi, khususnya teknologi *Augmented Reality*. Tujuan penggunaannya adalah untuk membuat aplikasi *mobile gaming* berbasis *Augmented Reality* dan menyajikan kisah perjuangan kemerdekaan Republik Indonesia. Agar pelajar tertarik dalam memainkan *Game* ini, *Game* ini telah dibuat semenarik mungkin. Pengguna *Game* ini dapat belajar tentang sejarah perjuangan kemerdekaan Republik Indonesia sambil bersenang-senang dan bersenang-senang bermain *Game*.

B. Rumusan Masalah

Berdasarkan dari latar belakang masalah diatas, dapat diambil rumusan masalah yaitu:

1. Sudah banyaknya *Game* edukasi sejarah yang dibuat namun belum ada yang menggunakan teknologi *Augmented Reality* sebagai media untuk menarik minat masyarakat khususnya pelajar dalam mengenalkan sejarah perjuangan kemerdekaan khususnya peristiwa 10 November 1945.

C. Tujuan Penelitian

Penelitian ini bertujuan agar menghasilkan aplikasi *Game* yang menggunakan teknologi *Augmented Reality* sebagai media untuk menarik minat masyarakat khususnya pelajar dalam mengenalkan sejarah perjuangan kemerdekaan khususnya peristiwa 10 November 1945.

D. Manfaat Penelitian

Penulis penelitian ini percaya bahwa hasil penelitian ini akan bermanfaat tidak hanya untuk penulis sendiri tetapi juga untuk masyarakat luas, khususnya bagi para pelajar. Adapun manfaat penelitian yang diharapkan penulis adalah :

1. Bagi Masyarakat

Memudahkan masyarakat khususnya para pelajar dalam mempelajari sejarah kemerdekaan Indonesia khususnya sejarah pertempuran 10 November 1945.

2. Bagi Mahasiswa

- a. Mengembangkan ilmu pengetahuan dalam multimedia khususnya *Augmented Reality* sehingga dapat digunakan diberbagai bidang aspek kehidupan, terutama sebagai sarana pembelajaran yang menarik.
- b. Sebagai pijakan dan referensi penelitian selanjutnya yang berhubungan dengan *teknologi Game* dan *augmented reality*.

E. Ruang Lingkup Penelitian

Adapun ruang lingkup penelitian pada penelitian ini yaitu sebagai berikut :

1. *Game* yang dibuat berfokus pada sejarah memperjuangan kemerdekaan pada 10 November 1945 dalam upaya mempertahankan kemerdekaan Indonesia.
2. *Object player* adalah sebuah *Tank* yang dimana diberikan misi untuk menghalau dan menghentikan serangan musuh yaitu sebuah *Turret* dan *Tank* musuh.
3. *Game* hanya dapat dimainkan ketika memiliki Android 7.0 keatas dan Support AR core.

F. Sistematika Penulisan

BAB I PENDAHULUAN

Latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat, dan sistematika penulisan semuanya tercakup dalam bab ini.

BAB II TINJAUAN PUSTAKA

Bab ini berisikan mengenai teori-teori yang berfungsi sebagai landasan dasar dan dukungan penelitian yang dilakukan penulis.

BAB III METODE PENELITIAN

Teknik-teknik atau metode yang digunakan untuk melaksanakan penelitian atau solusi yang disarankan dijelaskan dalam bab ini.

BAB IV HASIL DAN PEMBAHASAN

Bab ini memuat ringkasan hasil pengujian berupa tampilan program yang dihasilkan, serta penjelasan cara pemanfaatannya.

BAB V KESIMPULAN DAN SARAN

Bab ini berisikan kesimpulan dan saran-saran penelitian yang nantinya dapat digunakan oleh peneliti selanjutnya guna untuk memandu penelitian selanjutnya.

BAB II TINJAUAN PUSTAKA

A. Landasan Teori

1. Sejarah

Sejarah adalah suatu realitas yang terjadi pada masa lampau yang memuat fakta dan peristiwa yang unik dan berlaku. Sejarah merupakan suatu hal yang tidak bisa dilepas dari manusia itu sendiri, kita sebagai seorang manusia yang hidupnya dinamis akan terus menciptakan sebuah sejarah dan kemudian bermanfaat bagi kehidupan sekarang atau yang akan datang (Cikka, 2019).

a. Pentingnya Mempelajari Sejarah

Dalam mempelajari sejarah terdapat manfaat dan tujuan yang dapat dimabil seperti (Cikka, 2019):

- 1) Mendapatkan informasi mengenai asal usul usatu kejadian atau peristiwa pada masa terdahulu serta dapat mengambil pembelajaran dari kejadian atau peristiwa-peristiwa tersebut.
- 2) Dapat membentuk watak dan perilaku generasi muda dengan mempelajari perjuangan-perjuangan mereka terdahulu. Dimana generasi muda dapat mempelajari kesalahan masa lampau agar tidak diulangi dan mengambil pembelajaran yang perlu dikembangkan dan dilestarikan.
- 3) Dapat memupuk semangat masyarakat generasi sekarang agar meningkatkan prestasi yang telah diraih oleh pendahulu meraka serta mengembangkanya menjadi lebih baik.
- 4) Masyarakat dapat memelihara dan melestarikan peninggalan-peninggalan pendahulu meraka baik itu berupa peninggalan benda-benda atau ilmu pengetahuan.

Kesadaran dalam mempelajari sejarah harus dipupuk dan dipelajari oleh setiap masnyarakat dan sudah menjadi urusan berbangsa dan bernegara. Jika tidak merghargai para pahlawan terdahulu bisa jadi mereka tidak pancasilais dan buat apa mereka menjadi warga negara tersebut. Pentingnya mempelajari sejarah ini kita dapat menghagai nilai kepahlawanan, Jika kita

tidak memiliki kesadaran akan mempelajari sejarah maka kita tidak akan pernah tahu kenapa bangsa ini ada sehingga kita kehilangan identitas kita sebagai masyarakat itu sendiri (Abbas, 2017).

b. Sejarah Kemerdekaan Republik Indonesia

Kemerdekaan Republik Indonesia memiliki sejarah yang cukup panjang dan banyak menguras pengorbanan baik itu berupa material ataupun nyawa para pahlawan yang gugur. Kemerdekaan Indonesia awal mulanya dimulai dari berita ditajutkannya bom ke 2 kota di Jepang yaitu Nagasaki dan Hiroshima. Nagasaki di bom pada tanggal 6 Agustus 1945 dan Hiroshima pada tanggal 9 Agustus 1945. Kedua peristiwa tersebut membuat Jepang menyerah kepada Amerika dan sekutunya. Tidak tinggal diam Indonesia setelahnya mengubah nama BPUPKI (Badan Penyelidik Usaha-usaha Persiapan Kemerdekaan Indonesia) berganti nama menjadi PPKI (Panitia Persiapan Kemerdekaan Indonesia) yang menegaskan bahwa Indonesia berkeinginan untuk mencapai kemerdekaannya.

Soekarno, Moh. Hatta, sebagai pemimpin PPKI, dan Radjiman Wedyodiningrat, mantan ketua BPUPKI, diterbangkan ke Dalat, 250 kilometer timur laut Saigon untuk bertemu dengan Marsekal Terauchi di Vietnam. Mereka diberitahu bahwa pasukan Jepang berada di ambang kekalahan dan bahwa Indonesia akan memperoleh kemerdekaan. Sutan Syahrir yang berada di Indonesia mendengar dari radio bahwa Jepang telah menyerah kepada Sekutu pada 10 Agustus 1945. Syahrir mengetahuinya dari siaran radio asing, yang saat itu ilegal. Kabar tersebut akhirnya menyebar di kalangan pemuda, khususnya pendukung Syahrir.

Gejolak tekanan yang menuntut pengambilalihan kekuasaan Indonesia mencapai puncaknya pada 16 Agustus 1945, ketika para pengikut Syahrir meningkatkan tuntutan mereka. Kemudian mereka berkumpul di rumah Hatta pada sore hari, sedangkan pendukung Syahrir tiba di rumah Soekarno setelah pukul 10 malam. Ke-15 pemuda itu meminta agar Soekarno segera mendeklarasikan kemerdekaan Indonesia melalui radio, yang disusul dengan kudeta militer.

Mereka juga menolak upaya PPKI untuk memproklamasikan kemerdekaan pada 16 Agustus 1945.

Mengetahui proklamasi tanpa pertumpahan darah tidak mungkin lagi, Soekarno dan Moh. Hatta bersama anggota PPKI lainnya untuk rapat dan mempersiapkan teks proklamasi yang kemudian akan dibacakan pada pagi hari tanggal 17 Agustus 1945.

Sehingga tepat pada hari Jumat tanggal 17 Agustus 1945 Proklamasi Kemerdekaan Indonesia yang dibacakan oleh Ir. Soekarno yang kemudian ia menjadi Presiden Dan Moh. Hatta sebagai Wakil Presiden pertama Indonesia (Ramadhani, 2020).

c. Sejarah 10 November 1945

Status Indonesia tidak stabil setelah Proklamasi 17 Agustus 1945. Indonesia saat itu masih dalam pergolakan, terutama antara rakyat dan kekuatan asing. Salah satu peristiwa penting dalam sejarah Republik Indonesia adalah Hari Pahlawan yang jatuh pada tanggal 10 November. Karena pada tanggal 10 November 1945 terjadi pertempuran besar pascakemerdekaan yang dijuluki Pertempuran Surabaya. Menyusul proklamasi kemerdekaan Indonesia pada 17 Agustus 1945, pemerintah mengeluarkan dekrit yang mengamanatkan bahwa bendera nasional Sang Saka Merah Putih dikibarkan di seluruh negeri mulai 1 September 1945. Kampanye pengibaran bendera menyebar ke seluruh pelosok tanah air, termasuk Surabaya (Bone.co.id, 2020).

Pasukan Inggris tiba di Jakarta pada pertengahan September dan tiba di Surabaya pada tanggal 25 September 1945. Pasukan Inggris bergabung dengan AFNEI (Allied Forces Netherlands East Indies) dan datang bersama pasukan dari NICA (Netherlands Indies Civil Administration). Misi mereka adalah melucuti tentara Jepang dan mengembalikan mereka ke Jepang, serta membebaskan tawanan perang Jepang dan mengembalikan Indonesia kepada pemerintah Belanda sebagai koloni. Warga Surabaya marah karena mereka percaya Belanda telah menghina kemerdekaan Indonesia dan bendera Merah Putih.

Mereka berdemonstrasi dengan membentuk rantai manusia di depan Hotel Yamato, menuntut agar bendera Belanda dicabut sebelum mengibarkan bendera Indonesia. Para pejabat Indonesia bertemu dengan Belanda pada 27 Oktober 1945, dan pertemuan itu berakhir dengan nada masam ketika Ploegman mengeluarkan pistol dan terjadi perkelahian di ruang perundingan. Perkelahian terjadi sampai Ploegman dicckik sampai mati oleh Sidik di Hotel Yamato. Beberapa warga berusaha masuk ke hotel, tetapi Hariyono dan Koesno Wibowo merobek bagian biru bendera Belanda, mengubahnya menjadi merah putih.

Pihak Indonesia dan Inggris kemudian sepakat untuk menandatangani gencatan senjata pada tanggal 29 Oktober. Namun, keesokan harinya, kedua belah pihak bertempur, mengakibatkan Brigadir Jenderal Mallaby, komandan tentara Inggris, ditembak dan dibunuh sampai milisi meledakkan mobilnya. mengemudi masuk. Pengganti Mallaby, Mayor Jenderal Robert Mansergh, mengeluarkan ultimatum kepada semua pemimpin dan warga Indonesia bersenjata untuk melapor dan meletakkan senjata mereka di lokasi yang telah ditentukan. Tidak hanya itu, mereka juga mengeluarkan ultimatum kepada rakyat Indonesia dengan mengangkat tangan pada pukul 06.00 tanggal 10 November 1945. Ultimatum tersebut membuat marah warga Surabaya hingga konflik pada tanggal 10 November. Peperangan ini berlangsung setidaknya selama 3 minggu.

2. *Game*

Game berasal dari kata bahasa Inggris yang memiliki arti dasar "Permainan". Permainan dalam hal ini merujuk pada pengertian "kelincahan intelektual" (*intellectual playbility*). *Game* juga bisa diartikan sebagai arena keputusan dan aksi pemainnya. Ada target – target dan misi untuk dapat dicapai pemainnya. Kelincahan intelektual, pada tingkat tertentu merupakan ukuran sejauh mana *Game* itu menarik untuk dimainkan secara maksimal (Rohmawati et al., 2019).

a. Elemen Dasar *Game*

Menurut Teresa Dillon, elemen-elemen dasar *Game* dapat dilihat seperti berikut :

1) *Game Rules*

Game Rules (aturan permainan) adalah aturan perintah, cara menjalankan, dan fungsi item dan karakter yang ada di dunia *Game*. Dunia *Game* dapat berbentuk pulau, planet imajiner, dan lokasi serupa lainnya yang digunakan sebagai setting tempat dalam bermain *Game*.

2) Plot

Plot menawarkan informasi tentang apa yang akan dilakukan pemain dalam permainan atau instruksi tentang cara menyelesaikan tugas dalam permainan.

3) Tema

Tema permainan dalam permainan biasanya adalah genre permainan, yang berisi informasi tentang jenis permainan.

4) Karakter

Pemain sebagai karakter utama dan karakter lain dengan karakteristik dan sifat tertentu.

5) Objek

Hal-hal yang digunakan pemain untuk memecahkan masalah atau kesulitan untuk memainkannya, pemain harus memiliki kemampuan dan pengetahuan yang diperlukan.

6) Teks, Grafik dan Suara

Game adalah perpaduan antara teks, gambar, dan media suara, meskipun beberapa mungkin tidak menggunakan ketiganya.

7) Animasi

Animasi ini selalu hadir di lingkungan *Game*, terutama dalam hal pergerakan karakter, properti, dan item.

8) *User Interface*

Ini adalah fitur yang memungkinkan pemain untuk berinteraksi dengan *Game*.

b. Jenis-Jenis *Game*

Game dapat dikategorikan beberapa jenis menurut platform, dimensi dan genre *Game*. Platform merupakan kombinasi spesifik dari komponen elektronik atau perangkat computer yang memungkinkan *Game* dapat beroperasi. Menurut platformnya *Game* dibagi menjadi beberapa jenis yaitu :

1) *Game Arcade*

Yaitu *Game* yang biasanya memiliki box tersendiri yang memang dikhususkan untuk satu *Game* tertentu. Memiliki sebuah fitur yang dapat membuat pemain lebih merasa masuk dalam *Game* tersebut.

2) *Game PC*

Yaitu *Game* yang beroperasi atau dimainkan di computer pribadi.

3) *Game Konsol*

Yaitu *Game* yang dimainkan menggunakan sebuah konsol tertentu seperti *Xbox 360*, *Playstation 1*, *Playstation 2*, *Playstation 3*, *Playstation 4*, *Nintendo wii*, dan masih banyak lagi.

4) *Hanheld Games*

Yaitu *Game* yang dimainkan di konsol khusus yang bisa dibawa kemana-mana (portable), seperti *Nintendi Swich*, *PSP* dan *Steam Deck*.

5) *Game Mobile*

Yaitu *Game* yang dapat dimainkan khusus di *mobile phone* atau smart phone.

Genre merupakan pola umum yang ada pada *Game* tersebut. Seiring perkembangan jaman genre *Game* pun semakin banyak. Namun secara umum *Game* dapat dibagi menjadi beberapa jenis genre, yaitu :

1) *Game Strategi*

Yaitu *Game* yang memusatkan permainan dalam kemampuan mengambil sebuah keputusan untuk meraih kemenangan.

2) *RPG (Role-Playing Games)*

Yaitu *Game* yang dimana kita memerankan karakter dalam latar fiksi yang dimana pemain harus mengikuti jalan cerita (*Strory Line*) yang ada dalam *Game* tersebut dan memaksa pemain untuk mengikuti

perkembangan karakter yang dimankannya dan melihat perkembangannya (*Character Development*).

3) *Game Olahraga (Sports Games)*

Yaitu *Game* yang mensimulasikan permainan olahraga yang ada di dunia nyata.

4) *Game Simulasi*

Yaitu *Game* yang mensimulasikan pemain sebagai orang yang melakukan suatu kegiatan layaknya berada di dunia nyata dengan tugas untuk mencapai tujuan tertentu.

5) *Game Petualangan*

Hampir sama dengan genre *RPG Game* petualangan merupakan *Game* cerita interaktif mengenai karakter yang dapat dikontrol oleh pemain.

6) *Game Puzzle*

Yaitu *Game* yang bertujuan untuk memecahkan semua teka-teki atau puzzle.

7) *Game Aksi*

Yaitu *Game* yang membutuhkan keterampilan seperti mengelola informasi sensorik dan tindakan secara tepat. *Game* ini memerlukan Kerjasama antara mata dan tangan.

8) *Game Edukasi*

Yaitu *Game* yang bertujuan untuk menambah edukasi kepada pemain dari berbagai bidang yang dipilih dalam sebuah permainan/*Game* seperti sejarah, matematika, lingkungan, biologi dan berbagai bidang ilmu pengetahuan lainnya.

Menurut dimensinya jenis-jenis game dapat dibagi menjadi 3 jenis yaitu sebagai berikut (Tanjung, 2013):

1) *Game 2D*

Game dua dimensi (2D) adalah *Game* yang semua objeknya berada dalam satu bidang datar. Dalam *Game 2D*, pemain hanya dapat bergerak secara horizontal dan vertikal, atau dengan menggunakan gerakan koordinat, mereka hanya dapat bergerak pada sumbu X dan Y,

Ada dua gerakan kamera dalam *Game 2D*. Yang pertama adalah kamera statis, di mana latar belakang dan objek dalam *Game 2D* tidak bergerak sama sekali; tetris adalah contoh dari jenis ini. Yang kedua adalah Side Scrolling, di mana *Game* yang kita mainkan memiliki kamera yang bisa bergeser ke kanan atau ke kiri dengan kecepatan tergantung pada gerakan karakter dan kecepatan saat kita memainkan/memindahkan *Game*; judul seperti Super Mario Bros., Sonic, dan Megaman termasuk dalam kategori ini.

2) *Game 2.5D*

Setelah *Game 2D*, ada *Game 3D* dengan tampilan datar (3D plane). Jenis permainan ini tidak 2D atau sepenuhnya 3D. Jenis teori grafik ini dikenal sebagai 2.5D atau pseudo-3D, dan proyeksi isometrik, diametrik, atau trimetri dalam teori permainan. *Gameplay Game 2.5D* biasanya mirip dengan *Game 2D*, di mana kita hanya bisa bergerak secara horizontal dan vertikal, meskipun item tertentu menggunakan teknik rendering 3D.

3) *Game 3D*

Game 3D menggunakan representasi data geometris tiga dimensi (X, Y, Z) yang disimpan di komputer untuk perhitungan dan rendering visual 2D. Pemain dapat melihat sebuah objek dari sudut 360°.

3. *Augmented Reality*

Augmented Reality atau dalam bahasa Indonesia diterjemahkan menjadi Realitas Tambahan adalah sebuah teknik yang menggabungkan benda dua dimensi maupun tiga dimensi lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata. Dalam kehidupan sehari-hari *Augmented Reality* sudah mendominasi pasar dunia. Banyak bisnis mulai menggunakan *Augmented Reality* untuk mempromosikan produk mereka. Perusahaan mobil misalnya, akan menggunakan *augmented reality* untuk mendemonstrasikan produk mereka dalam bentuk desain mobil 3D yang dibuat mirip dengan produk mereka (Pamoedji & Maryuni, 2017). Ada dua inputan dalam *augmented reality* (Harahap et al., 2020):

a. *Marker*

Simbol/gambar digunakan sebagai titik acuan bagi komputer untuk menampilkan gambar dalam *augmented reality* dengan penanda. Kamera dalam sistem mengumpulkan foto-foto objek secara real time dan menganalisisnya untuk memperkirakan lokasi, orientasi, dan pergerakan tampilan visualisasi dalam kaitannya dengan objek target.

b. Markerless

Sistem *augmented reality* Markerless menggunakan perpaduan perangkat teknologi seperti akselerometer, kompas, dan GPS (GPS).

4. Perangkat Pengembang

Perangkat pengembang merupakan kumpulan *software* atau perangkat lunak yang digunakan oleh *Developer* dalam mengembangkan dan membuat suatu aplikasi. Perangkat pengembang yang dipakai dalam penelitian ini adalah sebagai berikut :

a. Blender 3D

Blender adalah program pemodelan 3D yang gratis dan *open source*. *Modeling*, *rigging*, animasi, simulasi, *rendering*, *compositing*, dan pelacakan gerak, serta informasi video dan produksi *Game*, semuanya didukung oleh *Blender*. *Blender* sangat ideal untuk individu maupun studio kecil yang mengerjakan proyek 3D (Zebua et al., 2020).

b. Unity

Unity adalah alat untuk membuat *Game* multi-platform yang mudah digunakan. *Unity* menarik dan menawarkan beragam aplikasi profesional. *Unity* memiliki antarmuka pengguna yang sederhana. Editor ini didirikan setelah puluhan ribu jam diinvestasikan untuk menjadikannya nomor satu dalam urutan peringkat dua editor *Game* teratas. Visual *Unity* dibuat menggunakan grafik tingkat tinggi untuk *OpenGL* dan *DirectX*. *Unity* mendukung semua format file, terutama yang umum seperti semua jenis aplikasi seni (Kustriawan, 2018).

c. Adobe Photoshop

Adobe Photoshop adalah program pengeditan gambar yang canggih dengan berbagai fitur yang dikhususkan untuk pengeditan gambar/foto dan

pembuatan efek. Adobe Photoshop merupakan software pengolah gambar yang sangat powerful dengan segala fasilitasnya, seperti yang terlibat pada berbagai website, brosur, koran, majalah, dan media lainnya (Aulianti et al., 2021).

d. Visual studio

Microsoft *Visual studio* adalah paket perangkat lunak yang dapat digunakan untuk membuat program dalam bentuk aplikasi konsol, aplikasi Windows, atau aplikasi Web, baik itu aplikasi komersial, aplikasi pribadi, atau komponen aplikasi. Kompiler, SDK, Lingkungan Pengembangan Terintegrasi (IDE), dan dokumentasi semuanya disertakan dengan *Visual studio* (biasanya Perpustakaan MSDN). Visual C++, Visual C#, Visual Basic, Visual Basic.NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe adalah beberapa kompiler yang disertakan dalam paket *Visual studio*.

Microsoft *Visual studio* dapat digunakan untuk membuat kode asli (bahasa mesin yang berjalan di Windows) atau aplikasi kode terkelola (dalam bentuk Microsoft Intermediate Language di atas .NET Framework).

e. Bahasa Pemrograman C#

C# (diucapkan "C Sharp") adalah bahasa pemrograman berorientasi objek yang dibuat oleh Microsoft sebagai bagian dari upaya inisiatif kerangka. Bahasa komputer ini berbasis C++, namun dengan penyederhanaan tertentu telah dipengaruhi oleh karakteristik atau fitur bahasa yang ada pada bahasa pemrograman lain seperti Java, Delphi, Visual Basic, dan lain-lain. C# juga dapat diinstal pada komputer dan diproses secara offline. C# adalah bahasa pemrograman pengembangan *Game* yang juga dapat digunakan di *Unity* untuk membuat *Game* model 2D dan 3D. Akibatnya, C# dapat dihubungkan dengan *Unity* untuk membuat *Game* arsitektur bangunan dan simulasi untuk pemodelan dan rendering dalam satu aplikasi (Mongi et al., 2018).

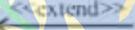
f. **UML (*Unfied Modeling Language*)**

UML (*Unfied Modeling Language*) adalah suatu bahasa yang digunakan sebagai sarana visualisasi, membangun dan merancang suatu aplikasi atau sistem informasi yang berorientasi objek (Kurniawan, 2018).

Adapun beberapa diagram model yang ada di sediakan oleh UML itu sendiri, contohnya seperti: *Use Case Diagram*, *Activity Diagram* dan *Seqence Diagram*.

1) *Use Case Diagram*

Salah satu dari berbagai diagram UML yang digunakan adalah *Use Case Diagram*, yang memberikan penjelasan visual tentang konteks interaksi antara aktor dan sistem, dan berfungsi sebagai kebutuhan sistem. Setiap use case menguraikan persyaratan perilaku (fungsionalitas) yang benar-benar dibutuhkan aktor untuk mencapai tujuan. Berikut simbol-simbol yang dipakai dalam *Use Case Diagram* (Kurniawan, 2018):

Simbol	Keterangan
	Aktor : Mewakili peran orang, sistem yang lain, atau alat ketika berkomunikasi dengan <i>use case</i>
	<i>Use case</i> : Abstraksi dan interaksi antara sistem dan aktor
	<i>Association</i> : Abstraksi dari penghubung antara aktor dengan <i>use case</i>
	<i>Generalisasi</i> : Menunjukkan spesialisasi aktor untuk dapat berpartisipasi dengan <i>use case</i>
	Menunjukkan bahwa suatu <i>use case</i> seluruhnya merupakan fungsionalitas dari <i>use case</i> lainnya
	Menunjukkan bahwa suatu <i>use case</i> merupakan tambahan fungsional dari <i>use case</i> lainnya jika suatu kondisi terpenuhi

Gambar 1. Simbol *Use Case Diagram*

2) *Activity Diagram*

Activity Diagram merupakan salah satu diagram dari UML itu sendiri yang dimana menyajikan serangkaian kejadian atau tindakan (*work flow*) yang mirip dengan diagram alur (Flow Chart). Reruntutan proses *Activity Diagram* digambarkan secara vertical. Berikut simbol-simbol yang dipakai dalam *Activity Diagram* (Siregar et al., 2018):

Simbol	Nama	Keterangan
	Status awal	Sebuah diagram aktivitas memiliki sebuah status awal.
	Aktivitas	Aktivitas yang dilakukan sistem, aktivitas biasanya diawali dengan kata kerja.
	Percabangan / Decision	Percabangan dimana ada pilihan aktivitas yang lebih dari satu.
	Penggabungan / Join	Penggabungan dimana yang mana lebih dari satu aktivitas kini digabungkan jadi satu.
	Status Akhir	Status akhir yang dilakukan sistem, sebuah diagram aktivitas memiliki sebuah status akhir.
	Swimlane	Swimlane memisahkan organisasi bisnis yang bertanggung jawab terhadap aktivitas yang terjadi.

Gambar 2. Simbol Activity Diagram

3) Sequence Diagram

Sequence Diagram adalah representasi visual tentang bagaimana item berinteraksi satu sama lain dan bagaimana mereka berkomunikasi (melalui indikasi atau petunjuk). *Sequence Diagram* digunakan untuk menjelaskan bagaimana entitas dan sistem berinteraksi, termasuk pesan yang dipertukarkan saat melakukannya sesuai dengan urutan eksekusi. Berikut simbol-simbol yang dipakai dalam *Sequence Diagram* (Siregar et al., 2018):

NO	GAMBAR	NAMA	KETERANGAN
1		<i>Actor</i>	Menggambarkan seseorang atau sesuatu (seperti perangkat, sistem lain) yang berinteraksi dengan sistem.
2		<i>Life Line</i>	Objek <i>entity</i> , antarmuka yang saling berinteraksi.
3		<i>Object Message</i>	Menggambarkan pesan hubungan antar objek yang menunjukkan urutan kejadian yang terjadi.
4		<i>Message to Self</i>	Menggambarkan pesan hubungan objek itu sendiri yang menunjukkan urutan kejadian yang terjadi.

Gambar 3. Simbol *Sequence Diagram*

B. Penelitian Terkait

Berikut ini merupakan beberapa referensi jurnal penelitian sebelumnya tentang pembuatan *Game* sejarah dan *Augmented Reality*. Berikut ini penelitian terdahulu yang berkaitan dengan *Game* edukasi sejarah dan *Augmented Reality* :

Penelitian yang dilakukan oleh Indra S. Boham , Steven R. Sentinuwo dan Alwin M. Sambul, 2017, "Rancang Bangun Aplikasi *Game* Pengenalan Sejarah Perang Tondano", Teknik Informatika Universitas Sam Ratulangi Manado. Pada penelitian tersebut dari 30 responder terdapat 25 (83%) responder yang mengatakan *Gamenya* mudah dimainkan. Terdapat 16 (54%) responder yang menyatakan mereka sangat terhibur dan 13 responder yang terhibur (43%) dan 1 orang yang cukup terhibur (3%). Sebanyak 24 (80%) responder mengatakan tampilan *Gamenya* bagus. Dan dari 30 responder semuanya (100%) mendapatkan pengetahuan setelah memainkan *Game* tersebut. Memberikan hasil akhir bahwa aplikasi *Game* ini dapat menarik masyarakat agar mempelajari sejarah dengan cara interaktif dan menyenangkan (Boham & Sentinuwo, 2017).

Penelitian yang dilakukan oleh Fiqih Fauzan Ar-rafi dan Achmad Udin Zailani, 2020, "RANCANG BANGUN *GAME* EDUKASI SEJARAH BERBASIS *ANDROID* MENGGUNAKAN *GAME ENGINE* *UNITY 3D*", Prodi Teknik Informatika, Fakultas Teknik, Universitas Pamulang. Dengan perancangan aplikasi edukasi ini bertujuan untuk membuat *Game* edukasi yang dapat membuat pelajaran

sejarah menjadi lebih menarik dengan menggunakan metode permainan yang menghibur, sehingga pembelajaran sejarah tidak lagi membosankan (Ar-rafi & Zailani, 2020).

Penelitian yang dilakukan oleh Sheila Nurul Huda dan Fazma Rizqy Alfi Bahri, 2021, "*Game* Edukasi Sejarah Berbagai Peristiwa yang Ada di Indonesia", Program Studi Informatika Universitas Islam Indonesia. Penulis penelitian ini menyatakan bahwa permainan pendidikan sejarah efektif dalam meningkatkan pemahaman siswa tentang informasi pendidikan sejarah yang ditawarkan kepada khalayak umum. Genre Petualangan, adalah yang paling populer karena berbagai kesulitan yang termasuk dalam genre ini membuat pembelajaran melalui permainan instruksional lebih menarik. Karena sangat mudah untuk diunduh, platform seluler menjadi platform populer untuk memainkan *Game* edukasi sejarah (Bahri & Huda, 2022).

Penelitian yang dilakukan oleh Dedi Hidayat dan Dedy Irfan, 2018, "RANCANG BANGUN APLIKASI DENAH KAMPUS UNIVERSITAS NEGERI PADANG MENGGUNAKAN *AUGMENTED REALITY* BERBASIS *ANDROID*", Prodi Pendidikan Teknik Informatika Fakultas Teknik, Universitas Negeri Padang. Penelitian ini menunjukkan bahwa implementasi *Augmented Reality* pada aplikasi ini dapat menyediakan informasi bangunan yang terdapat pada Universitas Negeri Padang secara menyeluruh dan lebih menarik. Masyarakat dan mahasiswa Universitas Negeri Padang akan dapat mempelajari lebih lanjut tentang bangunan universitas berkat aplikasi ini (Dedi Hidayat & Dedy Irfan, 2018).

Pada penelitian yang dilakuakn oleh Christian O. Karundeng, Dringhuzen J. Mamahit dan Brave A. Sugiарso, 2018, "Rancang Bangun Aplikasi Pengenalan Satwa Langka di Indonesia Menggunakan *Augmented Reality*", Teknik Informatika Universitas Sam Ratulangi Manado. Pada penelitian ini Aplikasi *Augmented Reality* dalam memperkenalkan satwa langka Indonesia bisa menjadi sebuah sarana informasi dan teknologi yang menarik untuk memperkenalkan satwa langka yang ada di Indonesia yang terbilang banyak kepada masyarakat (Karundeng et al., 2018).

C. Kerangka Pikir



Gambar 4. Kerangka Pikir

Sejarah merupakan hal yang tidak boleh dilupakan sebagai sosok manusia, apalagi sebagai masyarakat Indonesia yang memiliki sejarah panjang dalam meraih dan mempertahankan kemerdekaannya. Pelajaran sejarah terhadap sebagian orang dapat sangat membosankan. Kita mengingat nama dan kapan peristiwa itu terjadi dan belajar melalui buku. Hal ini membuat generasi mudah bosan mempelajari sejarah. Maka dari itu diperlukan sebuah inovasi dalam belajar sejarah.

Implementasi *Augmented Reality* dalam pembuatan *Game mobile* untuk memperkenalkan sejarah kemerdekaan Indonesia khususnya kejadian pertempuran yang terjadi pada 10 november 1945 sebagai pengingat perjuangan akan pahlawan pada zaman dahulu dan meningkatkan rasa nasionalisme. Selain sebagai sarana hiburan *Game* ini diharapkan dapat dijadikan sebagai sarana edukasi bagi

masyarakat dan khususnya bagi pelajar. Implementasi *Augmented Reality* ini bertujuan agar *Game* lebih menarik dimainkan dan lebih interaktif terhadap lingkungan sekitar.



BAB III

METODE PENELITIAN

A. Tempat dan Waktu Penelitian

Waktu penelitian dilakukan pada bulan Juli 2022 yang bertempat di SMK Negeri 2 SELAYAR Jl. Anas Ahmad no.2 Polebuning Kecamatan Bontomanai Kabupaten Kepulauan Selayar.

B. Alat dan Bahan

Dalam pembuatan *Game* ini, alat dan bahan meliputi *hardware* dan *software*. *Hardware* berupa:

1. Laptop ACER SWIFT SF314-42 dengan spesifikasi :
 - Processor AMD RAYZEN 5 6 CPUs 2.4GHz
 - AMD Radeon Graphic card
 - Memory 8GB
 - SSD 500GB
 - OS WINDOWS 10
2. Smartphone Redmi Note 10 5G dengan spesifikasi :
 - Processor MediaTek Dimensity 700 Dual 5G
 - Memory 4GB
 - Pneyimpanan 128GB
 - OS ADROID 11

Sedangkan *software* berupa aplikasi pembuatan *Game* seperti :

1. Unity
2. Blender
3. visual studio
4. adobe photoshop
5. sistem operasi windows 10.

C. Perancangan Sistem

Perancangan sistem merupakan tahapan dalam metodologi pengembangan perangkat lunak yang terjadi setelah tahap analisis kebutuhan untuk menawarkan gambaran yang lengkap dari aplikasi. Perancangan sistem dapat didefinisikan

sebagai menggambar, merancang, dan membuat sketsa dari beberapa bagian yang berbeda ke dalam sistem yang utuh dan berfungsi.

1. Rancangan Penelitian

Alur rancangan penelitian yang digunakan dalam penelitian ini menggunakan kerangka penelitian sebagai berikut:



Gambar 5. Rancangan Penelitian

Penulis menggunakan metode pengembangan multimedia oleh Luther pada tahun 1994 yang telah disempurnakan oleh Sutopo pada tahun 2003 dalam penelitian ini. Model pengembangan multimedia ini memiliki enam tahap: konsep, perancangan, pengumpulan bahan, pembuatan, pengujian, dan distribusi. Rincian tahap-tahap pengimplementasian metode Luther-Sutopo (1994) sebagai berikut (Fatimah et al., 2017) :

a. Tahap Konsep (*Concept*)

Dalam metode pembuatan multimedia ini, konsep merupakan tahapan awal dari pembuatan aplikasi. Penulis mengidentifikasi tujuan penelitian pada tahap konsep, termasuk identifikasi audiens, jenis aplikasi (presentasi, interaktif, dll), tujuan aplikasi (informasi, hiburan, pelatihan, dll), dan persyaratan umum dari sebuah aplikasi.

b. Tahap Perancangan (*Design*)

Pada tahap Perancangan (*Design*), penulis melakukan analisis berupa *Design Treatment*, *Storyline*, dan *Game Rules*. Analisis kebutuhan (kebutuhan fungsional dan non-fungsional), desain sistem, desain struktur menu, teknik desain, dan desain antarmuka juga dilakukan oleh penulis.

c. Tahap Pengumpulan Bahan (*Material Collecting*)

Penulis mengumpulkan bahan-bahan dan bahan-bahan yang diperlukan dalam memproduksi aplikasi, seperti tokoh atau karakter, properti, objek, dan ruang lingkup (*area*) yang akan ditampilkan, selama tahap pengumpulan bahan.

d. Tahap Pembuatan (*Assembly*)

Tahapan ini adalah tahapan pembuatan aplikasi berdasarkan objek atau material yang telah dikumpulkan dari tahapan sebelumnya.

e. Tahap Pengujian (*Testing*)

Pada tahap pengujian, penulis menguji aplikasi yang telah dibuat dengan menjalankan aplikasi dan melihat apakah sudah seperti yang diharapkan atau tidak.

f. Tahap Distribusi (*Distribution*)

Pada tahap distribusi, penulis mendistribusikan aplikasi yang telah dibuat kepada sampel *User* untuk mengetahui apakah tujuan penelitian telah terpenuhi atau belum.

2. Konsep Pembangunan *Game*

Game ini berlatarkan pada kejadian 10 November yang ada di Surabaya. Saat permainan dimulai *Game* akan memperlihatkan *CutScene* dialog yang menjelaskan sejarah 10 November 1945 dan bagaimana peperangan ini bisa terjadi. Secara umum pada *Game* ini pemain berperan sebagai *Tank* yang harus menjerat musuh yang berupa *tank* dan *Turret* yang merupakan tantara inggris dengan membuat HP mereka menjadi 0. Pada *Game* ini terdapat 3 jenis tingkat kesulitan yaitu : *easy*, *normal* dan *hard*. Masing-masing tingkat kesulitan akan memberikan tantangan yang berbeda dengan jumlah musuh yang diberikan berbeda. Ketika semua musuh telah dikalahkan maka permainan akan berakhir dan

pemain telah meraih kemenangan, akan tetapi Ketika HP pemain mencapai 0 maka permainan akan selesai pula (*Game over*) dengan kekalahan pemain. Setelah meraih kemenangan permainan akan berlanjut pada *CutScene* konklusi yang terjadi pada saat perang berakhir.

3. *Design Treatment*

Design Treatment berisis informasi dasar dari sebuah aplikasi yang akan dikembangkan. Adapaun *Design Treatment* pada *Game* yang dibangun pada penelitian ini adalah sebagai berikut :

- a. *Judul Game* : *Thank's*
- b. *Definisi Judul* : Pembambilan judul *Game* ini berdasarkan dari kalimat Bahasa Inggris "*Thank*" yang berarti terima kasih dimana kita masyarakat sekarang berterimakasih atas pengorbanan pahlawan yang berjuang pada masa lalu. Dan kata "*Thank*" juga bisa jadi plesetan dari kata "*Tank*" yang merupakan konsep *Game* ini.
- c. *Genre* : Aksi & Edukasi
- d. *Platform Game* : *Mobile*
- e. *Software* digunakan : *Unity, Blender 3D, Visual studio*
- f. *Target Player* : Remaja sampai kalangan dewasa
- g. *Player Interaction* : *Single Player (Player vs Game)*
- h. *Basic Goal* : Pemain berperan sebagai *Game* dan bertugas untuk menghalau dan menjatuhkan musuh yang menyerang.
- i. *Basic Mechanic* : *Games Shooting*

4. *Storyline*

Game ini bercerita tentang perjuangan kemerdekaan masyarakat Indonesia pada tanggal 10 November 1945 yang bertempat di Surabaya. Pemain berperan

sebagai *Tank* yang bertugas untuk menghalau dan menjatuhkan musuh yang merupakan pasukan tantara inggris yang berupa sebuah *Turret* dan *Tank*-musuh dengan cara menembaknya dan memuat HP (*Health Point*) lawan menjadi 0. Jika HP (*Health Point*) pemain telah mencapai 0, maka permainan akan berakhir (*Game Over*). Akan ada 3 tingkat kesulitan dalam *Game* ini yaitu *easy*, *normal* dan *hard*. Setiap tingkat kesulitan akan mempunyai tantangan tersendiri seperti *easy* akan mempunyai 1 *Turret* dan 1 *Tank*, *normal* akan mempunyai 2 *Turret* dan 1 *Tank* dan *hard* akan mempunyai 3 *Turret* dan 2 *Tank*.

5. Analisa Kebutuhan Fungsional

Analisis kebutuhan fungsional meliputi proses kegiatan yang diimplementasikan dalam suatu sistem dan menjelaskan kebutuhan sistem yang dibutuhkan agar dapat beroperasi dengan baik sesuai kebutuhan. Perangkat lunak Menggunakan UML (*Unified Modelling Language*). Instrumen yang digunakan adalah *Use Case Diagram* dan *Activity Diagram*.

a. *Use Case Diagram*

Use Case Diagram pada *Game* ini dapat dilihat pada gambar dibawah ini :

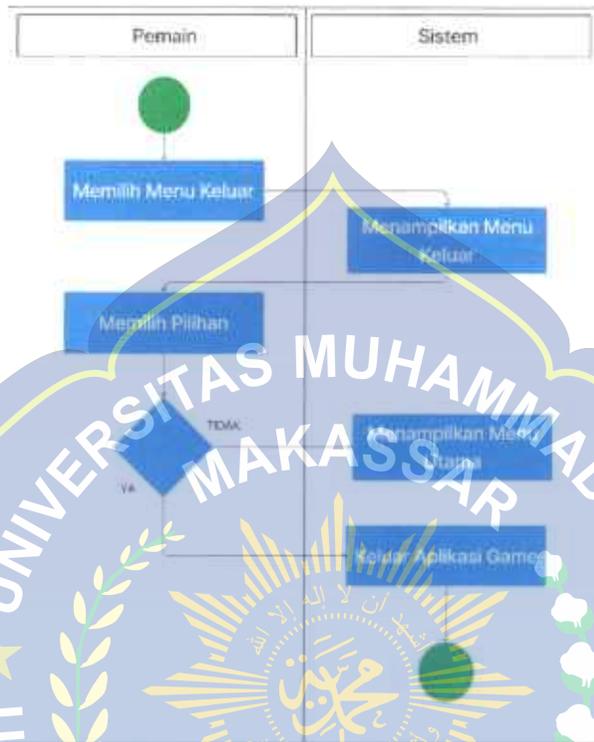


Gambar 7. Activity Diagram Menu Mulai

Keterangan gambar

- a) Pemain memilih menu mulai untuk memulai *Game*.
 - b) Sistem akan menampilkan menu tingkat kesulitan.
 - c) Pemain akan memilih tingkat kesulitan sebelum bermain *Game*.
 - d) Sistem akan menampilkan *Game* atau permainan.
 - e) Setelah pemain menyelesaikan permainan maka sistem akan kembali ke menu utama.
- 2) Activity Diagram Menu Keluar

Activity Diagram menu keluar bisa dilihat seperti pada gambar dibawah ini :



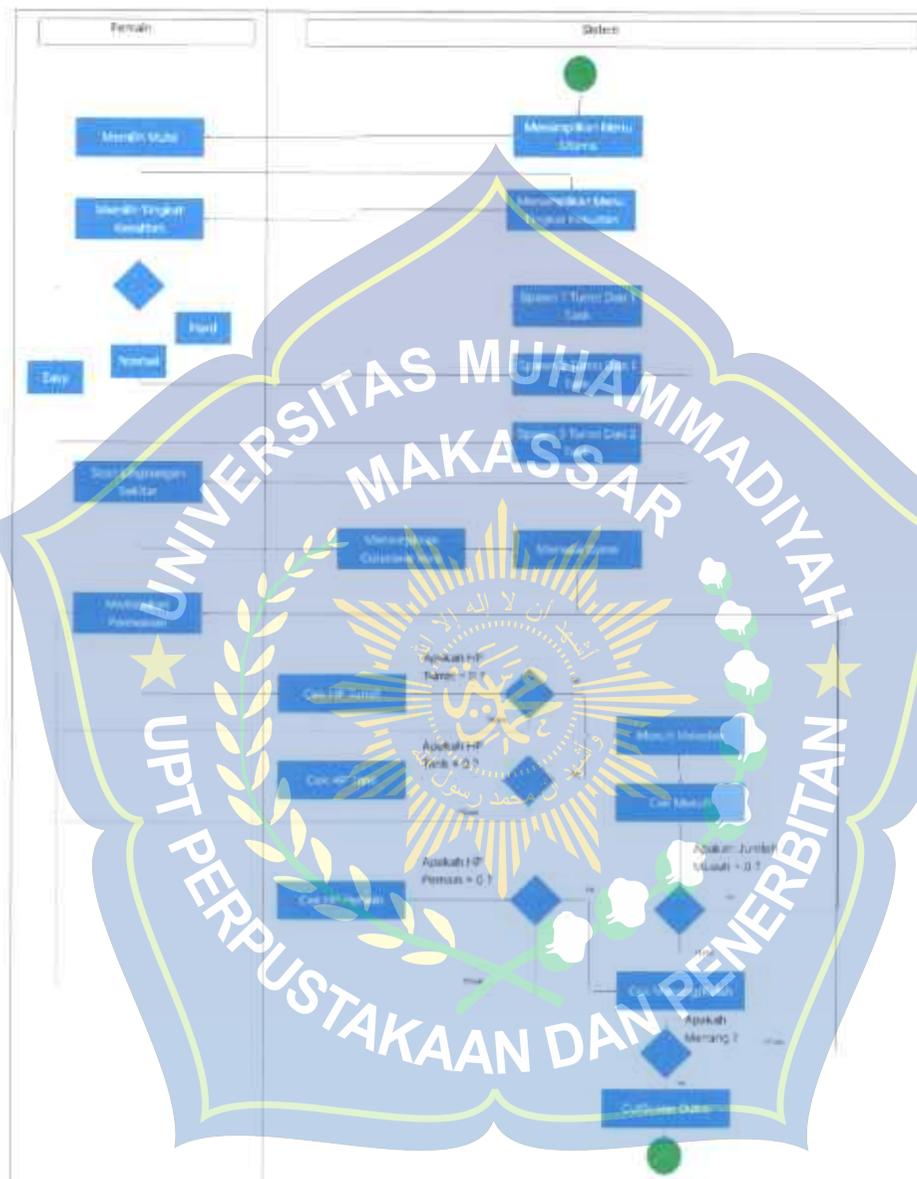
Gambar 8. *Activity Diagram* Menu Keluar

Keterangan gambar :

- a) Pemain memilih menu keluar untuk keluar *Game*.
- b) Sistem akan menampilkan menu keluar.
- c) Pemain akan memilih tombol ya atau tidak.
- d) Jika Pemain memilih ya maka akan keluar aplikasi *Game*. Jika tidak, maka akan menampilkan menu utama.

3) *Activity Diagram* Ketika *Game* Dimulai

Activity Diagram Ketika *Game* dimulai bisa dilihat seperti pada gambar dibawah ini :



Gambar 9. Activity Diagram Ketika Game Dimulai

Keterangan gambar :

- a) Sistem menampilkan menu utama.
- b) Pemain memilih menu lalu sistem akan menampilkan menu pemilihan tingkat kesulitan.
- c) Pemain memilih tingkat kesulitan yaitu easy, normal, atau hard. Setelah pemain memilih tingkat kesulitan, maka sistem akan menyimpan berapa *Turret* dan *Tank*-musuh yang akan dimunculkan dalam *Game* sesuai tingkat kesulitan yang dipilih.

- d) Pemain akan diarahkan menScan lingkungan sekitar menggunakan kamera *Smartphon*nya agar dijadikan field permainan nanti.
- e) Sistem akan memulai permainan dan pemain akan memainkannya. Ketika HP *Turret* atau *Tank*-Musuh 0 maka mereka akan meledak dan akan dihitung apakah jumlah musuhnya sudah 0 atau tidak. Jika iya maka permainan selesai. Jika tidak maka permainan tetap berjalan.
- f) Ketika HP pemain mencapai 0 maka permainan akan selesai pula.
- 4) *Activity Diagram* Ketika Pemilihan *Tank*
- Activity Diagram* Ketika Pemilihan *Tank* dimulai bisa dilihat seperti pada gambar dibawah ini :



Gambar 10. *Activity Diagram* Ketika Pemilihan *Tank*

Keterangan gambar :

- Pemain memilih menu pilih *Tank*.
- Sistem akan menampilkan menu dimana pemain akan melihat serta memilih *Tank* yang akan dipilih.

- c) Pemain akan memilih jenis *Tank*.
- d) Sistem akan menampilkan menu tingkat kesulitan.
- e) Pemain akan memainkan *Tank*.

6. Analisa Kebutuhan Non-Fungsional

Analisis kebutuhan non-fungsional merupakan langkah penting dalam menentukan spesifikasi sistem. Spesifikasi ini juga berisi pertanyaan mengenai komponen apa yang diperlukan untuk memulai sistem. Analisa kebutuhan non-fungsional ini menjelaskan tentang analisis kebutuhan perangkat keras, perangkat lunak, dan pengguna.

a. Perangkat Keras

Salah satu aspek terpenting dari penelitian ini adalah perangkat keras (*Hardware*), karena program yang dibangun tidak akan dapat beroperasi tanpanya. Agar program berfungsi dengan benar, diperlukan perangkat keras yang sesuai dengan kebutuhan aplikasi. *Developer* sebagai pembuat *Game* ini memiliki 2 perangkat keras yaitu *Smartphone* dan laptop. *Smartphone* berguna untuk menguji aplikasi dan laptop berguna untuk membuat aplikasi.

Perangkat keras yang dibutuhkan oleh *Developer* dalam pembuatan *Game* ini bisa dilihat pada Tabel 1 dan Tabel 2.

Tabel 1. Perangkat Laptop *Developer*

<i>Hardware</i>	Spesifikasi
Jenis Perangkat	Laptop
Processor	Rayzen 5 4500U @2.4GHz
Memory	8 GB
SSD	500GB
VGA	AMD Radeon

Tabel 2. Perangkat *Smartphone Developer*

<i>Hardware</i>	<i>Spesifikasi</i>
Jenis Perangkat	<i>Smartphone Android</i>
<i>Processor</i>	MediaTek Dimensity 700 Dual 5G
<i>Memory</i>	4 GB
Penyimpanan	128GB
<i>Andoid Version</i>	11

Perangkat keras yang dibutuhkan oleh *User* dalam memainkan *Game* ini bisa dilihat pada Table 3.

Tabel 3. Perangkat *Smartphone User*

<i>Hardware</i>	<i>Spesifikasi</i>
Jenis Perangkat	<i>Smartphone Android</i>
<i>Processor</i>	Helio G70
<i>Memory</i>	3 GB
Penyimpanan	32GB
<i>Andoid Version</i>	8 keatas

Perangkat keras yang dibutuhkan oleh *Developer* dalam mengukur intensitas cahaya pada *Game* ini bisa dilihat pada Table 5 dan 6 dibawah ini.

Tabel 4.

Tabel 4. Perangkat Pengujian I Untuk Intensitas Cahaya

<i>Hardware</i>	<i>Spesifikasi</i>
Jenis Perangkat	<i>Smartphone Android</i>
Nama Perangkat	<i>Redmi Note 10 5G</i>
<i>Processor</i>	MediaTek Dimensity 700 Dual 5G
<i>Memory</i>	4 GB
Penyimpanan	128GB
<i>Andoid Version</i>	11

Tabel 5. Perangkat Pengujian II Untuk Intensitas Cahaya

<i>Hardware</i>	<i>Spesifikasi</i>
Jenis Perangkat	<i>Smartphone Android</i>
Nama Perangkat	<i>Poco M3 Pro 5G</i>
<i>Processor</i>	MediaTek Dimensity 700 Dual 5G
<i>Memory</i>	6 GB
Penyimpanan	128GB
<i>Android Version</i>	11

b. Perangkat Lunak

Hal terpenting dalam mendukung kinerja suatu sistem adalah perangkat lunak. Dalam sebuah sistem, perangkat lunak digunakan untuk memberikan perintah ke perangkat keras, memungkinkan mereka untuk berkomunikasi satu sama lain. Diperlukan perangkat lunak tertentu yang dapat memungkinkan untuk memproduksi aplikasi *Augmented Reality*, seperti aplikasi pemrosesan 3D, dan perangkat lunak untuk membuat aplikasi *Augmented Reality*. Berikut adalah perangkat lunak yang dipakai dalam pembuatan aplikasi ini :

1. *UNITY* sebagai media pengembang.
2. *BLENDER 3D* sebagai pemodelan dan perancangan objek 3D.
3. Sistem Operasi *WINDOWS 10*.
4. *VISUAL STUDIO* sebagai media pengembang khususnya pembuatan kode.
5. *ADOBE PHOTOSHOP* pembuatan objek 2D.
6. *Light Meter* untuk pengukuran intensitas cahaya.

D. Teknik Pengujian Sistem

Pengujian sistem yang dilakukan untuk menguji aplikasi *Game* ini adalah metode pengujian *Black Box*, pengujian ahli dan pengujian pengguna. Metode pengujian *Black Box* digunakan untuk mengevaluasi sistem dari perspektif pengguna, dengan penekanan pada kinerja, persyaratan, dan antarmuka sistem, dari pada kode komputer yang ada. Tabel 5 di bawah ini mencantumkan desain

pengujian sistem yang akan diperiksa menggunakan metodologi pengujian *Black Box*.

Tabel 6. Rancangan Tabel Pengujian *BlackBox*

No	Komponen Yang Di Uji	Pengujian
1	Menembak	Tombol Tembak.
2	Bergerak	Tombol Gerak.
3	Scan Lingkungan	Scan lingkungan untuk <i>Augmented Reality</i> .
4	CutScene	CutScene yang menjeaskan sejarah 10 november sebelum permainan dimulai.
5	Pemilihan Tingkat Kesulitan	Pemilihan tingkat kesulitan dengan 3 tingkat kesulitan dari <i>easy, normal</i> dan <i>hard</i> .
6	Pemilihan Tank	Pemilihan <i>Tank</i> , dimana <i>player</i> akan memilih jenis <i>Tank</i> atau <i>skin Tank</i> yang diminati.
7	Mulai	Memulai permainan dan menampilkan menu tingkat kesulitan.
8	Keluar	Tombol untuk keluar aplikasi.

E. Teknik Analisis Data

Data yang diperoleh akan dievaluasi dan diinterpretasikan untuk diubah menjadi informasi. Prosesnya dimulai dengan memisahkan data yang diperoleh menjadi dua kelompok data, yaitu data kualitatif dan data kuantitatif. Data kualitatif diperoleh dari observasi dan dideskripsikan dengan kata-kata atau kalimat, dipisahkan menurut kategorinya untuk mendapatkan suatu kesimpulan.

Menurut Miles dan Huberman terdapat tiga model interaktif dalam analisis data yaitu (Prof. Dr. Sugiyono, 2013) :

1. Reduksi Data (*Data Reduction*)

Reduksi data meliputi meringkas, memfokuskan pada aspek yang paling signifikan yang berkaitan dengan masalah studi, mencari tema dan pola, dan akhirnya menciptakan gambaran yang lebih jelas dan memudahkan pengumpulan data lebih lanjut.

2. Penyajian Data (*Data Display*)

Tahap selanjutnya adalah menampilkan data yang telah direduksi. Tabel, grafik, flow chart, piktogram, dan representasi visual lainnya dari data dapat digunakan dalam penelitian kuantitatif. Materi disusun dan diletakkan dalam suatu pola hubungan melalui penyajian, sehingga lebih mudah untuk diinterpretasikan. Penyajian data dalam penelitian kualitatif dapat berbentuk penjelasan singkat, bagan, keterkaitan kategori, *flow chart*, dan alat bantu visual lainnya.

3. Penarikan Kesimpulan (*Conclusion Drawing/Verification*)

Menurut Miles dan Huberman, penarikan kesimpulan dan penilaian adalah fase ketiga dalam pemeriksaan data kualitatif. Hasil awal masih bersifat sementara, dan akan direvisi jika tidak ditemukan bukti yang signifikan untuk membenarkan langkah pengumpulan data selanjutnya. Namun, ketika peneliti mengumpulkan data di lapangan, temuan yang dicapai pada tahap awal didukung oleh bukti yang handal dan konsisten, kesimpulan yang dicapai dapat dipercaya.

Dalam penelitian kualitatif, kesimpulan adalah penemuan baru yang belum pernah ditemukan sebelumnya. Temuan dapat dinyatakan dalam bentuk atau gambaran suatu hal yang sebelumnya masih samar-samar atau hubungan yang gelap setelah penelitian menjadi nyata, hipotesis atau teori, sebab akibat atau interaksi.

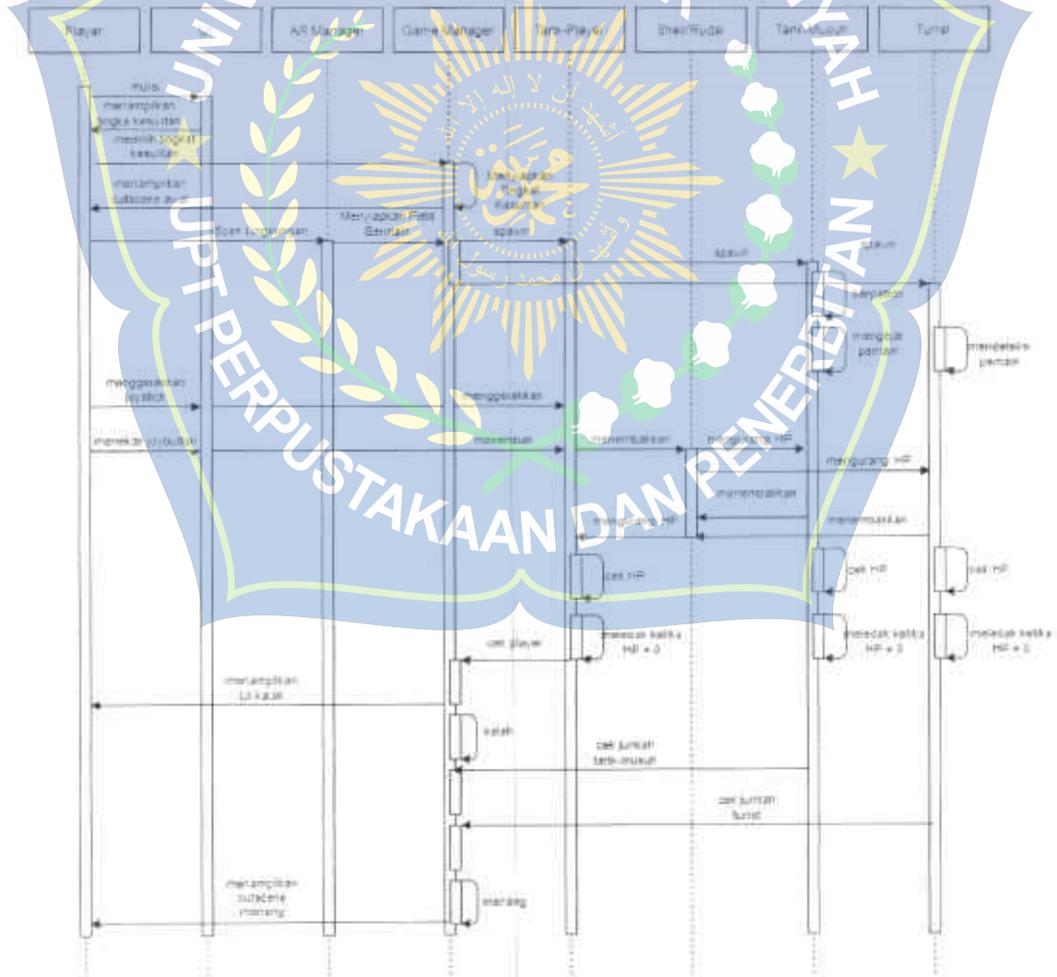
BAB IV

HASIL DAN PEMBAHASAN

A. Hasil

1. Sequence Diagram

Sequence Diagram pada pembuatan aplikasi Game ini merupakan interaksi berbagai objek Game, sistem Game dan pemain itu sendiri untuk melaksanakan fungsionalitas Game itu sendiri. Berikut adalah Sequence Diagram pada Game yang akan dibuat :

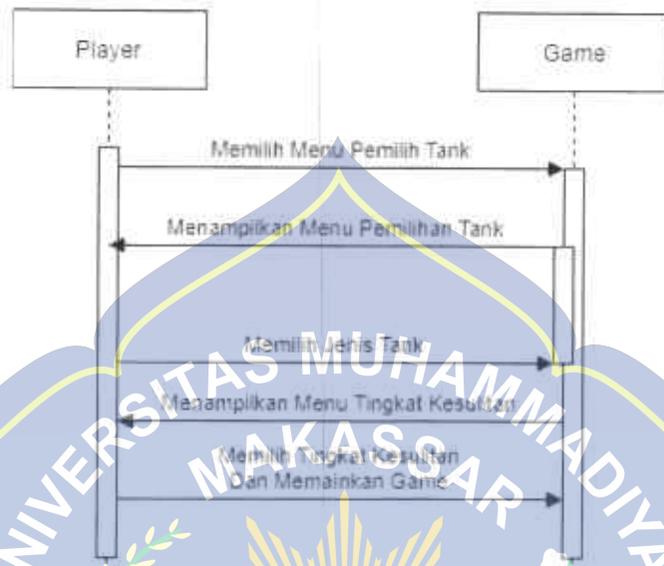


Gambar 11. Sequence Diagram Dalam Game

Pada awal permainan dibuka dari perangkat *Smartphone* maka akan menampilkan halaman utama. Kemudian *Player/ Pemain* akan menekan tombol mulai/*play* untuk memulai permainan. Setelah memilih *play*, *player* akan dihadapkan dengan pemilihan tingkat kesulitan *player* akan memilih salah satu dari tingkat kesulitan tersebut dan *Game manager* akan menyimpan dan menyipakan *Game* berdasarkan tingkat kesulitan yang *player* pilih. Setelah itu *Game manager* akan menampilkan tampilan *CutScene Intro* sebagai pembuka permainan. Setelah *CutScene Intro* selesai maka pemain akan mengScan lingkungan dan *AR Manager* akan menjadikannya sebagai *field* untuk bermain dan mengoperkan pesan untuk *Game manager* untuk *spawn* atau menampilkan musuh yang terdiri dari *Tank*, *Tank-musuh* dan *Turret*.

Setelah menampilkan *Tank-player*, *Tank-musuh* dan *Turret*, *Tank-musuh* akan memiliki fungsi berpatroli ketika *player* tidak ada disekitarnya dan mengejar pemain ketika *player* berada pada radius sekitar *Tank-musuh*. *Turret* Hanya memiliki fungsi mendeteksi pemain ketika berada pada radius tertentu. *Tank-player* bisa digerakkan dengan menggunakan *JoyStick* yang berada pada UI dan dapat menembakkan *Shell* dengan menekan tombol *Joybutton* yang ada pada UI juga. *Tank-Musuh* dan *Turret* juga memiliki fungsi tembak ketika *player* berada didekat mereka. *Shell* akan mengurangi HP musuh dan *player* itu sendiri. Ketika HP mereka mencapai titik 0 maka akan memanikan fungsi meledak dan *Game objek* tersebut akan menghilang dari *field Game*. *Game manager* akan mesatkan apakah *Tank-player*, *Tank-musuh* atau *Turret* masih ada dalam permainan atau tidak. Ketika *Tank-player* yang tidak ada, maka permainan akan selesai dan *player* meraih kekalahan. Ketika *Tank-musuh* dan *Turret* yang tidak ada maka *player* akan meraih kemenangan dan akan menampilkan *CutScene Outro/CutScene* menang.

Adapun *Sequence Diagram* untuk pemlihatn *Game* sebagai berikut :



Gambar 12. *Sequence Diagram* Pemilihan Tank

Pada awal membuka aplikasi permainan, *player* akan memilih menu pemilihan Tank. Sistem akan menampilkan menu pemilihan Tank yang dimana *player* dapat memilih jenis Tank yang diminati untuk bermain. Setelah memilih Tank yang akan dimainkan maka *player* akan langsung diarahkan ke menu pemilihan tingkat kesulitan dan disinilah *player* dapat memainkan Game dengan Tank yang dipilih.

2. Game Rules

Game "Thank's" merupakan edukasi dalam memperkenalkan sejarah kemerdekaan republic Indonesia yang dimainkan oleh satu pemain saja (*Single Player*). *Game Rules* pada Game ini terbilang cukup sederhana yaitu dimana pemain hanya enghalau dan mejatuhkan musuh dengan membuat HP (*Health Point*) musuh menjadi 0 dan HP pemain tidak boleh menjadi 0. Terdapat 3 tingkat kesulitan yang ada pada Game ini, berikut adalah rinciannya :

Tabel 7. Rincian Tingkat Kesulitan

<i>Easy</i>	
Tingkat Kesulitan	Mudah
Jumlah <i>Turret</i>	1

Jumlah <i>Tank</i>	1
Normal	
Tingkat Kesulitan	Menengah
Jumlah <i>Turret</i>	2
Jumlah <i>Tank</i>	1
Hard	
Tingkat Kesulitan	Sulit
Jumlah <i>Turret</i>	3
Jumlah <i>Tank</i>	2

3. *Storyboard*

Storyboard adalah sarana untuk menyampaikan plot dari sebuah aplikasi. Tujuan penulis memanfaatkan pendekatan *Storyboard* adalah untuk menjelaskan kepada pengguna tentang konten aplikasi ini dan cara kerjanya. *Storyboard* digunakan sebagai panduan oleh *Developer* agar proses perancangan dan pengembangan aplikasi ini agar tetap operasional sehubungan dengan apa yang diinginkan *Developer*. Berikut adalah *Storyboard* yang akan digunakan didalam *Game* (Wibawa et al., 2021).

a. *Scene* Menu Utama

Menu utama berisi pilhan menu yang meliputi mulai (*play*) dan keluar (*quit*). Mulai untuk memulai permainan dan keluar untuk keluar dari aplikasi. Pada menu ini memiliki *background Game* dengan *background* langit berwarna biru serta pohon kelapa dan reruntuhan rumah. Menu utama juga memiliki tulisan dari judul *Game* yaitu "*Thank's*" yang dimana huruf T,H dan A memiliki warna merah dan N,K,' dan S memiliki warna putih. Visual dari *Scene* menu utama dapat dilihat pada gambar berikut ini :



Gambar 13. *Scene* Menu Utama

b. *Scene* Pemilihan Tingkat Kesulitan

Menu pemilihan tingkat kesulitan berisi pilhan menu yang meliputi *easy*, *normal* dan *hard*. Pemain harus memilih salah satunya agar dapat menuju *Scene* selanjutnya. *Background* menu ini memiliki warna biru kehitaman yang dimana terdapat 3 tombol yang mewakili masing-masing tingkat keulitan yaitu *easy* berwarna hijau, *normal* berwarna biru dan *hard* berwarna merah dan semuanya memiliki warna font putih. Visual dari *Scene* pemilihan tingkat kesulitan dapat dilihat pada gambar berikut ini :



Gambar 14. *Scene* Pemilihan Tingkat Kesulitan

c. *Scene* Menampilkan *CutScene Intro*

Scene ini adalah *CutScene* yang akan menceritakan peperangan 10 November 1945 di Surabaya dan mengapa peperangan tersebut bisa terjadi. *Scene* ini terdiri dari berbagai gambar dan teks yang digabung. Salah satu visual dari *Scene* ini dapat dilihat pada gambar berikut ini :



Gambar 15. Scene Menampilkan CutScene Intro

d. Scene Scan Lingkungan

Scene ini merupakan Scene dimana pemain mengScan lingkungan mereka untuk dijadikan field dalam Game. Field didalam Game ditandai dengan terbentuknya titik-titik putih yang menandakan bahwa lingkungan tersebut sudah dapat dimainkan. Visual dari Scene ini dapat dilihat pada gambar berikut ini :



Gambar 16. Scene Scan Lingkungan

e. Scene Utama Game

Scene ini merupakan Scene utama yang dimana pemain berperan sebagai Game untuk menjatuhkan musuh. Latar tempat permainan ini merupakan hasil dari Scan lingkungan sekitar oleh pemain. Visual dari Scene ini dapat dilihat pada gambar berikut ini :



Gambar 17. *Scene* Utama Game

f. *Scene* Menang

Scene ini akan muncul ketika pemain berhasil menyelesaikan permainan dan meraih kemenangan. *Scene* ini akan menampilkan kata yang bertuliskan “MENANG” yang berwarna hijau dan memiliki 2 buah tombol yang digunakan untuk kembali ke menu utama atau keluar dari aplikasi. Visual dari *Scene* ini dapat dilihat pada gambar berikut ini :



Gambar 18. *Scene* Menang

g. *Scene* Manampilkan *CutScene* *Outro*

Scene ini akan muncul ketika pemain berhasil menyelesaikan permainan dan meraih kemenangan. *Scene* ini terdiri dari berbagai gambar dan teks yang digabung. Salah satu visual dari *Scene* ini dapat dilihat pada gambar berikut ini :



Gambar 19. Scene Menampilkan CutScene Outro

h. Scene Kalah

Scene ini akan muncul ketika pemain gagal dan membuat HP Game pemain menjadi 0. Scene ini akan menampilkan kata yang bertuliskan “KALAH” yang berwarna merah dan memiliki 2 buah tombol yang digunakan untuk kembali ke menu utama atau keluar dari aplikasi. Visual dari Scene ini dapat dilihat pada gambar berikut ini :



Gambar 20. Scene Kalah

4. Perancangan Komponen Permainan

Perancangan komponen permainan meliputi deskripsi karakter yaitu *Tank* yang dimainkan oleh pemain serta *Tank*-musuh beserta *Turre*-nya yang akan muncul pada aplikasi *Game* yang dibangun.

a. Deskripsi Karakter *Tank* Pemain

Karakter utama dalam permainan ini adalah *Tank* yang dikendalikan oleh pemain. *Tank* ini dikisahkan sebagai *Tank* sitaan dari tantara jepang saat penyerahan Gudang senjata setelah jepang kalah dan menyerahkannya ke Indonesia. *Game* ini memiliki tugas untuk melindungi rakyat dengan

menghalau dan menjatuhkan tantara inggris. Desain *Tank* bisa dilihat pada gambar dibawah ini :



Gambar 21. Desain *Tank*-Pemain

b. Deskripsi Karakter *Tank* Musuh

Salah satu musuh yang pemain akan hadapi yaitu *Tank* dari tantara inggris. Desain *Tank* ini bisa di lihat pada gambar dibawah ini :



Gambar 22. Desain *Tank*-Enemy

c. Deskripsi Karakter *Turret* Musuh

Musuh selanjutnya yang pemain akan hadapi yaitu *Turret* dari tantara inggris. Desain *Turret* ini bisa di lihat pada gambar dibawah ini :



Gambar 23. Desain *Turret*

d. Deskripsi *Shell/Rudal*

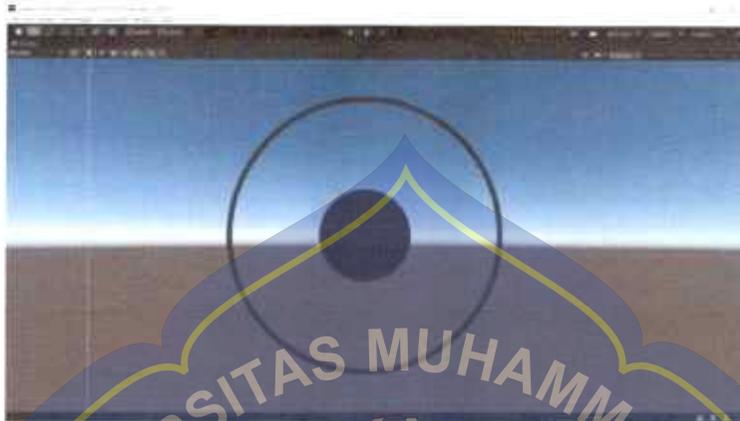
Shell atau Rudal digunakan didalam *Game* ini sebagai pemberi kerusakan (*demage*) kepada entitas atau objek dalam *Game*. Mekanik dari *Shell* ini ialah Ketika ia bertemu dengan rigidbody dari object *Game* maka *Shell* akan meledak dan memberikan kerusakan kepada pemain atau musuh, Kerusakan tersebut akan mengurangi HP pemain atau musuh. Desain *Shell* ini bisa di lihat pada gambar dibawah ini :



Gambar 24. Gambar *Shell*

e. *JoyStick*

JoyStick digunakan untuk menggerakkan *Tank* pemain kearah yang diinginkan. Desain *JoyStick* ini bisa di lihat pada gambar dibawah ini :



Gambar 25. Desain *JoyStick*

f. *Joybutton*

Joybutton digunakan untuk menembakkan *Shell*. Desain *JoyStick* ini bisa di lihat pada gambar dibawah ini :



Gambar 26. Desain *Joybutton*

g. *UI Health Point (HP)*

UI HP digunakan untuk mengindikasikan jumlah *HP* yang ada dalam objek *Game* seperti *Tank*, *Turret*, dan *Tank-musuh*.



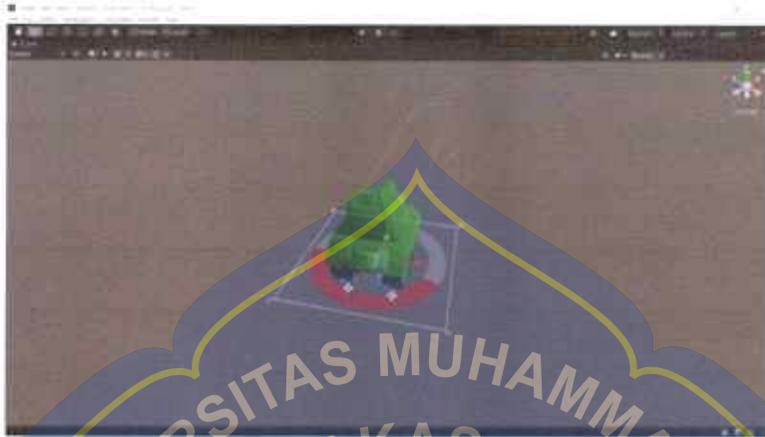
Gambar 27. Desain UI HP

Desain dari UI HP ini adalah sebuah lingkaran yang dimana lingkaran ini memiliki lubang ditengahnya dan garis pinggir lingkaran inilah yang akan menjadi tolak ukur dari HP pemain atau musuh.



Gambar 28. Desain HP UI Ketika HP Penuh

Desain HP UI ketika HP penuh memiliki warna hijau dan melekat pada bagian bawah objek *Game* baik itu untuk *Tank*-pemain, *Tank*-musuh dan *Turret*.



Gambar 29. Desain *HP UI* Ketika *HP* Berkurang

Desain *HP UI* ketika *HP* berkurang memiliki warna merah dan melekat pada bagian bawah objek *Game* baik itu untuk *Tank*-pemain, *Tank*-musuh dan *Turret*.

h. Algoritma Jarak Tembakan *Shell/Rudal*



Gambar 30. Letak Jatuh Awal Rudal

Untuk menghitung jarak atau letak jatuhnya rudal maka digunakan rumus sebagai berikut :

$$P = V * Gaya$$

Keterangan :

P : Posisi yang Akan dituju Rudal.

V : Kecepatan yang ada yang dimana berbentuk matriks dengan posisi x,y,z.

Gaya : Gaya dorong yang doberikan oleh rudal.

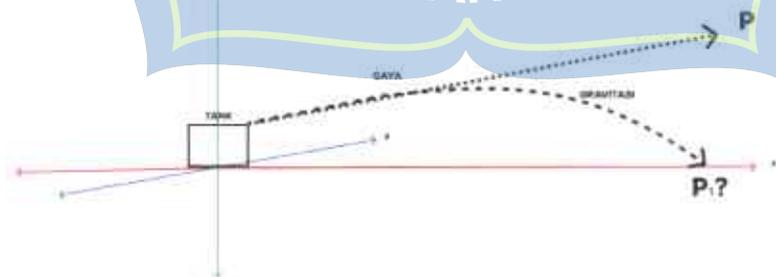
Untuk menghitungnya misalkan $V = \{0x, 1 y, 2z\}$ dan gaya yang diberikan adalah 5 maka dapat dihitung P atau letak rudal akan berjalan sebagai berikut :

$$P = V * \text{Gaya}$$

$$P = \{0,0x, 1 y, 2z\} * 5$$

$$P = \{0x, 5y, 10z,\}$$

Maka tujuan rudal yaitu berada di koordinat $\{0x, 5y, 10z,\}$. Akan tetapi tujuan rudalnya tidak berhenti karena belum mengimplementasikan gravitasi dari system unity. Ketika mengaktifkan system gravitasi pada *rigidbody* rudal di unity maka rudal akan mengalami perubahan sehingga gerak jatunya menjadi gerak jatuh parabola yang dapa dilihat pada gambar berikut :



Gambar 31. Letak Jatuh Akhir Rudal

Untuk menghitung jarak atau letak jatuhnya rudal dikarenakan gravitasi maka digunakan rumus sebagai berikut :

$$P_1 = V + A * \text{Gaya}$$

Ket :

P_t : Posisi yang Akan dituju Rudal.

V : Kecepatan yang ada yang dimana berbentuk matriks dengan posisi x, y, z .

A : Akselerasi akibat gravitasi yang bernilai sumbu sudut $y = - 9,8$ atau bisa dilihat dengan koordinat $\{0x, -9,8y, 0z\}$.

Gaya : Gaya dorong yang diberikan oleh rudal.

Untuk menghitungnya misalkan $V = \{0x, 1 y, 2z\}$ dan gaya yang diberikan adalah 5 maka dapat dihitung P atau letak rudal akan berjalan

$$P_t = V + A * \text{Gaya}$$

$$P_t = \{0x, 1 y, 2z\} + \{0x, -9,8y, 0z\} * 5$$

$$P_t = \{0x, -48y, 10z, \}$$

i. Algoritma Pemberi Kerusakan Akibat *Shell/Rudal*

Untuk menghitung kerusakan yang diakibatkan rudal iyalah dengan mengecek apakah rudal bertumbukan dengan *rigidbody* baik itu *Tank-musuh*, *Turret* atau *rigidbody* yang ada pada lantai sehingga mengakibatkan *collision* atau tumbukan antara kedua objek tersebut. Ketika bertumbukan maka akan dihitung dengan cara berikut :

1) Menghitung Jarak Antara Jatunya Rudal dan Target

Untuk menghitung jarak antara rudal dan posisi target ialah dengan menggunakan rumus :

$$\text{Jarak ke Target} = \text{Posisi Target} - \text{Posisi Rudal Meledak}$$

Misalkan posisi target adalah $\{ 1x, 2y, 3z \}$ dan posisi rudal meledak adalah $\{ 1.5x, 2y, 4z \}$ Maka dapat dihitung :

$$\text{Jarak ke Target} = \{ 1x, 2y, 3z \} - \{ 1.5x, 2y, 4z \}$$

$$\text{Jarak ke Target} = \{ -0.5x, 0y, -1z \}$$

2) Menghitung Jarak Ledakan

Untuk menghitung jarak ledakan maka digunakan rumus sebagai berikut :

$$\begin{aligned} \text{Jarak Ledakan} &= \sqrt{(\text{Jarak ke Target})^2} \\ \text{Jarak Ledakan} &= \sqrt{\{x^2 + y^2 + z^2\}} \end{aligned}$$

Dari perhitungan sebelumnya dapat dihitung jarak ledakan ialah sebagai berikut:

$$\text{Jarak Ledakan} = \sqrt{\{x^2 + y^2 + z^2\}}$$

$$\text{Jarak Ledakan} = \sqrt{\{-0,5^2 + 0^2 + -1^2\}}$$

$$\text{Jarak Ledakan} = \sqrt{\{0,25 + 1 + 1\}}$$

$$\text{Jarak Ledakan} = \sqrt{2,25} = 1,5$$

3) Menghitung Jarak Relatif Ledakan Rudal

Untuk menghitung jarak relatif ledakan dapat menggunakan rumus sebagai berikut :

$$\text{Jarak Relatif} = (\text{Radius Ledakan} - \text{Jarak Ledakan}) / \text{Radius Ledakan}$$

Untuk menghitung jarak relative maka ditentukan pertama kali yaitu berapa radius ledakan yang dimana radius ledakan ditentukan oleh *developer* itu sendiri. Misalnya radius ledakan adalah 2 maka dapat dihitung :

$$\text{Jarak Relatif} = (\text{Radius Ledakan} - \text{Jarak Ledakan}) / \text{Radius Ledakan}$$

$$\text{Jarak Relatif} = (2 - 1,5) / 2$$

$$\text{Jarak Relatif} = 0,25$$

4) Memberikan Kerusakan

Untuk menghitung kerusakan akibat rudal dapat menggunakan rumus sebagai berikut :

$$\text{Kerusakan} = \text{Jarak Relatif} * \text{Kerusakan Max}$$

Untuk menghitung kerusakan akibat rudal maka ditentukan pertama kali yaitu berapa kerusakan maksimum yang dimana ditentukan oleh

developer itu sendiri. Misalnya kerusakan maksimum adalah 20 maka dapat dihitung :

Kerusakan = Jarak Relatif * Kerusakan Max

Kerusakan = 0,25 * 20

Kerusakan = 5

Untuk kerusakan yang dihasilkan yaitu bernilai 5 sehingga HP musuh akan berkurang 5 point.

5) Mengecek Apakah Terjadi Kerusakan Atau Tidak

Untuk mengecek apakah terjadi kerusakan atau tidak maka menggunakan fungsi dari unity itu sendiri yaitu :

```
kerusakan = Mathf.Max(0f, kerusakan);
```

Fungsi ini bertujuan agar tidak ada kerusakan negative dari hasil perhitungan yang dimana fungsi ini akan memilih angka terbesar antara 0 dan kerusakan yang ditimbulkan. Ketika kerusakanya bernilai positif maka itu terhitung sebagai *hit*, kerusakan yang dapat mengakibatkan HP musuh berkurang sesuai kerusakan yang telah dikalkulasi dan apabila negative maka kerusakan yang akan dikeluarkan 0 point atau dihitung sebagai *miss*.

5. Perancangan Antarmuka

a. Perancangan Antarmuka Menu Utama

Menu utama adalah menu yang berisi sejumlah pilihan menu yang dapat dipilih pengguna untuk menjalankan aplikasi yang berupa menu mulai dan keluar. Antarmuka menu utama digambarkan dalam Gambar di bawah ini :



Gambar 32. Antarmuka Menu Utama

b. Perancangan Antarmuka Menu Pemilihan Tingkat Kesulitan

Menu pemilihan tingkat kesulitan adalah menu yang berisi sejumlah pilihan tingkat kesulitan yang dapat dipilih pengguna untuk memainkan *Game* yang berupa *easy*, *normal* dan *hard*. Antarmuka pemilihan tingkat kesulitan digambarkan dalam Gambar di bawah ini.



Gambar 33. Antarmuka Menu Pemilihan Tingkat Kesulitan

c. Perancangan Antarmuka Pemilihan Tank

Antarmuka pemilihan *Tank* berupa menu yang menampilkan *Tank* dan dua buah tombol untuk memilih jenis *Tank* selanjutnya dan sebelumnya (*next* dan *previus*). Ketika ingin memilih *Tank* maka tinggal menekan tombol pilih yang berada di atas *Tank*. Antarmuka ini dapat dilihat dalam gambar dibawah ini:



Gambar 34. Antarmuka Pemilihan Tank

d. Perancangan Antarmuka Dalam Game

Antarmuka didalam *Game* ini berupa tampilan *Game* beserta 2 tombol yaitu tombol bagian kiri untuk menggerakkan *Tank* dan tombol bagian kanan untuk menembakkan peluru. Antarmuka ini dapat dilihat dalam gambar di bawah ini:



Gambar 35. Antarmuka Dalam Game

e. Perancangan Antarmuka Menang

Antarmuka menang adalah menu yang berisi pemberitahuan Ketika pemain telah memenangkan permainan. Antarmuka menang ini dapat dilihat dalam Gambar di bawah ini.



Gambar 36. Antarmuka Menang

f. Perancangan Antarmuka Kalah

Antarmuka kalah adalah menu yang berisi pemberitahuan ketika HP pemain telah menjadi 0 dan mengalami kekalahan. Antarmuka menang ini dapat dilihat dalam Gambar di bawah ini.



Gambar 37. Antarmuka Kalah

6. Perancangan *CutScene*

CutScene merupakan adegan sisipan atau peristiwa dalam suatu *Game* yang mengganggu jalannya sebuah *Game* dan tidak interaktif. Adegan yang digunakan seperti percapakan sebuah karakter, atau penjelasan suatu peristiwa yang ada dalam suatu *Game* atau permainan.

a. *CutScene Intro*

CutScene Intro merupakan *CutScene* yang digunakan sebelum permainan dimulai. Adapun beberapa adegan yang akan ditampilkan dalam *CutScene* ini adalah :

- 1) Pengeboman Hiroshima pada tanggal 6 Agustus dan Nagasaki pada tanggal 9 Agustus. Pengakuan kekalahan Jepang pada 14 Agustus 1945 pada waktu Jepang.



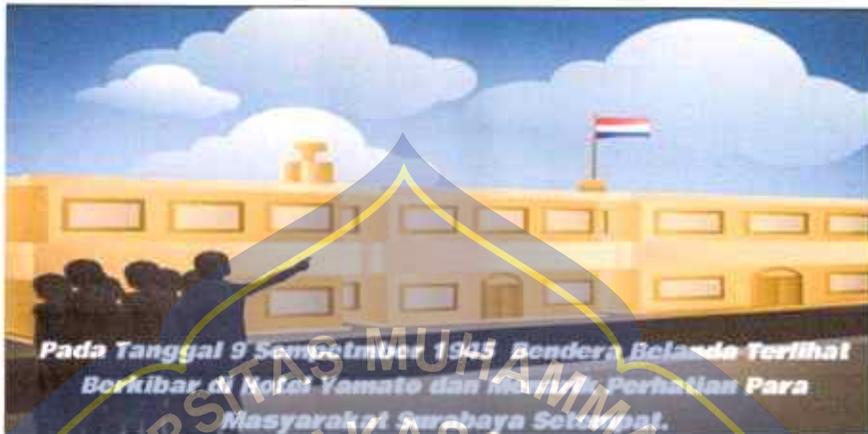
Gambar 38. *CutScene Intro I*

- 2) Penyerahan gudang senjata pihak Jepang sehingga memperkuat militer pihak Indonesia.



Gambar 39. *CutScene Intro II*

- 3) Pengibaran bendera Belanda di hotel Yamato pada tanggal 9 September 1945.



Gambar 40. CutScene Intro III

- 4) Amarah pemuda dan perobekan bendera Belanda.



Gambar 41. CutScene Intro IV

- 5) Kedatangan tentara sekutu yang dimana warga setempat merasa curiga.



Gambar 42. CutScene Intro V

- 6) Munculnya konflik antara warga dan tantara sekutu dan tewasnya Brigadir Jendral Mallaby.



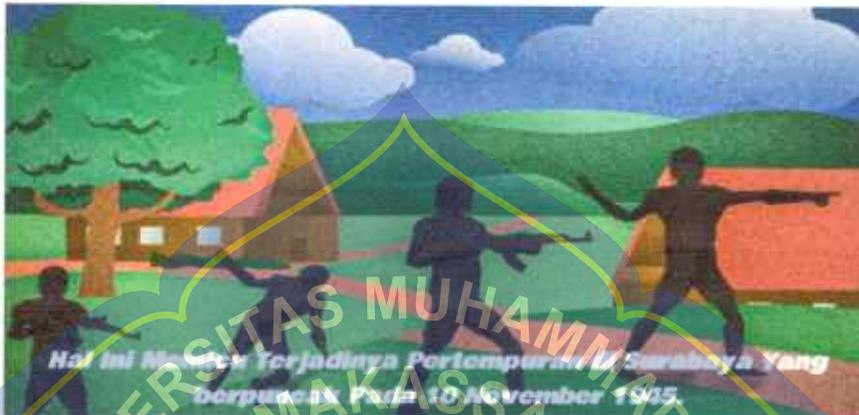
Gambar 43. *CutScene Intro VI*

- 7) Keluarnya ultimatum dari pihak sekutu dan membuat marah warga setempat.



Gambar 44. *CutScene Intro VII*

8) Terjadinya pertempuran dan berpuncak pada 10 november 1945.



Gambar 45. *CutScene Intro VII*

b. *CutScene Outro*

CutScene Outro merupakan adegan yang digunakan ketika pemain meraih kemenangan dalam menyelesaikan permainan. Adapun beberapa adegan yang akan ditampilkan dalam *CutScene* ini adalah :

- 1) Pertempuran merenggut banyak nyawa baik Pihak Inggris maupun Indonesia



Gambar 46. *CutScene Outro I*

- 2) Pada akhirnya Surabaya dikuasai oleh pihak Inggris dan diserahkan kepada pihak Belanda.



Gambar 47. *CutScene Outro II*

- 3) Dinamika perpolitikan antara Indonesia-Belanda-Inggris ditentukan dalam pertempuran ini.



Gambar 48. *CutScene Outro III*

- 4) Inggris makin enggan untuk menerima beban bekas kolonialisme Belanda dan cenderung kopertif dengan Indonesia.



Gambar 49. CutScene Outro IV

- 5) Pertempuran ini pula yang menjadi cikal bakal lahirnya hari pahlawan yang dirayakan setiap 10 November.



Gambar 50. CutScene Outro V

B. Pengujian

1. Pengujian Aplikasi

Untuk memastikan bahwa *Game* dapat berjalan secara fungsional dan berjalan baik sesuai dengan keinginan pembuat atau pengembang maka akan dilakukan proses pengujian pada *Game*. Pengujian pada *Game* ini menggunakan metode *black box testing* dan di uji pada perangkat *Developer* untuk memastikan apakah aplikasi dapat berjalan dengan baik atau tidak.

a. Pengujian Komponen *Game*

1) Pengujian Komponen Mulai

Tabel 8. Pengujian Komponen Mulai

No	Skenario pengujian	Test case	Hasil Yang di harapkan	Hasil
1	Komponen Mulai	Klik Mulai Pada Aplikasi	Game Terbuka dan menampilkan pemilihan tingkat kesulitan,	Valid

Pengujian dilakukan dengan menekan tombol *play* pada saat berada di menu utama dalam *Game*. Hasil dari pengujian ini yaitu ditampikannya menu pemilihan tingkat kesulitan.

2) Pengujian Komponen Keluar

Tabel 9. Pengujian Komponen Keluar

No	Skenario pengujian	Test case	Hasil Yang di harapkan	Hasil
1	Komponen Keluar	Klik UI keluar pada aplikasi	Aplikasi <i>Game</i> akan mengarahkan apakah anda ingin keluar atau tidak.	Valid
2	Komponen Keluar Dan yang dipiliih Ya	Klik UI ingin keluar Ya	Aplikasi <i>Game</i> akan tertutup.	Valid
3	Komponen Keluar Dan yang dipiliih Tidak	Klik UI ingin keluar Tidak	Aplikasi <i>Game</i> akan Kembali ke menu utama	Valid

Pengujian dilakukan dengan menekan tombol keluar/*quit* pada menu utama. Hasil dari pengujian ini akan menampilkan menu apakah pemain yakin ingin keluar dari *Game* atau tidak. Ketika yang dipilih

adalah “Ya”, maka *Game* akan menutup sendiri. Ketika yang dipilih adalah “Tidak” maka akan kembali ke menu utama.

3) Pengujian Komponen Tingkat Kesulitan

Tabel 10. Pengujian Komponen Tingkat Kesulitan

No	Skenario pengujian	Test case	Hasil Yang di harapkan	Hasil
1	Komponen Tingkat Kesulitan	Klik UI Pemilihan tingkat kesulitan	Aplikasi akan mengaharkan kedalam <i>Game</i> dengan tingkat kesulitan yang dipilih antara <i>Easy</i> , <i>Normal</i> , atau <i>Hard</i>	Valid
2	Komponen Pemilihan tingkat kesulitan <i>Easy</i>	Pemilihan tingkat kesulitan <i>Easy</i>	Aplikasi <i>Game</i> akan menerapkan tingkat kesulitan <i>Easy</i> dengan Spawn 1 <i>Turret</i> dan 1 <i>Tank</i>	Valid
3	Komponen Pemilihan tingkat kesulitan <i>Normal</i>	Pemilihan tingkat kesulitan <i>Normal</i>	Aplikasi <i>Game</i> akan menerapkan tingkat kesulitan <i>Easy</i> dengan Spawn 2 <i>Turret</i> dan 1 <i>Tank</i>	Valid
4	Komponen Pemilihan tingkat kesulitan <i>Hard</i>	Pemilihan tingkat kesulitan <i>Hard</i>	Aplikasi <i>Game</i> akan menerapkan tingkat kesulitan <i>Easy</i> dengan Spawn 3 <i>Turret</i> dan 2 <i>Tank</i>	Valid

Pengujian dilakukan dengan cara memilih tombol pada tingkat kesulitan yang terdiri dari tombol *easy*, *normal* dan *hard*. Hasil yang dicapai dari pengujian ini adalah ketika tombol *easy* yang dipilih, maka jumlah *Game*-musuh yang akan ditampilkan berjumlah 1 dan *Turret* berjumlah 1 pula. Ketika tombol *normal* yang dipilih, maka jumlah *Game*-musuh yang akan ditampilkan berjumlah 1 dan *Turret* berjumlah 2. Ketika tombol *hard* yang dipilih, maka jumlah *Game*-musuh yang akan ditampilkan berjumlah 2 dan *Turret* berjumlah 3. Hasil tersebut sesuai dengan yang *Developer* harapkan.

4) Pengujian Komponen *Scan* Lingkungan

Tabel 11. Pengujian Komponen *Scan* Lingkungan

No	Skenario pengujian	<i>Test case</i>	Hasil Yang di harapkan	Hasil
1	Komponen <i>Scan</i> Lingkungan	<i>Scan</i> Lingkungan akan otomatis Ketika <i>Game</i> dimainkan	<i>Game</i> akan meng <i>Scan</i> lingkungan dan menerapkannya menjadi field bermain.	Valid

Pengujian dilakukan dengan mengscan lingkungan saat *Game* akan dimulai. Hasil dari pengujian ini adalah diperlihatkannya titik-titik putih yang menjadi penanda bahwa lingkungan telah ter*Scan*. Hasil tersebut sesuai dengan yang *Developer* harapkan.

5) Pengujian Komponen *CutScene*

Tabel 12. Pengujian Komponen *CutScene*

No	Skenario pengujian	Test case	Hasil Yang di harapkan	Hasil
1	Komponen <i>CutScene Intro</i>	<i>CutScene</i> akan mulai ketika telah memilih tingkat kesulitan yang diinginkan dan akan memulai permainan	Memperlihatkan <i>CutScene</i> secara menyeluruh.	Valid
2	Komponen <i>CutScene Outro</i>	<i>CutScene</i> akan mulai ketika telah meraih kemenangan.	Memperlihatkan <i>CutScene</i> secara menyeluruh.	Valid

Pengujian komponen *CutScene Intro* ini dilakukan dengan cara memainkan *Game* pada saat setelah menekan salah satu tombol tingkat kesulitan. Hasilnya menampilkan *CutScene Intro* yang dimana akan menampilkan sejarah bagaimana peperangan 10 november terjadi. Untuk pengujian *CutScene Outro* dilakukan saat pemain meraih kemenangan. Hasilnya menampilkan *CutScene Outro* yang dimana

akan menampilkan hasil dari peperangan 10 november tersebut. Hasil tersebut sesuai dengan yang *Developer* harapkan.

6) Pengujian Komponen Pemilihan *Tank*

Tabel 13. Pengujian Komponen Pemilihan *Tank*

No	Skenario pengujian	Test case	Hasil Yang di harapkan	Hasil
1	Komponen Pemilihan <i>Tank</i>	Memilih Jenis <i>Tank</i> yang diinginkan. Yaitu antara <i>Tank</i> jenis 1 – 4.	Akan Menampilkan <i>Tank</i> yang dipilih ke dalam <i>Game</i> .	Valid

Pengujian komponen ini dilakukan saat *player* memilih menu pilih *Game* saat di menu utama. *Player* akan memilih satu dari 4 jenis *Tank* yang disediakan sesuai dengan keinginan *player*. Hasil dari pengujian ini adalah sistem akan menampilkan jenis *Tank* yang dipilih *player* tersebut kedalam *Tank* pada saat dimainkan.

7) Pengujian Komponen Bergerak

Tabel 14. Pengujian Komponen Bergerak

No	Skenario pengujian	Test case	Hasil Yang di harapkan	Hasil
1	Komponen Bergerak	Menggerakkan <i>JoyStick</i> pada <i>UI Game</i>	<i>Game</i> yang dikendalikan <i>player</i> akan bergerak.	Valid

Pengujian komponen ini dilakukan saat *Game* sedang berjalan dengan cara menggerakkan *JoyStick*. Hasil dari pengujian ini adalah

Tank-pemain dapat berjalan kearah yang sesuai dengan *JoyStick* arahkan. Hasil tersebut sesuai dengan yang *Developer* harapkan.

8) Pengujian Komponen Menembak

Tabel 15. Pengujian Komponen Menembak

No	Skenario pengujian	Test case	Hasil Yang di harapkan	Hasil
1	Komponen Menembak	Klik <i>Joybutton</i> pada UI <i>Game</i>	<i>Game</i> yang dikendalikan player akan menembakkan <i>Shell</i> atau rudal.	Valid

Pengujian komponen ini dilakukan saat *Game* sedang berjalan dengan cara menekan *Joybutton*. Hasil dari pengujian ini adalah *Tank*-pemain dapat menembakkan *Shell*. Hasil tersebut sesuai dengan yang *Developer* harapkan.

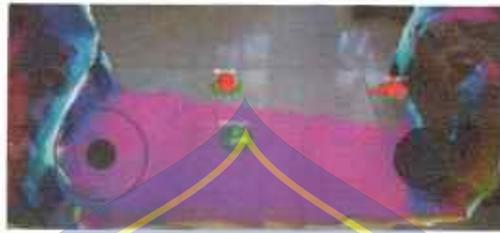
b. Hasil Pengujian Perbandingan Intensitas Cahaya

Pengujian intensitas cahaya dilakukan agar mengetahui dengan insentitas cahaya manakah aplikasi dapat berjalan dengan baik dan lancar. Pengujian ini terdiri dari pengujian *Indoor* dan *Outdoor*. Dan dilakukan di dua perangkat yang berbeda yaitu *Redmi Note 10 5G* dan *Poco M3 Pro 5G*.

Tabel 16. Tabel Pengujian Intensitas Cahaya *Indoor Redmi Note 10 5G*

Intensitas Cahaya <i>Indoor</i> (lux)	Gambar Hasil Pengujian	Kesimpulan Pegujian
Intensitas Cahaya 0-50 lux		Tidak Bekerja

Intensitas
Cahaya 51 –
200 lux



Cukup Bekerja

Intensitas
Cahaya >
200 lux



Bekerja
Dengan Baik

Dari pengujian intensitas cahaya *Indoor* dengan perangkat *Redmi Note 10 5G* tersebut dapat disimpulkan bahwa system AR pada *Game* dapat bekerja bila memiliki intensitas cahaya 51 lux keatas. Bila mendapatkan intensitas cahaya yang di bawah 51 lux sistem AR *Game* tidak dapat bekerja.

Tabel 17. Tabel Pengujian Intensitas Cahaya *Outdoor* *Redmi Note 10 5G*

Intensitas Cahaya <i>Outdoor</i>	Gambar Hasil Pengujian	Kesimpulan Pegujian
Intensitas Cahaya 0-50 lux		Tidak Bekerja
Intensitas Cahaya 51 – 200 lux		Cukup Baik

Intensitas
Cahaya >
200 lux



Bekerja
Dengan Baik

Dari pengujian intensitas cahaya *Outdoor* tersebut dengan perangkat *Redmi Note 10 5G* dapat disimpulkan bahwa sistem AR pada *Game* dapat bekerja bila memiliki intensitas cahaya 51 lux keatas. Bila mendapatkan intensitas cahaya yang di bawah 51 lux sistem AR *Game* tidak dapat bekerja.

Tabel 18. Tabel Pengujian Intensitas Cahaya *Indoor Poco M3 Pro 5G*

Intensitas Cahaya <i>Indoor</i> (lux)	Gambar Hasil Pengujian	Kesimpulan Pengujian
Intensitas Cahaya 0-50 lux	A screenshot from an AR game showing a dark blue background with some faint, illegible text or graphics.	Tidak Bekerja
Intensitas Cahaya 51 - 200 lux	A screenshot from an AR game showing a pinkish-purple ground with some green and red objects. A black circle is visible on the right side of the screen.	Cukup Bekerja
Intensitas Cahaya > 200 lux	A screenshot from an AR game showing a pinkish-purple ground with some green and red objects. A black circle is visible on the right side of the screen.	Bekerja Dengan Baik

Dari pengujian intensitas cahaya *Indoor* tersebut dapat disimpulkan bahwa system AR pada *Game* dengan perangkat *Poco M3 Pro 5G* dapat

bekerja bila memiliki intensitas cahaya 51 *lux* keatas. Bila mendapatkan intensitas cahaya yang dibaah 51 *lux* sistem AR *Game* tidak dapat bekerja.

Tabel 19. Tabel Pengujian Intensitas Cahaya *Outdoor Poco M3 Pro 5G*

Intensitas Cahaya <i>Outdoor</i> (<i>lux</i>)	Gambar Hasil Pengujian	Kesimpulan Pegujian
Intensitas Cahaya 0-50 <i>lux</i>		Tidak Bekerja
Intensitas Cahaya 51 – 200 <i>lux</i>		Cukup Bekerja
Intensitas Cahaya > 200 <i>lux</i>		Bekerja Dengan Baik

Dari pengujian intensitas cahaya *Outdoor* tersebut dapat disimpulkan bahwa sistem AR pada *Game* dengan perangkat *Poco M3 Pro 5G* dapat bekerja bila memiliki intensitas cahaya 51 *lux* keatas. Bila mendapatkan intensitas cahaya yang dibaah 51 *lux* sistem AR *Game* tidak dapat bekerja.

c. Hasil Pengujian Dan Respon Dari Siswa

Pengujian dilakukan di SMKN 2 Selayar yang dimana aplikasi akan diujikan ke beberapa siswa SMK. Total partisipan siswa yang menguji dan mencoba aplikasi ini ialah 40 siswa dan siswi yang dimana terdiri dari kelas

XII (dua belas) dan XI (sebelas). Berikut adalah daftar pertanyaan dan hasil dari pengujian dan respon dari total 40 siswa dan siswi tersebut :

1) Bagaimana Menurut Anda Aplikasi Game Ini.

Data yang diperoleh dari 40 responden siswa dan siswi mengenai seberapa bagusnya aplikasi *Game* yang diuji terdapat 15 siswa dan siswi (37,5%) yang memilih sangat bagus dan 25 siswa dan siswi (62,5%) yang memilih bagus, buruk dan sangat buruk memiliki 0 responden. Tabel dan gambar responden bisa dilihat dibawah ini :

Tabel 20. Hasil Responden Dari Pertanyaan I

Pertanyaan	Sangat Bagus	Bagus	Buruk	Sangat Buruk
Bagaimana Menurut Anda Aplikasi Game Ini	15	25	0	0

Bagaimana Menurut Anda Aplikasi Game Ini :
40 jawaban



Gambar 51. Hasil Responden Dari Pertanyaan I

2) Apakah Anda Mendapatkan Pembelajaran dari Memainkan Game Ini.

Data yang diperoleh dari 40 responden siswa dan siswi mengenai apakah siswa dan siswi atau responden mendapatkan pembelajaran dari aplikasi *Game* yang diuji terdapat 38 siswa (95%) yang memilih sangat

ya dan 2 siswa (5%) yang memilih tidak. Tabel dan gambar responden bisa dilihat dibawah ini :

Tabel 21. Hasil Responden Dari Pertanyaan II

Pertanyaan	Ya	Tidak
Apakah Anda Mendapatkan Pembelajaran dari Memainkan <i>Game</i> Ini.	38	2

Apakah Anda Mendapatkan Pembelajaran dari Memainkan *Game* Ini ?
40 jawaban



Gambar 52. Hasil Responden Dari Pertanyaan II

3) Seberapa Menariknya Pembelajaran Dalam *Game* Ini

Data yang diperoleh dari 40 responden siswa dan siswi mengenai seberapa menariknya pembelajaran aplikasi *Game* yang diuji terdapat 20 siswa dan siswi (50%) yang memilih sangat bagus sangat menarik dan 20 siswa dan siswi (50%) yang memilih menarik. Kurang menarik dan tidak menarik memiliki 0 responden. Tabel dan gambar responden bisa dilihat dibawah ini :

Tabel 22. Hasil Responden Dari Pertanyaan III

Pertanyaan	Sangat Menarik	Menarik	Kurang Menarik	Tidak Buruk
Seberapa Menariknya Pembelajaran Dalam <i>Game</i> Ini	20	20	0	0

Seberapa Menarikkah Pembelajaran Dalam Game Ini ?

40 jawaban



Gambar 53. Hasil Responden Dari Pertanyaan III

4) Menurut Anda Seberapa Efektifkah Anda Dalam Mendapat Pembelajaran Dalam Aplikasi *Game* Ini

Data yang diperoleh dari 40 responden siswa dan siswi mengenai seberapa efektifnya mereka dalam mendapatkan pembelajaran dalam aplikasi *Game* yang diuji, dan hasilnya terdapat 13 siswa dan siswi (32,5%) yang memilih sangat efektif dan 25 siswa dan siswi (62,5%) yang memilih cukup efektif. Kurang efektif dipilih oleh 1 (5%) siswa/siswi dan tidak efektif memiliki 0 responden. Tabel dan gambar responden bisa dilihat dibawah ini :

Tabel 23. Hasil Responden Dari Pertanyaan IV

Pertanyaan	Sangat Efektif	Cukup Efektif	Kurang Efektif	Tidak Efektif
Menurut Anda Seberapa Efektifkah Anda Dalam Mendapat Pembelajaran Dalam Aplikasi <i>Game</i> Ini	13	25	1	0

Menurut Anda Seberapa Efektifkah Anda Dalam Mendapat Pembelajaran Dalam Aplikasi Game Ini
40 jawaban



Gambar 54. Hasil Responden Dari Pertanyaan IV

5) Apakah Anda Ingin Memainkan *Game* Edukasi Sejenis Ini Jika Ada

Data yang diperoleh dari 40 responden siswa dan siswi mengenai seberapa inginya mereka memainkan *Game* sejenis aplikasi *Game* yang diuji terdapat 13 siswa dan siswi (32,5%) yang memilih sangat efektif dan 25 siswa dan siswi (62,5%) yang memilih cukup efektif. Kurang efektif dipilih oleh 1 (5%) siswa/siswi dan tidak efektif memiliki 0 responden. Tabel dan gambar responden bisa dilihat dibawah ini :

Tabel 24. Hasil Responden Dari Pertanyaan V

Pertanyaan	Ya	Tidak
Apakah Anda Ingin Memainkan <i>Game</i> Edukasi Sejenis Ini Jika Ada	38	2

Apakah Anda Ingin Memainkan Game Edukasi Sejenis Ini Jika Ada ?
40 jawaban



Gambar 55. Hasil Responden Dari Pertanyaan V

6) Menurut Anda Bagaimana Sistem AR dalam Game Ini

Data yang diperoleh dari 40 responden siswa dan siswi mengenai seberapa bagusnya sistem AR (*Augmented Reality*) pada aplikasi *Game* yang diuji terdapat 19 siswa dan siswi (47.5%) yang memilih sangat bagus dan 20 siswa dan siswi (50%) yang memilih cukup bagus. Kurang bagus dipilih oleh 1 (5%) siswa/siswi dan buruk memiliki 0 responden. Tabel dan gambar responden bisa dilihat dibawah ini:

Tabel 25. Hasil Responden Dari Pertanyaan VI

Pertanyaan	Sangat Bagus	Cukup Bagus	Kurang Bagus	Buruk
Menurut Anda Bagaimana Sistem AR dalam Game Ini	19	20	1	0

Menurut Anda Bagaimana Sistem AR dalam Game Ini ?

40 jawaban



Gambar 56. Hasil Responden Dari Pertanyaan VI

7) Seberapa Menarik Sistem AR dalam Game Ini

Data yang diperoleh dari 40 responden siswa dan siswi mengenai seberapa menariknya sistem AR (*Augmented Reality*) pada aplikasi Game yang diuji terdapat 21 siswa dan siswi (52,5%) yang memilih sangat menarik dan 19 siswa dan siswi (47,5%) yang memilih cukup menarik. Kurang bagus dan buruk memiliki 0 responden. Tabel dan gambar responden bisa dilihat dibawah ini :

Tabel 26. Hasil Responden Dari Pertanyaan VII

Pertanyaan	Sangat Menarik	Cukup Menarik	Kurang Menarik	Tidak Menarik
Seberapa Menarik Sistem AR dalam Game Ini	21	19	0	0

Seberapa Menarik Sistem AR dalam Game Ini ?

40 jawaban



Gambar 57. Hasil Responden Dari Pertanyaan VII

d. Kesimpulan Dari Hasil Pengujian

Setelah dilakukan pengujian komponen dan pengujian dari 40 responden siswa dan siswi terhadap aplikasi *Game* yang dibangun, maka dapat disimpulkan hasil yang didapatkan yaitu :

- 1) Setiap komponen dapat berjalan dengan baik, dan berfungsi sesuai dengan harapan *Developer*.
- 2) Aplikasi *Game* ini memerlukan intensitas cahaya minimum di atas 51 *lux* dalam memainkannya dan tidak dapat dimainkan ketika berada di lingkungan dengan intensitas cahaya dibawah 51 *lux*.
- 3) Menggunakan sistem *Augmented Reality* dimana tempat kita bermain adalah lingkungan kita sendiri dan system *Augmented Reality* dianggap menarik oleh para siswa dan siswi.
- 4) Aplikasi ini mengajarkan sejarah pertempuran 10 November dari *CutScene Intro* dan *Outronya* secara menarik dengan menggabungkan belajar dan bermain.
- 5) Para siswa dan siswi ingin kembali bermain dan mencoba jenis *Game* edukasi yang serupa dan sejenis.
- 6) Aplikasi bersifat *mobile* dan dapat dimainkan dimana saja.

BAB V

KESIMPULAN DAN SARAN

A. Kesimpulan

Berdasarkan latar belakang dan pembahasan yang telah dibahas pada bab-bab sebelumnya, dan pengembangan yang dilakukan dalam perancangan implementasi *Augmented Reality* dalam mengembangkan *Game* untuk memperkenalkan sejarah kemerdekaan Republik Indonesia. Serta penelitian yang dilakukan di lingkungan SMKN 2 Selayar, maka dapat disimpulkan :

1. Aplikasi *Game Augmented Reality* dalam memperkenalkan sejarah kemerdekaan Republik Indonesia khususnya sejarah 10 November 1945 dibuat dan digunakan sebagai sarana media sarana pembelajaran yang baru dan menarik bagi para pelajar dalam mempelajari sejarah kemerdekaan negeri mereka yang menerapkan konsep belajar dan bermain dan menggunakan sistem AR yang dapat menambah pengalaman bermain dan *immersivnes* dalam *Game*.
2. Hasil Pengujian dengan menggunakan metode *black-box* pada aplikasi *Game* menunjukkan bahwa aplikasi ini dapat bejalan sesuai dengan harapan *Developer* yang dimana aplikasi *Game* ini dapat dimainkan di perangkat *mobile* yang kompetibel yang dimana bagi *android* mempunyai sistem *android* versi 7 Ke atas dan harus mendukung AR *core* serta memiliki intensitas cahaya ruangan yang cukup yaitu dengan intensitas cahaya minimum di atas 51 *lux* dan akan lebih baik lagi ketika berada di intensitas cahaya diatas 200 *lux*.
3. Hasil respon siswa dan siswi sangatlah mendukung aplikasi ini dimana mereka tertarik dan menikmati serta mendapatkan pembelajaran dari aplikasi ini, sehingga mereka ingin memainkan aplikasi edukasi sejenis aplikasi ini lagi.

B. Saran

Adapun saran yang diberikan peneliti terhadap peneliti yang akan melakukan penelitian selanjutnya yang berpaku pada penelitian ini ialah bagi penelitian berikutnya diharapkan dapat memakai teknologi dan *device* yang dapat

menggunakan sistem *LIDAR* yang dapat mengoptimalkan *Scan* lingkungan sehingga system AR tidak lagi berpaku pada intensitas cahaya lingkungan sekitar.



DAFTAR PUSTAKA

- Abbas, E. W. (2017). *Pendidikan Sejarah, Patriotisme & Karakter Bangsa Malaysia-Indonesia*. Program Suti Pendidikan IPS dan Program Studi Pendidikan Sejarah FKIP
- Ar-rafi, F. F., & Zailani, A. U. (2020). RANCANG BANGUN GAME EDUKASI SEJARAH BERBASIS ANDROID MENGGUNAKAN GAME ENGINE UNITY 3D. *PROSIDING SEMINAR INFORMATIKA DAN SISTEM INFORMASI*, 5(3), 229–238.
- Aulianti, W. D., Karim, S. A., & Riska, M. (2021). Pengembangan *Game* Pendidikan Anti Korupsi Berbasis Android. *Jurnal MediaTIK*, 4(2), 27–32.
- Bahri, F. R. A., & Huda, S. N. (2022). Tinjauan Literatur: *Game* Edukasi Sejarah Berbagai Peristiwa yang Ada di Indonesia. *AUTOMATA*, 3(1).
- Boham, I. S., & Sentinuwo, S. (2017). Rancang Bangun Aplikasi *Game* Pengenalan Sejarah Perang Tondano. *Jurnal Teknik Informatika*, 11(1).
- Bone.co.id. (2020). *Sejarah dan Makna Peringatan Hari Pahlawan 10 November*. Bone.Go.Id. <https://bone.go.id/2020/11/01/sejarah-dan-makna-peringatan-hari-pahlawan-10-november/>
- Cikka, H. (2019). SINOPSIS DALAM PEMBELAJARAN SEJARAH (Cara Mudah Memahami dan Mengingat Peristiwa Sejarah). *Scolae: Journal of Pedagogy*, 2(2), 300–306.
- Fatimah, D. D. S., Tresnawati, D., & Ma'rup, C. S. (2017). Perancangan *Game* Puzzle Untuk Pembelajaran Menggunakan Metodologi Multimedia. *Jurnal Algoritma*, 14(2), 281–287.
- Harahap, A., Sucipto, A., & Jupriyadi, J. (2020). Pemanfaatan *Augmented Reality* (Ar) Pada Media Pembelajaran Pengenalan Komponen Elektronika Berbasis Android. *Jurnal Ilmiah Infrastruktur Teknologi Informasi*, 1(1), 20–25.
- Karundeng, C. O., Mamahit, D. J., & Sugiarto, B. A. (2018). Rancang Bangun Aplikasi Pengenalan Satwa Langka di Indonesia Menggunakan *Augmented Reality*. *Jurnal Teknik Informatika*, 13(1).
- Kurniawan, T. A. (2018). Pemodelan use case (UML): evaluasi terhadap beberapa

- kesalahan dalam praktik. *J. Teknol. Inf. Dan Ilmu Komput*, 5(1), 77.
- Kustriawan, N. (2018). Pembuatan *Game* Bamboo Menggunakan Aplikasi Unity 3D. *Journal Of Applied Multimedia and Networking*, Vol 2 No 2. <https://jurnal.polibatam.ac.id/index.php/JAMN/article/view/650>
- Mongi, L. S., Lumenta, A. S. M., & Sambul, A. M. (2018). Rancang Bangun *Game* Adventure of Unsrat Menggunakan *Game* Engine Unity. *Jurnal Teknik Informatika*, 13(1).
- Pamoedji, A. K., & Maryuni, R. S. (2017). *Mudah Membuat Game Augmented Reality (AR) dan Virtual Reality (VR) dengan Unity 3D*. Elex Media Komputindo.
- Prof. Dr. Sugiyono. (2013). *METODE PENELITIAN KUANTITATIF, KUALITATIF, DAN R&D*. ALFABETA, CV.
- Ramadhani, N. (2020). *Sejarah 17 Agustus 1945 Sebagai Hari Kemerdekaan Indonesia*. [Www.Akseleran.Co.Id](http://www.Akseleran.Co.Id). <https://www.akseleran.co.id/blog/17-agustus/>
- Rohmawati, I., Sudargo, S., & Menarianti, I. (2019). Pengembangan *Game* Edukasi Tentang Budaya Nusantara “Tanara” Menggunakan Unity 3D Berbasis Android. *Jurnal SITECH: Sistem Informasi Dan Teknologi*, 2(2), 173–184.
- Siregar, H. F., Siregar, Y. H., & Melani, M. (2018). Perancangan Aplikasi Komik Hadist Berbasis Multimedia. (*JurTI*) *Jurnal Teknologi Informasi*, 2(2), 113–121.
- Tanjung, M. A. P. (2013). Analisis Pengaruh Storytelling Terhadap *Game* Lorong Waktu–Pangeran Diponegoro Sebagai Media Edukasi Sejarah. *Jurnal Ilmiah Komputer Dan Informatika (KOMPUTA)*.
- Tempo.co. (2010). *Game Pertempuran 10 November Ala ITS*. Ekno.Tempo.Co. <https://tekno.tempo.co/read/285449/Game-pertempuran-10-november-ala-its>
- Wibawa, A. P., Bangsa, P. G., Wahyuni, M., Karnadi, H., Hadi, U., Widyo Harsanto, P., Maharsi, I., Fx Widyatmoko, W., Ariani, A., & Setya Darmawan, Y. (2021). *PROSIDING SEMINAR NASIONAL DESAIN KOMUNIKASI VISUAL: Multi Peran Desain Komunikasi Visual Dalam Perkembangan Disiplin Ilmu dan Teknologi*. BP ISI Yogyakarta.

Zebua, T., Nadeak, B., & Sinaga, S. B. (2020). Pengenalan Dasar Aplikasi Blender 3D dalam Pembuatan Animasi 3D. *Jurnal ABDIMAS Budi Darma*, 1(1), 18–21.



LAMPIRAN

Lampiran 1 : Daftar Responden Angket

No	Nama	E-mail	Kelas	Jurusan
1	Syahrullah	sondhetnr@gmail.com	12	TKJ
2	Iam Ashari	iyam.ashari077@gmail.com	12	TKJ
3	Wasyiatul Akmal	wasyiatulakmal07@gmail.com	12	TKJ
4	Muhammad Zaenal	muhammadzaenal022005@gmail.com	12	TKJ
5	Ary Wahyudi	aryyudi14@gmail.com	12	TKJ
6	Herianto	innungheri@gmail.com	12	TKJ
7	Wahyu Dermawan	Wahyudermawan923@gmail.com	12	TKJ
8	Miftahul Jannah	mifta222003@gmail.com	12	TKJ
9	rifhal aryadi	rifalaryadi3@gmail.com	12	TKJ
10	Nirwana	nirwana026_2005@gmail.com	12	TKJ
11	Dina Afizah	afizahdina585@gmail.com	12	TKJ
12	Dwi Permata Insani	dwipermatainsani152005@gmail.com	12	TKJ
13	Muh Herdiansah Erlangga	erlangganga81@gmail.com	12	TKJ
14	Sahrani amanda	Sahraniamanda0202@gmail.com	12	TKJ
15	Sabrina	0821-9599-0632	11	TKJ
16	Andika eka saputra	andikaekasaputra41@gmail.com	12	TKJ
17	Nur fauzi	Fauzinur918@gmail.com	12	TKJ
18	Sri Handayani putri	srihandayaniputri753@gmail.com	12	TKJ
19	Martriana	martrianaa@gmail.com	12	TKJ
20	Az Zahri Reysaldi R	azzahrirreysaldi@gmail.com	11	TKJ
21	Bunga Annisa	-	11	TKJ
22	Muh.Aswan Afandi	w236163@gmail.com	11	TKJ
23	Refiq Reza	traofalgar090@gmail.com	11	TKJ
24	Nur Aisyah	hi_aisyahnur@gmail.com	12	TKJ

25	Paril Husada	parilhusada123456@gmail.com	11	TKJ
26	Fadel Alvian	fadelalvian1106@gmail.com	11	TKJ
27	Ilham Anugrah	ilhamanugrah516@gmail.com	11	TKJ
28	Sri Wahyuni Rahman	Sriwahyuniirahman348@gmail.com	11	TKJ
29	Irsal Wahyudi	Irsal Wahyudi3@gmail.com	11	TKJ
30	Feri Adiyaksa Yunus	feriadiyu5@gmail.com	11	TKJ
31	Syeli Angraeni	Syelysyely09@gmail.com	11	TKJ
32	Andi Fitra Pratama	-	11	TKJ
33	Aulya Firmanda	azavabeatrixza@gmail.com	11	TKJ
34	Maulidiya Anastasya	maulidiyaanastasya@gmail.com	11	TKJ
35	Herlit Anatasyia	Herlitaanastasyi@gmail.com	11	TKJ
36	Wahdini	wahdini113355@gmail.com	11	TKJ
37	Resti lestari	Lsstaryresty83@gmail.com	11	TKJ
38	Nirmalasari	-	11	TKJ
39	Sabrina	0821-9599-0632	11	TKJ
40	Sayyid Fauzan Assaggaf	mrfauzanmtsp@gmail.com	11	TKJ

Lampiran 2 : Source Code

Game Movement

using UnityEngine;

```
public class GameMovement : MonoBehaviour
{
    public int m_PlayerNumber = 1;
    public float m_Speed = 12f;
    public float m_TurnSpeed = 180f;
    public AudioSource m_MovementAudio;
    public AudioClip m_EngineIdling;
    public AudioClip m_EngineDriving;
    public float m_PitchRange = 0.2f;

    private string m_MovementAxisName;
    private string m_TurnAxisName;
    private Rigidbody m_Rigidbody;
    private float m_MovementInputValue;
    private float m_TurnInputValue;
    private float m_OriginalPitch;
    private ParticleSystem[] m_particleSystems;

    FixedJoystick JoyStick;

    private void Awake()
    {
        m_Rigidbody = GetComponent<Rigidbody>();
    }

    private void OnEnable()
    {
        m_Rigidbody.isKinematic = false;

        m_MovementInputValue = 0f;
        m_TurnInputValue = 0f;

        m_particleSystems = GetComponentsInChildren<ParticleSystem>();
        for (int i = 0; i < m_particleSystems.Length; ++i)
        {
            m_particleSystems[i].Play();
        }
    }

    private void OnDisable()
    {
        m_Rigidbody.isKinematic = true;

        for (int i = 0; i < m_particleSystems.Length; ++i)
        {
            m_particleSystems[i].Stop();
        }
    }
}
```

```

}

private void Start()
{
    JoyStick = FindObjectOfType<FixedJoyStick>();
    m_MovementAxisName = "Vertical";
    m_TurnAxisName = "Horizontal";
    m_OriginalPitch = m_MovementAudio.pitch;
}

private void Update()
{
    m_MovementInputValue = Input.GetAxis(m_MovementAxisName);
    m_TurnInputValue = Input.GetAxis(m_TurnAxisName);

    EngineAudio();
}

private void EngineAudio()
{
    if (Mathf.Abs(JoyStick.Vertical) < 0.1f &&
        Mathf.Abs(JoyStick.Horizontal) < 0.1f)
    {
        if (m_MovementAudio.clip == m_EngineDriving)
        {
            m_MovementAudio.clip = m_EngineIdling;
            m_MovementAudio.pitch = Random.Range(m_OriginalPitch -
            m_PitchRange, m_OriginalPitch + m_PitchRange);
            m_MovementAudio.Play();
        }
    }
    else
    {
        if (m_MovementAudio.clip == m_EngineIdling)
        {
            m_MovementAudio.clip = m_EngineDriving;
            m_MovementAudio.pitch = Random.Range(m_OriginalPitch -
            m_PitchRange, m_OriginalPitch + m_PitchRange);
            m_MovementAudio.Play();
        }
    }
}

private void FixedUpdate()
{
    Move();
    Turn();
}

private void Move()
{

```

```

        Vector3 movement = transform.forward * JoyStick.Vertical *
m_Speed * Time.deltaTime;

        m_Rigidbody.MovePosition(m_Rigidbody.position + movement);
    }

    private void Turn()
    {
        float turn = JoyStick.Horizontal * m_TurnSpeed *
Time.deltaTime;

        Quaternion turnRotation = Quaternion.Euler(0f, turn, 0f);
        m_Rigidbody.MoveRotation(m_Rigidbody.rotation * turnRotation);
    }
}

**GameShooting**
using UnityEngine;
using UnityEngine.UI;

public class GameShooting : MonoBehaviour
{
    public int m_PlayerNumber = 1;
    public Rigidbody m_Shell;
    public Transform m_FireTransform;
    public Slider m_AimSlider;
    public AudioSource m_ShootingAudio;
    public AudioClip m_ChargingClip;
    public AudioClip m_FireClip;
    public float m_LaunchForce = 5f;

    private float m_ChargeSpeed;
    public bool m_Fired;
    Joybutton Joybutton;

    private void OnEnable()
    {
    }

    private void Start()
    {
        Joybutton = FindObjectOfType<Joybutton>();
    }

    private void Update()
    {
        if (Joybutton.Pressed && !m_Fired)
        {

```

```

        Fire();
    }
    else if (!Joybutton.Pressed)
    {

        m_Fired = false;

    }
}

private void Fire()
{
    m_Fired = true;

    Rigidbody ShellInstance =
        Instantiate(m_Shell, m_FireTransform.position,
m_FireTransform.rotation) as Rigidbody;

    ShellInstance.velocity = m_LaunchForce * m_FireTransform.forward;

    m_ShootingAudio.clip = m_FireClip;
    m_ShootingAudio.Play();
}
}

**Health**
using UnityEngine;
using UnityEngine.UI;

public class GameHealth : MonoBehaviour
{
    public float m_StartingHealth = 100f;
    public Slider m_Slider;
    public Image m_FillImage;
    public Color m_FullHealthColor = Color.green;
    public Color m_ZeroHealthColor = Color.red;
    public GameObject m_ExplosionPrefab;

    private AudioSource m_ExplosionAudio;
    private ParticleSystem m_ExplosionParticles;
    private float m_CurrentHealth;
    public bool m_Dead;

    private void Awake()
    {

```

```

        m_ExplosionParticles =
Instantiate(m_ExplosionPrefab).GetComponent<ParticleSystem>();

        m_ExplosionAudio =
m_ExplosionParticles.GetComponent<AudioSource>();

        m_ExplosionParticles.GameObject.SetActive(false);
    }

    public void Update()
    {
        if (m_Dead)
        {
            Destroy(GameObject);
        }
    }

    private void OnEnable()
    {
        m_CurrentHealth = m_StartingHealth;
        m_Dead = false;

        SetHealthUI();
    }

    public void TakeDamage(float amount)
    {
        m_CurrentHealth -= amount;

        SetHealthUI();

        if (m_CurrentHealth <= 0f && !m_Dead)
        {
            OnDeath();
        }
    }

    private void SetHealthUI()
    {
        m_Slider.value = m_CurrentHealth;

        m_FillImage.color = Color.Lerp(m_ZeroHealthColor,
m_FullHealthColor, m_CurrentHealth / m_StartingHealth);
    }

    private void OnDeath()
    {
        m_Dead = true;

        m_ExplosionParticles.transform.position = transform.position;
        m_ExplosionParticles.GameObject.SetActive(true);

        m_ExplosionParticles.Play();
    }

```

```

        m_ExplosionAudio.Play();
        Destroy(GameObject);
    }
}

```

****GameAI****

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.AI;

public class GameAI2 : MonoBehaviour
{
    Transform tr_Player;
    float f_RotSpeed = 1f, f_MoveSpeed = 0.5f;

    public LayerMask whatIsGround, whatIsPlayer;

    public Vector3 walkPoint;
    bool walkPointSet = false;
    public float walkPointRange;

    public float timeBetweenAttacks;
    bool alreadyAttacked;
    public GameObject projectile;

    public float sightRange, attackRange;
    public bool playerInSightRange, playerInAttackRange;

    public Rigidbody m_Shell;
    public Transform m_FireTransform;
    public float m_LaunchForce = 3f;

    public GameObject[] PlacementObjectPf;
    GameObject GamePlace;
    string GameName;

    public float offset;

    float timer;
    int waitingTime = 2;

    // Use this for initialization
    void Start()
    {
        int selectedGame = PlayerPrefs.GetInt("selectedGame");
        GamePlace = PlacementObjectPf[selectedGame];
        GameName = GamePlace.name + "(Clone)";
        Debug.Log(GameName);
    }
}

```

```

}

// Update is called once per frame
void Update()
{
    /* Look at Player*/
    tr_Player = GameObject.Find(GameName).transform;

    playerInSightRange = Physics.CheckSphere(transform.position,
sightRange, whatIsPlayer);
    playerInAttackRange = Physics.CheckSphere(transform.position,
attackRange, whatIsPlayer);

    if (!playerInSightRange && !playerInAttackRange)
    {
        Patrolling();
        Debug.Log("Patrol");
    }
    if (playerInSightRange && !playerInAttackRange) ChasePlayer();
    if (playerInAttackRange && playerInSightRange) AttackPlayer();
}

private void Patrolling()
{
    if (!walkPointSet) SearchWalkPoint();

    Vector3 distanceToWalkPoint = transform.position - walkPoint;

    if (distanceToWalkPoint.magnitude < 1f)
        walkPointSet = false;
}

private void SearchWalkPoint()
{
    float randomZ = Random.Range(-walkPointRange, walkPointRange);
    float randomX = Random.Range(-walkPointRange, walkPointRange);

    transform.position = new Vector3(transform.position.x + randomX,
transform.position.y, transform.position.z + randomZ);

    if (Physics.Raycast(walkPoint, -transform.up, 2f, whatIsGround))
        walkPointSet = true;
}

private void ChasePlayer()
{
    transform.rotation = Quaternion.Slerp(transform.rotation

```

```

Quaternion.LookRotation(tr_Player.position - transform.position)
                                , f_RotSpeed *
Time.deltaTime);

    transform.position += transform.forward * f_MoveSpeed *
Time.deltaTime;
}

private void AttackPlayer()
{
    transform.LookAt(tr_Player);
    if (!alreadyAttacked)
    {
        timer += Time.deltaTime;
        if (timer > waitingTime)
        {
            Rigidbody ShellInstance =
Instantiate(m_Shell, m_FireTransform.position,
m_FireTransform.rotation) as Rigidbody;

            ShellInstance.velocity = m_LaunchForce *
m_FireTransform.forward;
            timer = 0;
        }

        alreadyAttacked = true;
        Invoke(nameof(ResetAttack), timeBetweenAttacks);
    }
}

private void ResetAttack()
{
    alreadyAttacked = false;
}
}

```

****Turret****

```

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

public class Turret : MonoBehaviour
{
    protected TurretState currentState;

    public Transform Target { get; set; }
}

```

```

public Rigidbody m_Shell;
public Transform m_FireTransform;
public float m_LaunchForce = 20f;
float timer;
int waitingTime = 2;
public bool shoot = false;

[SerializeField]
private Vector3 aimOffset;

[SerializeField]
private Transform rotator;

[SerializeField]
private Transform ghostRotator;

[SerializeField]
private float rotationSpeed;

[SerializeField]
private GameObject Shell;

public Quaternion DefaultRotation { get; set; }
public Transform Rotator { get => rotator; set => rotator = value; }
public Vector3 AimOffset { get => aimOffset; set => aimOffset = value;
}

public Transform GhostRotator { get => ghostRotator; set =>
ghostRotator = value; }

public float RotationSpeed { get => rotationSpeed; set => rotationSpeed
= value; }

private void Start()
{
    DefaultRotation = rotator.rotation;
    ChangeState(new IdleState());
}

private void Update()
{
    if(GhostRotator != null)
    {
        currentState.Update();
        if (shoot)
        {
            timer += Time.deltaTime;
            if (timer > waitingTime)
            {

```

```

        Shoot();
        timer = 0;
    }
}
}
else
{
    Destroy(GameObject);
}
}
public void Shoot()
{
    Rigidbody ShellInstance =
        Instantiate(m_Shell, m_FireTransform.position,
m_FireTransform.rotation) as Rigidbody;

    ShellInstance.velocity = m_LaunchForce * m_FireTransform.forward;
}

public void ChangeState(TurretState newState)
{
    if(newState != null)
    {
        newState.Exit(this);
    }
    this.currentState = newState;
    newState.Enter(this);
}

private void OnTriggerEnter(Collider other)
{
    currentState.OnTriggerEnter(other);
}

private void OnTriggerExit(Collider other)
{
    currentState.OnTriggerExit(other);
}
}
}

```

****GameManager****

```

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

public class GameManager : MonoBehaviour
{
    public static bool easy;
    public static bool normal;
    public static bool hard;
}

```

```

public static float delayTime = 0.3f;
public void DificultyMenu()
{
    easy = false;
    normal = false;
    hard = false;
    Invoke("toSceneDificultyMenus", delayTime);
}
public void StartEasy()
{
    Invoke("toSceneStartEasy", delayTime);
}
public void StartNormal()
{
    Invoke("toSceneStartNormal", delayTime);
}
public void StartHard()
{
    Invoke("toSceneStartHard", delayTime);
}
public void MainGame()
{
    Invoke("toSceneMainGame", delayTime);
}
public void StartAgain()
{
    Invoke("toSceneStartAgain", delayTime);
}
public void SureQuit()
{
    Invoke("toSceneSureQuit", delayTime);
}
public void QuitGame()
{
    Invoke("toSceneQuitGame", delayTime);
}
public void Menang()
{
    Invoke("toSceneMenang", delayTime);
}
public void PilihGame()
{
    Invoke("toScenePilihGame", delayTime);
}

```

```

//toScene no Delay
void toSceneDificultyMenus()
{
    SceneManager.LoadScene(4);
}

void toSceneSureQuit()
{
    SceneManager.LoadScene(3);
}

```

```

void toSceneQuitGame()
{
    Application.Quit();
}

void toSceneStartAgain()
{
    SceneManager.LoadScene(0);
}

void toSceneMainGame()
{
    if (easy == true)
    {
        Debug.Log("Main Easy");
        SceneManager.LoadScene(6);
    }
    else if (normal == true)
    {
        SceneManager.LoadScene(7);
    }
    else if (hard == true)
    {
        SceneManager.LoadScene(8);
    }
}

void toSceneStartHard()
{
    hard = true;
    SceneManager.LoadScene("Intro");
    Debug.Log(hard);
}

void toSceneStartEasy()
{
    easy = true;
    SceneManager.LoadScene("Intro");
    Debug.Log(easy);
}

void toSceneStartNormal()
{
    normal = true;
    SceneManager.LoadScene("Intro");
    Debug.Log(normal);
}

void toSceneMenang()
{
    SceneManager.LoadScene(2);
}

void toScenePilihGame()
{
    SceneManager.LoadScene(10);
}
}

```

****GameEventManager****

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

public class GameEventManager : MonoBehaviour
{
    public GameObject[] PlacementObjectPf;
    GameObject GamePlace;
    string GameName;
    private void Start()
    {
        int selectedGame = PlayerPrefs.GetInt("selectedGame");
        GamePlace = PlacementObjectPf[selectedGame];
        GameName = GamePlace.name + "(Clone)";
        Debug.Log(GameName);
    }
    void Update()
    {
        if (GameObject.Find("Turret(Clone)") == null &&
            GameObject.Find("Game - Enemy AR(Clone)") == null)
        {
            Debug.Log("Win");
            SceneManager.LoadScene("Outro");
        }
        if (GameObject.Find(GameName) == null)
        {
            SceneManager.LoadScene(1);
            Debug.Log("Lose");
        }
    }
}
```

****ARController****

```
using System.Collections.Generic;
using Niantic.ARDKit.AR;
using Niantic.ARDKit.AR.Anchors;
using Niantic.ARDKit.AR.ARSessionEventArgs;
using Niantic.ARDKit.AR.HitTest;
using Niantic.ARDKit.External;
using Niantic.ARDKit.Utilities;
using Niantic.ARDKit.Utilities.Input.Legacy;
using Niantic.ARDKit.Utilities.Logging;
using UnityEngine;

namespace Niantic.ARDKit.Examples.Helpers
{
    public class ARControllerEasy : MonoBehaviour
```

```

{
    public Camera Camera;

    [EnumFlagAttribute]
    public ARHitTestResultType HitTestType =
    ARHitTestResultType.ExistingPlane;

    public GameObject[] PlacementObjectPf;
    public GameObject PlacementObjectPfEnemyTurret;
    public GameObject PlacementObjectPfEnemyGame;

    public GameObject JoyStickControler;
    public GameObject GameManagers;
    int jumlahGame = 0;
    GameObject GamePlace;
    string GameName;

    private List<GameObject> _placedObjects = new List<GameObject>();
    private IARSession _session;
    private void Start()
    {
        ARSessionFactory.SessionInitialized +=
    OnAnyARSessionDidInitialize;
        JoyStickControler.SetActive(false);
        GameManagers.SetActive(false);
        int selectedGame = PlayerPrefs.GetInt("selectedGame");
        GamePlace = PlacementObjectPf[selectedGame];
        GameName = GamePlace.name + "(Clone)";
        Debug.Log(GameName);
    }

    private void
    OnAnyARSessionDidInitialize(AnyARSessionInitializedArgs args)
    {
        _session = args.Session;
        _session.Deinitialized += OnSessionDeinitialized;
    }

    private void OnSessionDeinitialized(ARSessionDeinitializedArgs
args)
    {
        ClearObjects();
    }

    private void OnDestroy()
    {
        ARSessionFactory.SessionInitialized -=
    OnAnyARSessionDidInitialize;
        _session = null;
        ClearObjects();
    }
}

```

```

}

private void ClearObjects()
{
    foreach (var placedObject in _placedObjects)
    {
        Destroy(placedObject);
    }

    _placedObjects.Clear();
}

private void Update()
{
    if (_session == null)
    {
        return;
    }

    if (PlatformAgnosticInput.touchCount <= 0)
    {
        return;
    }

    var touch = PlatformAgnosticInput.GetTouch(0);

    if ((PlatformAgnosticInput.touchCount > 0) && (touch.phase ==
    TouchPhase.Began) && (jumlahGame < 1))
    {
        TouchBegan(touch);
        if (GameObject.Find(GameName) != null)
        {
            jumlahGame++;
            Debug.Log(jumlahGame);
            JoyStickController.SetActive(true);
            GameManagers.SetActive(true);
        }
    }
}

private void TouchBegan(Touch touch)
{
    var currentFrame = _session.CurrentFrame;
    if (currentFrame == null)
    {
        return;
    }

    var results = currentFrame.HitTest
    (
        Camera.pixelWidth,
        Camera.pixelHeight,
        touch.position,
        HitTestType
    );

    int count = results.Count;
}

```

```

    if (count <= 0)
        return;

    var result = results[0];

    var hitPosition = result.WorldTransform.ToPosition();

    _placedObjects.Add(Instantiate(GamePlace, hitPosition,
Quaternion.identity));

    var hitPositionTurret = hitPosition;

    hitPositionTurret.y = hitPosition.y + 0.03f;
    hitPositionTurret.x = Random.Range((hitPosition.x - 0.5f),
(hitPosition.x + 1.00f));
    hitPositionTurret.z = Random.Range((hitPosition.z + 0.1f),
(hitPosition.z + 0.50f));

    _placedObjects.Add(Instantiate(PlacementObjectPfEnemyTuret,
hitPositionTurret, Quaternion.identity));

    var hitPositionGame = hitPosition;
    hitPositionGame.y = hitPosition.y;

    hitPositionGame.x = Random.Range((hitPosition.x + 0.1f),
(hitPosition.x + 1.00f));
    hitPositionGame.z = Random.Range((hitPosition.z - 0.5f),
(hitPosition.z + 0.50f));
    _placedObjects.Add(Instantiate(PlacementObjectPfEnemyGame,
hitPositionGame, Quaternion.identity));
    Debug.Log(hitPositionGame);
    var anchor = result.Anchor;
    Debug.LogFormat
    (
        "Spawning cube at {0} (anchor: {1})",
        hitPosition.ToString("F4"),
        anchor == null
            ? "none"
            : anchor.AnchorType + " " + anchor.Identifier
    );
    };
}
}
}

```

****GameSelection****

```

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

public class GameSelection : MonoBehaviour
{

```

```
public GameObject[] Game;
public int selectedGame = 0;

public void NextGame()
{
    Debug.Log(selectedGame);
    Game[selectedGame].SetActive(false);
    selectedGame = (selectedGame + 1) % Game.Length;
    Game[selectedGame].SetActive(true);
}

public void PreviousGame()
{
    Game[selectedGame].SetActive(false);
    selectedGame--;
    if(selectedGame < 0)
    {
        selectedGame += Game.Length;
    }
    Game[selectedGame].SetActive(true);
}

public void GameSelected()
{
    PlayerPrefs.SetInt("selectedGame", selectedGame);
}
}
```



Lampiran 3 : Hasil Pengujian Aplikasi



Gambar 1. Menu Utama



Gambar 2. Pemilihan Tingkat Kesulitan



Gambar 3. CutScene Awal



Gambar 4. Scan Lingkungan



Gambar 5. Dalam Game Easy



Gambar 6. Menang



Gambar 7. CutScene Outro



Gambar 8. Kalah



Gambar 9. Dalam Game Normal



Gambar 10. Dalam Game Hard



Gambar 11. Menu Pemilihan Tank



Gambar 12. Dalam Game Dengan Jenis Tank 2



Gambar 13. Dalam Game Dengan Jenis Tank 3



Gambar 14. Dalam Game Dengan Jenis Tank 4



Lampiran 4 : Hasil Pengujian Intensitas Cahaya *Indoor Redmi Note 10 5G*

1. Pengujian Dengan Intensitas Cahaya 0-50 *lux*



Gambar 15. Pengujian Intensitas Cahaya *Indoor* 0-50 *lux*

2. Pengujian Dengan Intensitas Cahaya 51-200 *lux*



Gambar 16. Pengujian Intensitas Cahaya *Indoor* 51-200 *lux*

3. Pengujian Dengan Intensitas Cahaya $> 200 \text{ lux}$



Gambar 17. Pengujian Intensitas Cahaya *Indoor* $> 200 \text{ lux}$



Lampiran 5 : Pengujian Intensitas Cahaya *Outdoor Redmi Note 10 5G*

1. Pengujian Dengan Intensitas Cahaya 0-50 lux



Gambar 18. Pengujian Intensitas Cahaya *Outdoor 0-50 lux*

2. Pengujian Dengan Intensitas Cahaya 51-200 lux



Gambar 19. Pengujian Dengan Intensitas Cahaya *Outdoor 51-200 lux*

3. Pengujian Dengan Intensitas Cahaya $> 200 \text{ lux}$



Gambar 20. Pengujian Dengan Intensitas Cahaya Outdoor $> 200 \text{ lux}$



Lampiran 6 : Pengujian Intensitas Cahaya Indoor Poco M3 Pro 5G

1. Pengujian Dengan Intensitas Cahaya 0-50 lux



Gambar 21. Pengujian Intensitas Cahaya Indoor 0-50 lux

2. Pengujian Dengan Intensitas Cahaya 51-200 lux



Gambar 22. Pengujian Dengan Intensitas Cahaya Indoor 51-200 lux

3. Pengujian Dengan Intensitas Cahaya $> 200 \text{ lux}$



Gambar 23. Pengujian Dengan Intensitas Cahaya *Indoor* $> 200 \text{ lux}$

Lampiran 7: Pengujian Intensitas Cahaya *Outdoor Poco M3 Pro 5G*

1. Pengujian Dengan Intensitas Cahaya 0-50 *lux*



Gambar 24. Pengujian Intensitas Cahaya *Outdoor 0-50 lux*

2. Pengujian Dengan Intensitas Cahaya 51-200 *lux*



Gambar 25. Pengujian Dengan Intensitas Cahaya *Outdoor 51-200 lux*

3. Pengujian Dengan Intensitas Cahaya $> 200 \text{ lux}$



Gambar 26. Pengujian Dengan Intensitas Cahaya *Outdoor* $> 200 \text{ lux}$



Lampiran 8 : Surat Keterangan Selesai Meneliti



Gambar 27. Surat Keterangan Selesai Meneliti