



TUGAS AKHIR - IF184802

PERMAINAN REALITAS VIRTUAL ARCHER MASTER MENGGUNAKAN TEKNOLOGI OCULUS RIFT DAN *SPEECH RECOGNITION*

ARIQ SUDIBYO
NRP 05111640000086

Dosen Pembimbing
Imam Kuswardayan, S.Kom., M.T.
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.

DEPARTEMEN TEKNIK INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

[Halaman ini sengaja dikosongkan]



TUGAS AKHIR - IF184802

**PERMAINAN REALITAS VIRTUAL ARCHER
MASTER MENGGUNAKAN TEKNOLOGI OCULUS
RIFT DAN SPEECH RECOGNITION**

**ARIQ SUDIBYO
NRP 05111640000086**

**Dosen Pembimbing
Imam Kuswardayan, S.Kom., M.T.
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.**

**DEPARTEMEN TEKNIK INFORMATIKA
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020**

[Halaman ini sengaja dikosongkan]



FINAL PROJECT - IF184802

VIRTUAL REALITY GAME ARCHER MASTER USING OCULUS RIFT AND SPEECH RECOGNITION TECHNOLOGY

**ARIQ SUDIBYO
NRP 05111640000086**

Advisor
Imam Kuswardayan, S.Kom., M.T.
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.

DEPARTMENT of INFORMATICS ENGINEERING
Faculty of Intelligent Electrical and Informatics Technology
Institut Teknologi Sepuluh Nopember
Surabaya 2020

[Halaman ini sengaja dikosongkan]

LEMBAR PENGESAHAN

PERMAINAN REALITAS VIRTUAL ARCHER MASTER MENGGUNAKAN TEKNOLOGI *OCULUS RIFT DAN SPEECH RECOGNITION*

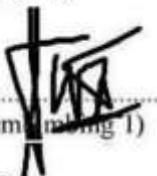
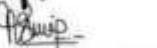
TUGAS AKHIR

Diajukan Guna Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Interaksi, Grafika dan Seni
Program Studi S-1 Departemen Teknik Informatika
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember

Oleh:
ARIQ SUDIBYO
NRP: 05111640000086

Disetujui oleh Dosen Pembimbing Tugas Akhir:

Imam Kuswardayan, S.Kom., MT.
NIP. 197612152003121001


(pembimbing 1)

(pembimbing 2)

Dr.Eng. Nanik Suciati, S.Kom., M.Kom.
NIP. 197104281994122001

**SURABAYA
JUNI, 2020**

[Halaman ini sengaja dikosongkan]

PERMAINAN REALITAS VIRTUAL ARCHER MASTER MENGGUNAKAN TEKNOLOGI OCULUS RIFT DAN SPEECH RECOGNITION

Nama Mahasiswa : Ariq Sudibyo
NRP : 05111640000086
Departemen : Teknik Informatika FTEIC-ITS
Dosen Pembimbing 1 : Imam Kuswardayan, S.Kom., M.T.
Dosen Pembimbing 2 : Dr. Eng. Nanik Suciati, S.Kom, M.Kom.

ABSTRAK

Perkembangan dunia video game saat ini semakin pesat, salah satu genre permainan yang cukup populer adalah shooter. Shooter merupakan subgenre action video game, yang sering digunakan untuk menguji kesadaran spasial, refleks, dan kecekatan pemain. Namun, sekarang kebanyakan game shooter masih berbasis komputer sehingga menyebabkan kesadaran spasial, refleks, dan kecekatan pemain tidak dapat diterapkan pada dunia nyata.

Kekurangan tersebut dapat diatasi dengan menggunakan teknologi realitas virtual. Dalam tugas akhir ini, penulis membangun sebuah permainan realitas virtual shooter panahan yang berjudul Archer Master. Permainan ini menggunakan speech recognition sebagai kendali untuk menggunakan kemampuan yang dimiliki pemain di dalam permainan. Permainan ini menggunakan teknologi Oculus Rift untuk menampilkan realitas virtual. Teknologi ini telah dilengkapi mikrofon terintegrasi yang dapat digunakan sebagai perangkat masukan speech recognition pada video game. Permainan ini dibangun dengan Unity Versi 2019.2.11f1 dengan bahasa pemrograman C# dan Windows Speech Recognition API. Aset permainan sebagian besar diambil dari Unity Asset Store dan internet.

Hasil dari implementasi permainan ini akan diuji menggunakan dua metode pengujian, pengujian fungsionalitas dan pengujian oleh pengguna. 16 poin pada pengujian fungsionalitas berhasil diuji. Sedangkan pengujian oleh pengguna dilakukan oleh 10 orang penguji dengan tiga buah parameter, antarmuka yang menghasilkan 78.6%, immersivity 82%, dan kenyamanan 83.5%. Dengan hasil tersebut dapat disimpulkan permainan Archer Master telah mengimplementasikan perancangan dengan baik.

Kata kunci: *Oculus Rift, Realitas Virtual, Speech Recognition.*

VIRTUAL REALITY GAME ARCHER MASTER USING OCULUS RIFT AND SPEECH RECOGNITION TECHNOLOGY

Name : Ariq Sudibyo
NRP : 05111640000086
Department : Informatics Engineering ELECTICS-ITS
Supervisor I : Imam Kuswardayan, S.Kom., M.T.
Supervisor II : Dr. Eng. Nanik Suciati, S.Kom, M.Kom.

ABSTRACT

The development of the video game world is currently growing rapidly, one genre of games that is quite popular is the shooter. Shooter is a video game action subgenre, which is often used to test spatial awareness, reflexes, and the player's dexterity. However, now most shooter games are still computer-based, causing the player's spatial awareness, reflexes, and dexterity cannot be applied to the real world.

These deficiencies can be overcome by using virtual reality technology. In this final project, the author builds an archery virtual reality game entitled Archer Master. This game uses speech recognition as a control to use the capabilities of players in the game. This game uses Oculus Rift technology to display virtual reality. This technology has been equipped with an integrated microphone that can be used as a speech recognition input device in video games. This game was built with Unity Version 2019.2.11f1 with the C # programming language and Windows Speech Recognition API. Game assets are mostly obtained from the Unity Asset Store and the internet.

The results of the implementation of this game will be tested using two testing methods, functionality testing and testing by users. 16 points on functionality testing successfully tested. While testing by users is carried out by 10 testers with three

parameters, interface that produces 78.6%, immersivity 82%, and comfort 83.5%. With these results it can be concluded that the game Archer Master has implemented the design properly.

Keywords: *Virtual Reality, Oculus Rift, Speech Recognition.*

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah Subhanahu Wata'ala karena atas karunia, nikmat dan rahmat-Nya penulis dapat menyelesaikan tugas akhir yang berjudul:

PERMAINAN REALITAS VIRTUAL ARCHER MASTER MENGGUNAKAN TEKNOLOGI OCULUS RIFT DAN SPEECH RECOGNITION

Sholawat serta salam semoga tak lupa kita sampaikan kepada Rasulullah Shallallahu 'Alaihi Wassalam, yang telah membawa risalah Islam dan Qur'an kepada seluruh umat manusia.

Melalui lembar ini, penulis ingin menyampaikan ucapan terima kasih dan penghormatan yang sebesar-besarnya kepada:

1. Allah SWT atas semua rahmat yang diberikan sehingga penulis dapat menyelesaikan Tugas Akhir ini.
2. Keluarga yang senantiasa mendoakan, memotivasi dan mendukung lahir maupun batin penulis dalam menyelesaikan tanggung jawab ini.
3. Bapak Imam Kuswardayan, S.Kom., MT. dan Ibu Dr. Eng. Nanik Suciati, S.Kom., M.Kom. selaku dosen pembimbing I dan II penulis, yang senantiasa membimbing, memberikan saran, arahan, serta bantuan-bantuan lainnya dalam menyelesaikan tugas akhir ini.
4. Teman-teman TC 2016 yang banyak membantu penulis dalam segala hal, yang berjuang bersama-sama penulis dari awal masuk perkuliahan hingga akhir masa perkuliahan ini.
5. User TA IGS lainnya yang berjuang bersama penulis dalam menyelesaikan tugas akhir ini.
6. Serta pihak-pihak lain yang mohon maaf tidak dapat disebutkan satu per satu, yang telah turut andil dalam membantu penulis selama perkuliahan.

Bagaimanapun juga penulis telah berusaha sebaik-baiknya dalam menyelesaikan tugas akhir ini. Namun, penulis mohon maaf apabila terdapat kekurangan ataupun kesalahan yang penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan untuk ke depannya.

Surabaya, Juni 2020

Ariq Sudibyo

DAFTAR ISI

LEMBAR PENGESAHAN	vii
ABSTRAK	ix
ABSTRACT	xi
KATA PENGANTAR.....	xiii
DAFTAR ISI	xv
DAFTAR GAMBAR	xvii
DAFTAR LAMPIRAN	xix
DAFTAR TABEL	xxi
DAFTAR KODE SUMBER	xxiii
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Permasalahan.....	2
1.3. Batasan Permasalahan	2
1.4. Tujuan.....	2
1.5. Manfaat.....	3
1.6. Metodologi	3
1.7. Sistematika Penulisan.....	5
BAB II TINJAUAN PUSTAKA	7
2.1. Bahasa Pemrograman <i>C#</i>	7
2.2. <i>Game Engine Unity</i>	7
2.3. Realitas Virtual.....	7
2.4. Pemodelan 3D	8
2.5. <i>Windows Speech Recognition API</i>	8
2.6. Permainan Realitas Virtual <i>Shooter Panahan</i>	8
BAB III ANALISIS DAN PERANCANGAN SISTEM.....	11
3.1. Analisis Sistem	11
3.1.1. Spesifikasi Kebutuhan Sistem	11
3.1.2. Identifikasi Pengguna	13
3.2. Perancangan Sistem.....	13
3.2.1. Deskripsi Umum Sistem.....	13
3.2.2. Arsitektur Sistem.....	14
3.3. Perancangan Permainan	15
3.3.1. Alur Permainan.....	15

3.3.2.	Aturan Permainan	16
3.3.3.	Mekanisme Permainan	17
3.3.4.	<i>Immersivity</i> Permainan	25
3.4.	Perancangan <i>User Interface</i> dan Aset Permainan	27
3.4.1.	<i>User Interface</i>	27
3.4.2.	Perancangan Aset	27
3.5.	Perancangan Tampilan Antarmuka	35
3.5.1.	Tampilan Awal Permainan	35
3.5.2.	Tampilan Permainan	36
3.5.3.	Tampilan Akhir Permainan	37
BAB IV IMPLEMENTASI SISTEM.....		39
4.1.	Lingkungan Implementasi	39
4.2.	Implementasi Gameplay	40
4.2.1.	Implementasi Awal Permainan	40
4.2.2.	Implementasi Permainan	54
4.2.3.	Implementasi Akhir Permainan	65
BAB V PENGUJIAN DAN EVALUASI		69
5.1.	Lingkungan Pengujian	69
5.2.	Pengujian Fungsionalitas	69
5.2.1.	Uji Coba Awal Permainan	70
5.2.2.	Uji Coba Permainan	71
5.2.3.	Uji Coba Akhir Permainan	75
5.2.4.	Hasil Uji Coba	76
5.3.	Pengujian Pengguna	76
5.3.1.	Skenario Pengujian Pengguna	77
5.3.2.	Daftar Penguji Permainan	79
5.3.3.	Hasil Pengujian Pengguna	79
5.3.4.	Kritik dan Saran Pengguna	81
5.4.	Evaluasi Pengujian	82
BAB VI KESIMPULAN DAN SARAN.....		83
6.1.	Kesimpulan	83
6.2.	Saran	83
DAFTAR PUSTAKA.....		85
LAMPIRAN		87
BIODATA PENULIS.....		97

DAFTAR GAMBAR

Gambar 3.1 Arsitektur Sistem	14
Gambar 3.2 Ilustrasi Active Skill Multiple Arrow	18
Gambar 3.3 Ilustrasi Active Skill Explosive Arrow	19
Gambar 3.4 Ilustrasi Active Skill Magic Arrow	19
Gambar 3.5 Rancangan Tingkat Kesulitan Level 1	20
Gambar 3.6 Rancangan Tingat Kesulitan Level 2	21
Gambar 3.7 Rancangan Tingat Kesulitan Level 3	22
Gambar 3.8 Rancangan Tingat Kesulitan Level 4	23
Gambar 3.9 Rancangan Tingat Kesulitan Level 5	24
Gambar 3.10 Aset busur dan panah	27
Gambar 3.11 Aset Pohon	28
Gambar 3.12 Aset Monster Goblin	28
Gambar 3.13 Aset Health Point Bar	29
Gambar 3.14 Aset Ledakan	29
Gambar 3.15 Aset Langit	30
Gambar 3.16 Aset Pohon dan Tanah	30
Gambar 3.17 Aset Monster Dragon	31
Gambar 3.18 Aset Langit	31
Gambar 3.19 Aset Monster orc	32
Gambar 3.20 Aset Health dan Magic Potion	32
Gambar 3.21 Aset monster ogre	33
Gambar 3.22 Aset Langit	33
Gambar 3.23 Aset Salju	34
Gambar 3.24 Rancangan Antarmuka Tampilan Awal Permainan	35
Gambar 3.25 Rancangan Antarmuka Tampilan Permainan	36
Gambar 3.26 Rancangan Antarmuka Tampilan Akhir Permainan	37
Gambar 4.1 Implementasi Sasaran Objek Awal Permainan	40
Gambar 4.2 Implementasi Tampilan Cara Bermain	41

Gambar 4.3 Implementasi Tampilan Video Peragaan Menembak Panah	42
Gambar 4.4 Implementasi Busur Dan Panah	43
Gambar 4.5 Implementasi Multiply Arrow Active Skill.....	43
Gambar 4.6 Implementasi Explosion Arrow Active Skill.....	44
Gambar 4.7 Implementasi Magic Arrow Active Skill.....	44
Gambar 4.8 Implementasi Permainan	54
Gambar 4.9 Tampilan Obyek Health Dan Magic Bar	62
Gambar 4.10 Tampilan Credit Scene	66
Gambar 4.11 Tampilan Lose Scene.....	66

DAFTAR LAMPIRAN

Lampiran 1 Penguji Coba (1)	87
Lampiran 2 Penguji Coba (2)	87
Lampiran 3 Penguji Coba (3)	88
Lampiran 4 Penguji Coba (4)	88
Lampiran 5 Penguji Coba (5)	89
Lampiran 6 Penguji Coba (6)	89
Lampiran 7 Penguji Coba (7)	90
Lampiran 8 Penguji Coba (8)	90
Lampiran 9 Penguji Coba (9)	91
Lampiran 10 Penguji Coba (10)	91

[Halaman ini sengaja dikosongkan]

DAFTAR TABEL

Tabel 3.1 Kebutuhan Fungsional Sistem.....	11
Tabel 3.2 Kebutuhan Non Fungsional Sistem.....	13
Tabel 5.1 Tabel Lingkungan Pengujian Sistem.....	69
Tabel 5.2 Hasil Uji Coba Awal Permainan	70
Tabel 5.3 Hasil Uji Coba Permainan.....	72
Tabel 5.4 Hasil Uji Coba Akhir Permainan.....	75
Tabel 5.5 Hasil Evaluasi.....	76
Tabel 5.6 Rentang Nilai.....	77
Tabel 5.7 Format Kuesioner.....	78
Tabel 5.8 Daftar Penguji Permainan	79
Tabel 5.9 Hasil Pengujian Pengguna.....	80
Tabel 5.10 Hasil Akhir Pengujian Pengguna	81
Tabel 5.11 Kritik dan Saran Pengguna.....	82

[Halaman ini sengaja dikosongkan]

DAFTAR KODE SUMBER

Kode Sumber 4.1 Kode Sumber Objek Sasaran Permainan.....	41
Kode Sumber 4.2 Kode Sumber Busur Dan Active Skill.....	50
Kode Sumber 4.3 Kode Sumber Panah	51
Kode Sumber 4.4 Kode Sumber Speech Recognition.....	53
Kode Sumber 4.5 Kode Sumber Enemy Monster	57
Kode Sumber 4.6 Kode Sumber Controller Monster Goblin	57
Kode Sumber 4.7 Kode Sumber Controller Monster Orc	58
Kode Sumber 4.8 Kode Sumber Controller Monster Ogre	58
Kode Sumber 4.9 Kode Sumber Controller Monster Dragon	59
Kode Sumber 4.10 Kode Sumber Generate Level	60
Kode Sumber 4.11 Kode Sumber Informasi Jumlah Monster.....	61
Kode Sumber 4.12 Kode Sumber Health Dan Magic Point Manager.....	63
Kode Sumber 4.13 Kode Sumber Health Bar Slider	63
Kode Sumber 4.14 Kode Sumber Magic Bar Slider	64
Kode Sumber 4.15 Kode Sumber Health Dan Magic Potion.....	65
Kode Sumber 4.16 Kode Sumber Memulai Kembali Permainan	67
Kode Sumber 4.17 Kode Sumber Keluar Dari Permainan.....	67

[Halaman ini sengaja dikosongkan]

BAB I

PENDAHULUAN

1.1. Latar Belakang

Perkembangan dunia video game saat ini semakin pesat, salah satu genre permainan yang cukup populer adalah *shooter*. *Shooter* merupakan subgenre *action* video game, yang sering digunakan untuk menguji kesadaran spasial, refleks, dan kecekatan pemain [1]. Namun, sekarang kebanyakan game *shooter* masih berbasis komputer sehingga menyebabkan kesadaran spasial, refleks, dan kecekatan pemain tidak dapat diterapkan pada dunia nyata.

Kekurangan tersebut dapat diatasi dengan menggunakan teknologi realitas virtual. Realitas virtual merupakan teknologi yang memungkinkan pengguna untuk menyimulasikan kehadiran fisik pada lingkungan 3-dimensi virtual. Teknologi ini sering digunakan sebagai sarana simulasi kegiatan yang tidak dapat dilakukan di dunia nyata.

Speech recognition adalah bidang ilmu interdisipliner linguistik komputasi yang memungkinkan pengenalan dan terjemahan bahasa lisan ke dalam teks dengan komputer [2]. Teknologi ini memberikan pemain kesempatan untuk terlibat dengan lingkungan sekitarnya seolah-olah lingkungan sekitarnya adalah permainan itu sendiri [3].

Dalam Tugas Akhir ini, penulis ingin membuat sebuah permainan yang berjudul *Archer Master*. *Archer Master* merupakan sebuah permainan realitas virtual *shooter* panahan dengan *speech recognition* sebagai kendali untuk menggunakan kemampuan yang dimiliki pemain dalam permainan. Permainan ini menggunakan teknologi *Oculus Rift* untuk menampilkan realitas virtual. Teknologi ini telah dilengkapi mikrofon terintegrasi yang dapat digunakan sebagai perangkat masukan *speech recognition* pada video game.

1.2. Rumusan Permasalahan

Berikut adalah beberapa hal yang menjadi rumusan masalah dalam Tugas Akhir ini:

1. Bagaimana cara merancang tingkat kesulitan dan skenario pada level permainan *Archer Master* ?
2. Bagaimana cara merancang aturan main berbasis realitas virtual dengan perangkat *Oculus Rift*?
3. Bagaimana cara merancang mekanisme penyerangan pemain dengan menerapkan *Speech Recognition* untuk mengenali suara yang diucapkan oleh pemain?
4. Bagaimana metode pengujian dan evaluasi permainan *Archer Master*?

1.3. Batasan Permasalahan

Permasalahan yang dibahas pada Tugas Akhir ini memiliki batasan masalah sebagai berikut:

1. *Game Engine* yang digunakan adalah *Unity*.
2. Perangkat input dan output audio yang digunakan adalah mikrofon dan *headphone* yang terintegrasi pada *Oculus Rift*.
3. Menggunakan *Windows Speech Recognition API* sebagai perangkat *speech recognition*.
4. Permainan yang dibuat merupakan permainan yang bekerja di *Personal Computer* (PC) yang mumpuni.
5. Permainan yang dibuat merupakan aplikasi realitas virtual yang membutuhkan *Oculus Rift* dan *Oculus Touch*.

1.4. Tujuan

Tujuan dari penggerjaan tugas akhir ini adalah membuat permainan realitas virtual *Archer Master* yang menerapkan *speech recognition* sebagai kontrol pemain dalam melakukan serangan. Permainan ini menggunakan *Oculus Rift* sebagai perangkat untuk menampilkan realitas virtual, *Oculus Touch*

sebagai alat kendali, dan mikrofon yang terdapat pada *Oculus Rift* sebagai perangkat masukan *speech recognition*.

1.5. Manfaat

Manfaat dari hasil pembuatan Tugas Akhir ini adalah:

1. Memberikan media hiburan bagi pemain.
2. Memberikan nuansa aktifitas memanah pada permainan realitas virtual yang mengacu pada dunia nyata.
3. Memberikan pengalaman baru pada pemain permainan realitas virtual dengan menggunakan fitur *speech recognition*.
4. Melatih kesadaran spasial, refleks, kecekatan dan kemampuan pengambilan keputusan pemain.

1.6. Metodologi

Langkah-langkah yang ditempuh dalam penggerjaan tugas akhir ini adalah sebagai berikut:

1. Studi literatur

Pada studi literatur ini, akan dipelajari sejumlah referensi yang diperlukan dalam pembuatan aplikasi realitas virtual yaitu bahasa pemrograman *C#*, *game engine Unity3D*, Pemodelan 3D, *Windows Speech Recognition API*, dan Permainan realitas virtual *shooter* panahan. Referensi-referensi ini didapatkan dari buku, internet serta materi-materi kuliah yang berhubungan dengan sistem yang akan dibangun.

2. Analisis dan desain sistem

Pada tahap ini analisis dan pendefinisian kebutuhan sistem dilakukan untuk masalah yang dihadapi, terutama analisis terkait bagaimana pembuatan skenario yang akan

diterapkan pada sistem. Selanjutnya, perancangan sistem dilakukan dengan beberapa tahap sebagai berikut:

- a. Mempelajari dokumentasi dan tutorial Unity3D.
- b. Mempelajari konsep *speech recognition* dengan menggunakan *library Windows.Speech*.
- c. Mempelajari dokumentasi dan tutorial *Oculus Rift*.
- d. Perancangan *gameplay* realitas virtual *Archer Master* dengan menambahkan fitur *speech recognition*.

3. Implementasi sistem

Pada tahap ini akan dilakukan pembangunan sistem. Sistem yang dimaksud disini, yaitu permainan realitas virtual yang dibangun dengan menggunakan perangkat Unity 3D dan perangkat *Oculus Rift*.

4. Pengujian dan evaluasi

Pada tahap ini pengujian sistem dilakukan kepada pengguna secara langsung. Pengujian dan evaluasi sistem dilakukan untuk mengevaluasi hasil analisis program. Tahapan-tahapan dari pengujian tersebut adalah sebagai berikut:

- a. Pengujian aplikasi.
- b. Pengujian terhadap fitur-fitur yang terdapat di aplikasi.
- c. Hasil pengujian berupa evaluasi sistem, diukur melalui kuesioner tentang permainan realitas virtual *Archer Master*.

5. Penyusunan buku Tugas Akhir

Pada tahap ini proses dokumentasi dan pembuatan laporan dari seluruh konsep dilakukan. Dokumentasi meliputi tinjauan pustaka, metode, implementasi, proses yang telah dilakukan, pengujian, evaluasi dan hasil-hasil yang telah didapatkan selama penggerjaan tugas akhir.

1.7. Sistematika Penulisan

Buku tugas akhir ini berfungsi untuk memberikan gambaran dari pengerjaan tugas akhir. Selain itu, buku ini juga diharapkan dapat berguna bagi pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku tugas akhir terdiri dari beberapa bagian seperti berikut ini:

- | | |
|----------------|---|
| Bab I | Pendahuluan
Bab ini berisi latar belakang masalah, rumusan masalah, tujuan dan manfaat pembuatan tugas akhir, batasan masalah, metodologi yang digunakan, dan sistematika penyusunan tugas akhir. |
| Bab II | Tinjauan Pustaka
Bab ini menjelaskan beberapa pustaka-pustaka yang dijadikan penunjang yang berhubungan dengan pokok pembahasan yang mendasari pembuatan tugas akhir. |
| Bab III | Analisis dan Perancangan Sistem
Bab ini membahas tentang analisis dan perancangan sistem yang akan dibangun. |
| Bab IV | Implementasi Sistem
Bab ini membahas tentang bagaimana implementasi sistem dari analisis dan desain yang sudah dirancang. |
| Bab V | Pengujian dan Evaluasi
Bab ini membahas pengujian dari metode yang ditawarkan dalam tugas akhir untuk mengetahui kesesuaian metode dengan data yang ada. |
| Bab VI | Kesimpulan dan Saran
Bab ini berisi kesimpulan dari hasil pengujian yang telah dilakukan. Bab ini juga membahas saran-saran untuk pengembangan sistem lebih lanjut. |

Daftar Pustaka

Merupakan daftar referensi yang digunakan untuk mengembangkan tugas akhir.

Lampiran

Merupakan bab tambahan yang berisi data atau daftar istilah yang penting pada tugas akhir ini.

BAB II

TINJAUAN PUSTAKA

Pada bagian ini akan dibahas mengenai teori-teori yang menjadi dasar dari penyelesaian tugas akhir penulis. Pembahasan bertujuan untuk memberikan gambaran secara umum terhadap program yang akan dibuat dan berguna sebagai penunjang dalam pengembangan tugas akhir.

2.1. Bahasa Pemrograman C#

C# (dibaca: *C sharp*) merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh *Microsoft* sebagai bagian dari inisiatif kerangka *.NET Framework*. Bahasa pemrograman ini dibuat berbasiskan bahasa *C++* yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti *Java*, *Delphi*, *Visual Basic*, dan lain-lain) dengan beberapa penyederhanaan [4].

2.2. Game Engine Unity

Unity adalah salah satu *game engine* lintas *platform* yang dikembangkan oleh *Unity Technologies* dimana pertama kali diumumkan dan dirilis ke publik pada Juni 2005 di konferensi *Apple Inc* sebagai *game engine* eksklusif pada sistem operasi *Mac OS X*. *Game engine* ini dapat digunakan untuk membuat aplikasi permainan 2-dimensi, 3-dimensi, realitas virtual, realitas teraugmentasi dan juga simulasi. *Game engine* ini juga telah diadopsi diluar aplikasi permainan yaitu film, otomotif, arsitektur, rekayasa, dan konstruksi [5].

2.3. Realitas Virtual

Realitas Virtual adalah pengalaman simulasi yang dapat mirip atau sama sekali berbeda dari dunia nyata. Aplikasi realitas

virtual dapat mencakup hiburan) dan tujuan pendidikan. Saat ini sistem realitas virtual standar menggunakan headset realitas virtual atau lingkungan multi-proyeksi untuk menghasilkan gambar, suara, dan sensasi realistik lainnya yang mensimulasikan kehadiran fisik pengguna di lingkungan virtual. Seseorang yang menggunakan peralatan realitas virtual dapat melihat-lihat dunia buatan, bergerak di dalamnya, dan berinteraksi dengan fitur atau barang virtual [6].

2.4. Pemodelan 3D

Pemodelan 3D adalah prosedur pengembangan model 3 Dimensi menggunakan perangkat lunak khusus. Prosedur ini dilakukan sebagai proses untuk membuat model yang mewakili objek yang sebenarnya tiga dimensi. Objek yang dibuatkan modelnya dapat membentuk objek hidup atau mati. Sebuah model tiga dimensi yang dibuat dengan menggunakan titik bulat di dalam ruang 3D, yang diperdebatkan dengan berbagai data geometris seperti garis, bidang datar, dan permukaan melengkung yang menghasilkan bentuk [7].

2.5. Windows Speech Recognition API

Untuk menerapkan *Speech Recognition* pada *Unity*, dapat digunakan *Windows Speech Recognition API* dengan melakukan *import Unity.Windows.Speech* pada *script C# Unity*. Perangkat ini melakukan streaming *speech-to-text*, yang berarti mengumpulkan potongan audio kecil saat direkam dan mengembalikan hasil secara real time. Setelah jeda berbicara, pengenal akan menampilkan hasil text berdasarkan seluruh potongan pembicaraan [8]. Pada permainan ini terdapat 3 kata yang akan dikenali yaitu *Multiply*, *Explosion*, dan *Magic*.

2.6. Permainan Realitas Virtual Shooter Panahan

Di masa lalu, calon pemanah perlu membeli atau meminjam busur, dan memahami cara menggunakannya sebelum

menembakkan panah. Permainan realitas virtual memecah hambatan yang mungkin mengintimidasi orang dan mencegah mereka dari mencoba memanah. Permainan interaktif ini membutuhkan beberapa keterampilan dan gerakan, terutama dibandingkan dengan video game lama yang hanya membutuhkan sedikit gerakan jari tangan [9]. Terdapat beberapa permainan realitas virtual *shooter* panahan yang sudah ada diantaranya yaitu *Walk on Arrow*, *BARDO*, *QuiVr Vanguard*, *In Death*. Namun, diantara permainan tersebut belum ada yang menggunakan kendali *speech recognition*.

[Halaman ini sengaja dikosongkan]

BAB III

ANALISIS DAN PERANCANGAN SISTEM

Bab ini membahas tentang desain dan perancangan sistem realitas virtual untuk *Archer Master*. Pembahasan yang dilakukan meliputi analisis sistem, perancangan sistem, skenario simulasi, perancangan *user interface*, aset permainan, dan perancangan antar muka sistem.

3.1. Analisis Sistem

Sub bab ini akan membahas tentang analisis kebutuhan sistem, meliputi spesifikasi kebutuhan sistem, baik itu kebutuhan fungsional sistem maupun kebutuhan non-fungsional sistem, dan identifikasi pengguna sistem.

3.1.1. Spesifikasi Kebutuhan Sistem

Pada sistem ini terdapat beberapa kebutuhan fungsional dan kebutuhan non-fungsional yang mendukung berjalannya sistem. Kebutuhan fungsional sistem dapat dilihat pada Tabel 3.1, sedangkan kebutuhan non-fungsional sistem dapat dilihat pada Tabel 3.2.

Tabel 3.1 Kebutuhan Fungsional Sistem

Kode	Deskripsi
Fungsionalitas Awal Permainan	
F1	Pemain dapat menggunakan busur dan panah dengan kendali <i>Oculus Touch</i> .
F2	Pemain dapat melepasan panah dengan menekan tombol pada <i>Oculus Touch</i> .
F3	Pemain dapat menggunakan <i>Active Skill</i> dengan mengucapkan kata yang mewakili <i>Active Skill</i> tersebut.
F4	Pemain dapat melihat Informasi <i>Active Skill</i> .

F5	Pemain dapat memulai permainan.
Fungsionalitas Permainan	
F6	Panah yang dilepaskan pemain dapat mengurangi <i>health point monster</i> .
F7	Pemain dapat menyerang <i>monster</i> menggunakan <i>Active Skill</i>
F8	Pemain dapat mengalahkan <i>monster</i> .
F9	Pemain dapat menambah <i>health point</i> dan <i>magic point</i> dengan menembak panah pada <i>health potion</i> dan <i>magic potion</i> .
F10	Pemain dapat melihat jumlah <i>monster</i> yang masih hidup.
F11	Pemain memiliki <i>health point</i> dan <i>magic point</i> yang dapat berkurang.
F12	Pemain kalah saat <i>health point</i> mencapai 0.
F13	<i>Monster</i> dapat menyerang pemain.
F14	<i>Monster</i> dapat berjalan mendekati pemain dan menghindari halangan.
F15	Pemain dapat memenangkan permainan
Fungsionalitas Akhir Permainan	
F16	Pemain dapat keluar dari permainan setelah kalah atau menang.
F17	Pemain dapat memulai kembali permainan setelah kalah atau menang.

Tabel 3.2 Kebutuhan Non Fungsional Sistem

Kode	Deskripsi
NF1	Sistem dapat dijalankan pada Sistem Operasi Windows 10
NF2	Sistem memiliki tampilan antar muka yang mudah dipahami
NF3	Sistem dapat dijalankan pada perangkat komputer

3.1.2. Identifikasi Pengguna

Pengguna yang dapat memainkan permainan *Archer Master* adalah siapa saja (umum). Sehingga, pengguna berhak menggunakan seluruh fungsionalitas yang terdapat pada sistem.

3.2. Perancangan Sistem

Sub bab ini membahas tentang bagaimana sistem ini dirancang, meliputi deskripsi umum sistem, arsitektur sistem.

3.2.1. Deskripsi Umum Sistem

Archer Master merupakan permainan realitas virtual berbasis *personal computer* bergenre *survival shooter* yang memanfaatkan *Oculus Rift* dan *speech recognition* sebagai kendalinya. Sistem ini merupakan permainan realitas virtual yang menghadirkan atmosfer dan suasana ketika berada di dunia fantasi.

Pembangunan sistem ini dimulai dari membuat *terrain* yang bertema fantasi yang akan dijadikan sebagai lingkungan nyata di dalam permainan realitas virtual.

Proses pembangunan sistem selanjutnya yaitu dengan merancang gameplay yang sesuai dengan tema dari permainan ini. Mencari aset aset yang sesuai, dan melengkapi aset dengan membuat secara manual.

3.2.2. Arsitektur Sistem

Archer Master mengintegrasikan antara *Personal Computer* (PC) dan *Oculus Rift*. Permainan diproses di PC dan ditampilkan di *Oculus Rift*. Dengan menggunakan *Microphone* yang terdapat pada *Oculus Rift*, permainan *Archer Master* dapat mendengarkan apa yang diucapkan pemain. Lalu dengan *Oculus Touch* pemain dapat mengarahkan panah dan busur yang ada di dalam permainan. Dengan begini, permainan *Archer Master* lebih terasa nyata, karena menggunakan tangan dan suara pemain untuk berinteraksi dalam permainan.

Seperti yang dijelaskan diatas, Gambar 3.1 berikut merupakan tampilan diagram dari alur sistem permainan *Archer Master*.



Gambar 3.1 Arsitektur Sistem

Dari *player*, sistem menerima input berupa perintah suara melalui *microphone* *Oculus Rift* dan isyarat gerakan tangan melalui *Oculus Touch*, lalu diproses di PC untuk dijadikan data, hasil dari pemrosesan data tersebut di keluarkan berupa tampilan 3D realitas virtual dan efek suara melalui *Oculus Rift*.

3.3. Perancangan Permainan

Pada sub bab ini menjelaskan tentang skenario permainan untuk menentukan kondisi menang atau kalah. Selain itu akan dibahas pula aturan permainan.

3.3.1. Alur Permainan

Alur permainan dari *Archer Master* antara lain:

1. Saat permainan dijalankan maka pemain akan melihat layar cara bermain dan objek sasaran yang dapat ditembak untuk memulai permainan.
2. Setelah menembak objek sasaran, maka pemain akan menuju level pertama permainan.
3. Di dalam permainan terdapat obyek-obyek yang dapat berinteraksi dengan pemain diantaranya adalah *monster*, busur ,panah , *health potion* dan *magic potion*.
4. *Monster* akan bergerak menuju pemain lalu menyerang pemain saat berada dalam jarak serang *monster*.
5. Untuk menuju level selanjutnya, pemain diharuskan untuk mengalahkan semua *monster* yang ada pada level tersebut.
6. Pemain dapat menggunakan *Active Skill* dengan mengucapkan kata yang mewakili sebuah *Active Skill*. Kata yang diucapkan pemain akan dikenali oleh *speech recognition* pada permainan. *Active Skill* digunakan untuk mengalahkan *monster* dengan mudah.
7. Pemain dapat mengalahkan *monster* dengan menembak *monster* menggunakan panah atau *Active Skill* hingga *health* pada *monster* mencapai 0.
8. Pemain dapat menambah *health point* dan *magic point* dengan menembak panah pada *health potion* dan *magic potion*.
9. Pemain dikatakan kalah apabila *health point* pemain mencapai 0.
10. Pemain dikatakan menang apabila telah menyelesaikan level terakhir permainan.

11. Setelah pemain kalah atau menang, pemain dapat memulai permainan kembali atau keluar dari permainan.

3.3.2. Aturan Permainan

Dalam memainkan permainan ini, terdapat aturan sebagai berikut:

1. Pada awal permainan, *health point* dan *magic point* pemain akan berjumlah 100.
2. Kondisi kalah jika *health point* habis (berkurang hingga 0)
3. *Active Skill* merupakan jurus yang dapat digunakan pemain jika *magic point* mencukupi. Cara menggunakan *Active Skill* yaitu dengan mengucapkan kata yang mewakili sebuah *Active Skill*. Kata tersebut akan dikenali oleh *speech recognition* lalu *Active Skill* yang sesuai akan diaktifkan.
4. Penggunaan *Active Skill* mengurangi *magic point*.
5. *Health Potion* jika ditembak oleh pemain dapat menambah *Health Point*.
6. *Magic Potion* jika ditembak oleh pemain dapat menambah *Magic Point*.
7. Pemain dapat melihat ke segala arah , menembak panah , dan menembak *Active Skill*.
8. Pemain harus dapat menjaga *health point* agar tidak habis.
9. Pemain harus mengalahkan semua monster untuk mencapai level selanjutnya.
10. *Monster* merupakan musuh yang dapat mengejar dan menyerang pemain.
11. *Monster* tersebar diseluruh tempat.
12. Jumlah *monster* yang ada dapat dilihat oleh pemain.
13. Besar *health point monster* tidak diketahui oleh pemain.
14. Untuk melakukan interaksi, pemain harus melakukan gerakan seperti memanah di dunia nyata dan/atau mengucapkan *Active Skill* yang ada.

3.3.3. Mekanisme Permainan

1. Pemain

- a. Memiliki *Health Point (HP)* berjumlah 100. Berkurang jika terkena serangan *monster*.
- b. Memiliki *Magic Point (MP)* berjumlah 100. *Magic Point* akan bertambah sebesar 20 tiap 5 detik.
- c. Pemain dapat menggunakan *Active Skill* jika memiliki jumlah *magic point* yang dibutuhkan.
- d. Pemain dapat menyerang *monster* dengan menarik panah pada busur, lalu menekan tombol pada jari telunjuk *Oculus Touch* kanan.
- e. Untuk menggunakan *Active Skill*. Pemain diharuskan mengucapkan kata *Multiply*, *Explosion*, atau *Magic*. Kata ini akan dikenali oleh *speech recognition* pada permainan dan mengaktifkan *Active Skill* yang sesuai dengan kata yang dikenali.

2. Busur dan Panah

- a. Panah dapat dilepaskan jika jarak antara *Oculus Touch* kiri dan kanan lebih dari $0.25f$.
- b. Setelah dilepaskan, panah akan terisi kembali jika jarak *Oculus Touch* kiri dan kanan kurang dari $0.25f$ dengan durasi *cooldown* 0.25 detik.
- c. Jika panah terkena *monster* akan mengurangi *health point* *monster* sebesar 10.

3. Health Potion dan Magic Potion

- a. *Health Potion* menambah *health point* pemain sebesar 20 jika ditembak oleh pemain.
- b. *Magic Potion* menambah *magic point* pemain sebesar 20 jika ditembak oleh pemain.

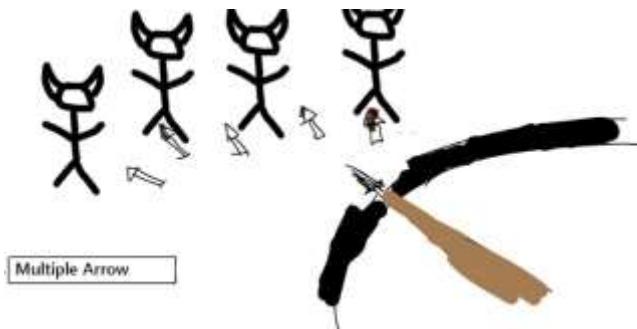
4. Monster

- a. Goblin
 - Memiliki 10 HP.
 - Memiliki kecepatan bergerak sebesar 1.5.
 - Bergerak mengejar pemain.
 - Serangan jarak dekat menghasilkan 5 damage.

- b. Orc
 - Memiliki 20 HP.
 - Memiliki kecepatan bergerak sebesar 1.5.
 - Bergerak mengejar pemain.
 - Serangan jarak dekat menghasilkan 10 damage.
- c. Ogre
 - Memiliki 60 HP.
 - Memiliki kecepatan bergerak sebesar 1.5.
 - Bergerak mengejar pemain.
 - Serangan jarak dekat menghasilkan 15 damage.
- d. Dragon
 - Memiliki 30 HP.
 - Memiliki kecepatan bergerak 5.
 - Bergerak mengejar pemain.
 - Serangan jarak jauh menghasilkan 15 damage.

5. Active Skill

- a. *Active Skill Multiple Arrow*

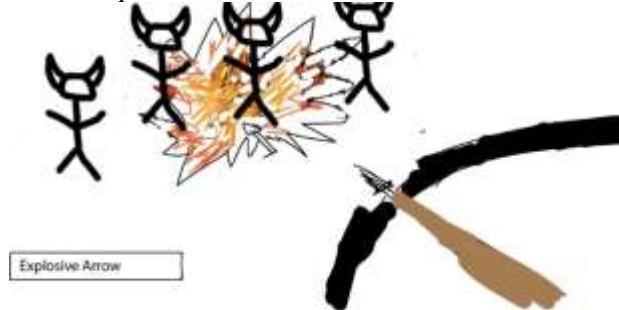


Gambar 3.2 Ilustrasi Active Skill Multiple Arrow

- Menembakkan 5 panah secara bersamaan. Untuk lebih jelasnya dapat dilihat pada Gambar 3.2.
- Membutuhkan 20 *magic point*.

- Diaktifkan dengan mengucap kata “Multiply” yang akan dikenali oleh *speech recognition*.
- Menghasilkan 10 *damage* pada *monster*.

b. *Active Skill Explosive Arrow*



Gambar 3.3 Ilustrasi Active Skill Explosive Arrow

- Menembakkan panah eksplosif. Untuk lebih jelasnya dapat dilihat pada Gambar 3.3.
- Membutuhkan 30 *magic point*.
- Diaktifkan dengan mengucap kata “*Explosion*” yang akan dikenali oleh *speech recognition*.
- Menghasilkan 30 *damage* pada *monster* dalam area ledakan.

c. *Active Skill Magic Arrow*

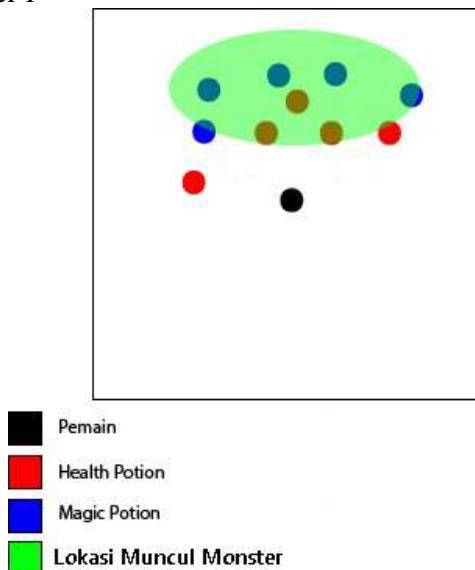


Gambar 3.4 Ilustrasi Active Skill Magic Arrow

- Menembakkan 15 panah selama 3 detik. Untuk lebih jelasnya dapat dilihat pada Gambar 3.4.
- Membutuhkan 50 *magic point*.
- Diaktifkan dengan mengucap kata “*Magic*” yang akan dikenali oleh *speech recognition*.
- menghasilkan 20 *damage* pada *monster*.

6. Level

a. Level 1

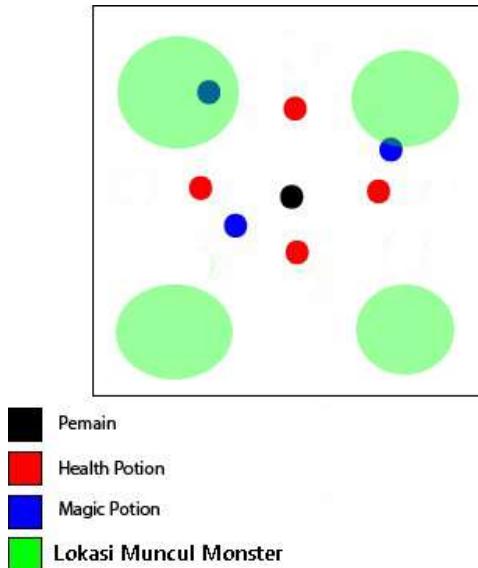


Gambar 3.5 Rancangan Tingkat Kesulitan Level 1

- Monster hanya muncul di depan pemain.
- Monster 8 goblin.
- Terdapat 5 Health Potion.
- Terdapat 5 Magic Potion.
- Lebih jelasnya dapat dilihat pada Gambar 3.5.

- Koordinat musuh yang muncul adalah koordinat acak pada area yang telah ditentukan.

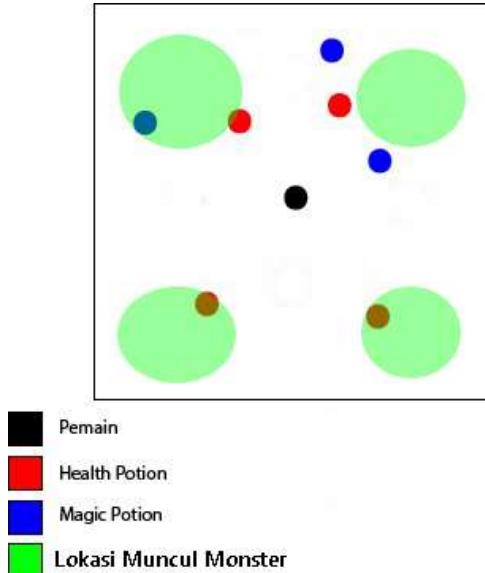
b. Level 2



Gambar 3.6 Rancangan Tingat Kesulitan Level 2

- Monster 5 goblin dan 3 orc.
- Terdapat 4 Health Potion.
- Terdapat 3 Magic Potion.
- Lebih jelasnya dapat dilihat pada Gambar 3.6.
- Koordinat kemunculan monster adalah koordinat acak pada area yang telah ditentukan.
- Jumlah monster yang muncul pada area yang telah ditentukan adalah acak.

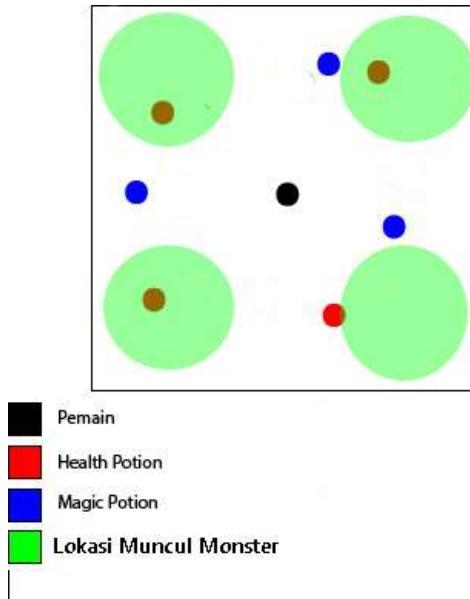
c. Level 3



Gambar 3.7 Rancangan Tingat Kesulitan Level 3

- Monster 3 goblin, 5 orc, dan 2 ogre.
- Terdapat 4 Health Potion.
- Terdapat 3 Magic Potion.
- Lebih jelasnya dapat dilihat pada Gambar 3.7.
- Koordinat kemunculan monster adalah koordinat acak pada area yang telah ditentukan.
- Jumlah monster yang muncul pada area yang telah ditentukan adalah acak.

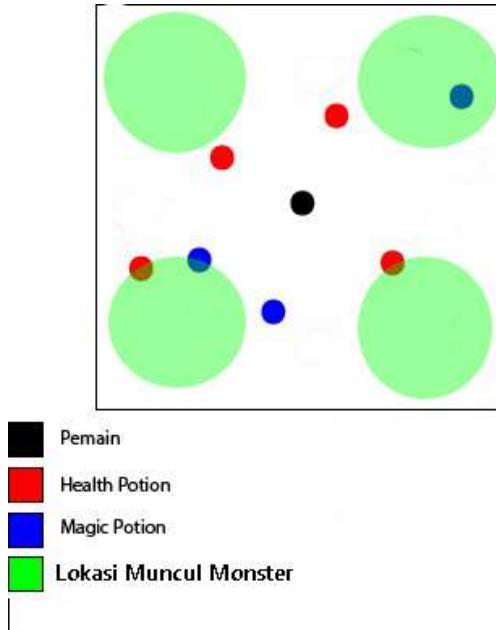
d. Level 4



Gambar 3.8 Rancangan Tingat Kesulitan Level 4

- Monster 5 orc, 4 ogre dan 1 dragon.
- Terdapat 4 Health Potion.
- Terdapat 3 Magic Potion.
- Lebih jelasnya dapat dilihat pada Gambar 3.8.
- Koordinat kemunculan monster adalah koordinat acak pada area yang telah ditentukan.
- Jumlah monster yang muncul pada area yang telah ditentukan adalah acak.

e. Level 5



Gambar 3.9 Rancangan Tingat Kesulitan Level 5

- Monster 15 orc, 3 ogre dan 2 dragon.
- Terdapat 4 Health Potion.
- Terdapat 3 Magic Potion.
- Lebih jelasnya dapat dilihat pada Gambar 3.9.
- Koordinat kemunculan monster adalah koordinat acak pada area yang telah ditentukan.
- Jumlah monster yang muncul pada area yang telah ditentukan adalah acak.

3.3.4. *Immersivity* Permainan

Immersion pada realitas virtual adalah persepsi hadir secara fisik di dunia realitas virtual. Persepsi ini dibuat dengan mengelilingi pengguna realitas virtual dengan gambar, suara atau rangsangan lain yang menyediakan lingkungan total secara menarik [10]. Pada permainan *Archer Master* terdapat detil yang dibuat penulis agar permainan bersifat *immersive* diantaranya yaitu:

1. Visibilitas Pemain
 - a. Dengan menggunakan paket integrasi Oculus pada Unity, pandangan pemain akan berpindah sesuai masukan gerakan perangkat *Oculus Rift*. Pemain dapat melihat ke segala arah seolah-olah berada di dalam dunia permainan tersebut.
2. Busur dan Panah
 - a. Dengan menggunakan paket integrasi Oculus pada Unity, Busur akan bergerak dan berputar sesuai masukan gerakan perangkat *Oculus Touch Kiri*.
 - b. Untuk menarik panah pada busur, pemain diharuskan mendekatkan tangan kiri dan kanan, menekan tombol *Oculus Touch Kanan* lalu menjauhkan kedua tangan. Gerakan ini merupakan bentuk persamaan dengan gerakan seorang pemanah sedang menarik panah pada tali busur panah. Setelah panah ditarik, pemain dapat melepas tombol *Oculus Touch Kanan* yang sedang ditekan untuk melepaskan anak panah
 - c. Panah akan melesat sesuai lintasan yang diarahkan oleh pemain, kecepatan anak panah dihitung dengan cara melakukan perkalian antara jauh tarikan panah pemain dalam skala 0 hingga 1 dan kecepatan maksimal yang telah ditentukan penulis.

- d. Jika panah mengenai *monster*, maka panah akan menghasilkan *damage* sesuai perkalian antara jauh tarikan panah pemain dalam skala 0 hingga 1 dan nilai 10.
 - e. Setelah panah dilepaskan, pemain diharuskan mendekatkan kedua *Oculus Touch* untuk dapat menarik panah kembali.
3. Speech Recognition
 - a. Dengan menggunakan *Windows Speech Recognition API*, permainan dapat mengenali kata yang diucapkan oleh pemain. Ketika pemain mengucapkan kata *Multiply*, *Explosion* atau *Magic*, permainan akan mengenali kata tersebut lalu mengaktifkan *Active Skill Multiple Arrow*, *Explosive Arrow*, atau *Magic Arrow*. Dalam hal ini seolah-olah suara pemain dapat memanggil kemampuan khusus berupa *Active Skill*.

3.4. Perancangan *User Interface* dan Aset Permainan

Pada sub bab ini menjelaskan tentang *user interface* dan aset yang terdapat dalam permainan. Selain itu akan dibahas pula penggunaannya pada permainan.

3.4.1. *User Interface*

User interface pada permainan ini antara lain:

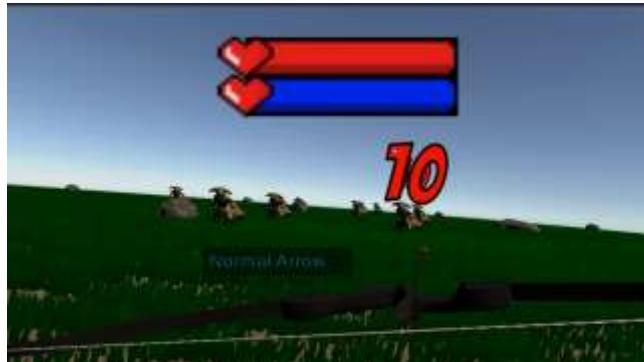
1. Panel *health bar* dan *magic bar*
2. Panel Informasi *Active Skill* dan jumlah *monster*

3.4.2. Perancangan Aset

Semua aset 3D merupakan aset yang diambil dari Unity Asset Store dan Internet yaitu:

1. *_BowAndArrow*

Digunakan untuk menampilkan busur dan panah yang terdapat dalam permainan, seperti ditampilkan pada Gambar 3.10.



Gambar 3.10 Aset busur dan panah

2. Connifers [BOTD]

Digunakan untuk menampilkan pohon yang terdapat dalam permainan, seperti ditampilkan pada gambar 3.11.



Gambar 3.11 Aset Pohon

3. Goblin

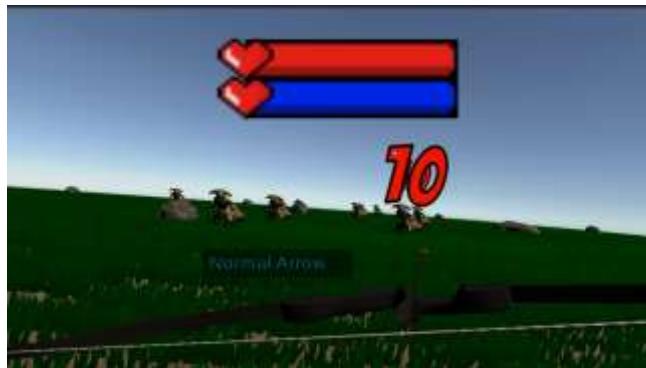
Digunakan untuk menampilkan *monster goblin* yang terdapat dalam permainan. Seperti yang ditampilkan pada Gambar 3.12.



Gambar 3.12 Aset Monster Goblin

4. HealthBar

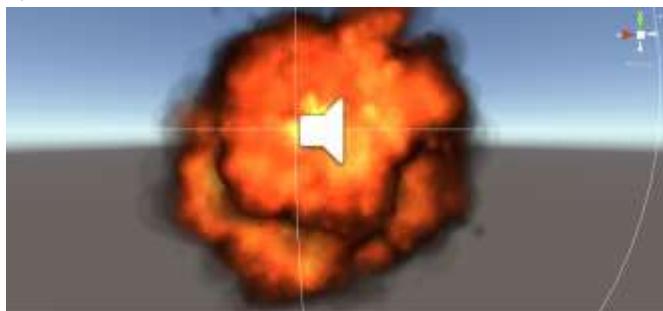
Digunakan untuk menampilkan *Health Point Bar* yang terdapat dalam permainan. Seperti yang ditampilkan pada Gambar 3.13.



Gambar 3.13 Aset Health Point Bar

5. BarrelModel

Digunakan untuk menampilkan efek ledakan yang terdapat dalam permainan. Seperti yang ditunjukkan pada Gambar 3.14.



Gambar 3.14 Aset Ledakan

6. ColorSkies

Digunakan untuk menampilkan langit yang terdapat dalam permainan. Seperti yang ditunjukkan pada Gambar 3.15.



Gambar 3.15 Aset Langit

7. Fantasy Forest Environment Free Sample

Digunakan untuk menampilkan pohon dan tanah yang terdapat dalam permainan. Seperti yang ditunjukkan pada Gambar 3.16.



Gambar 3.16 Aset Pohon dan Tanah

8.FourEvilDragonsHP

Digunakan untuk menampilkan *monster dragon* yang terdapat dalam permainan. Seperti yang ditunjukkan pada Gambar 3.17.



Gambar 3.17 Aset Monster Dragon

9.UrbanNightSky

Digunakan untuk menampilkan langit yang terdapat dalam permainan. Seperti yang ditunjukkan pada Gambar 3.18.



Gambar 3.18 Aset Langit

10.Mixamo

Digunakan untuk menampilkan *monster orc* yang terdapat dalam permainan. Seperti yang ditunjukkan pada Gambar 3.19.



Gambar 3.19 Aset Monster orc

11.RPG Pack

Digunakan untuk menampilkan *Health* dan *Magic Potion* yang terdapat dalam permainan. Seperti yang ditunjukkan pada Gambar 3.20.



Gambar 3.20 Aset Health dan Magic Potion

12.Troll_Canibal

Digunakan untuk menampilkan *monster ogre* yang terdapat dalam permainan. Seperti yang ditunjukkan pada Gambar 3.21.



Gambar 3.21 Aset monster ogre

13._DesertKits64_Sample

Digunakan untuk menampilkan tanah dan bangunan yang terdapat dalam permainan. Seperti yang ditunjukkan pada Gambar 3.22.



Gambar 3.22 Aset Langit

14. Winter Zone Mini 1.0

Digunakan untuk menampilkan tanah dan bangunan yang terdapat dalam permainan. Seperti yang ditunjukkan pada Gambar 3.23.



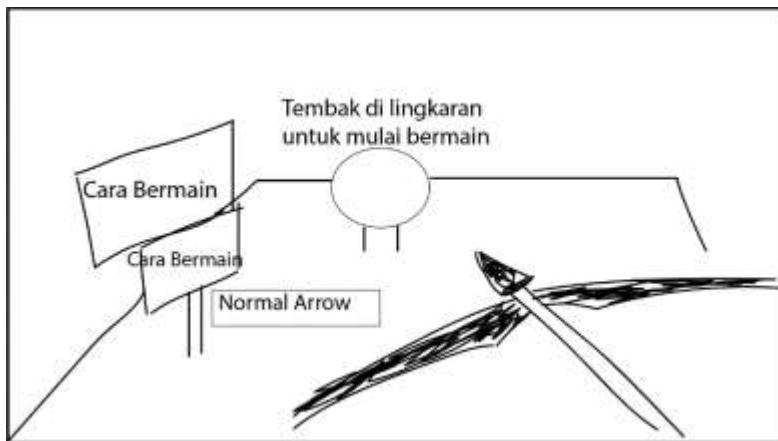
Gambar 3.23 Aset Salju

3.5. Perancangan Tampilan Antarmuka

Sub bab ini membahas bagaimana rancangan antarmuka pengguna yang digunakan pada permainan *Archer Master*. Rancangan antarmuka yang dibahas meliputi ketentuan masukan dan rancangan halaman tampilan

3.5.1. Tampilan Awal Permainan

Tampilan menu awal merupakan tampilan yang pertama kali muncul ketika aplikasi dijalankan. Pada tampilan awal terdapat tampilan cara bermain dan objek sasaran untuk memulai permainan. Tampilan rancangan antarmuka menu awal dapat dilihat pada Gambar 3.24.



Gambar 3.24 Rancangan Antarmuka Tampilan Awal Permainan

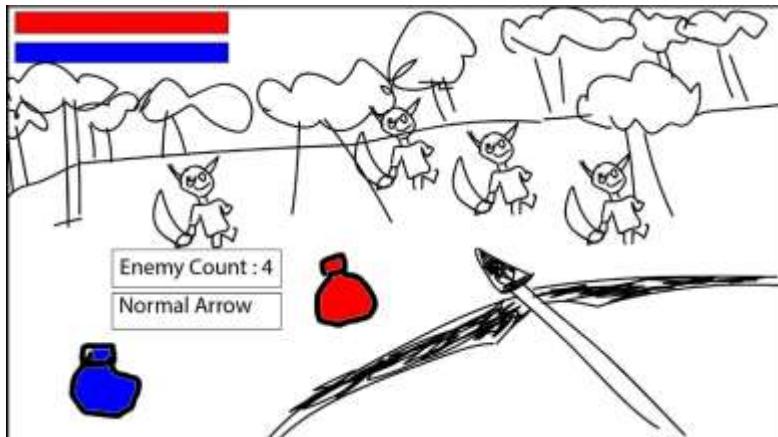
Berikut penjelasan dari Gambar 3.24:

1. Menampilkan objek dengan teks cara bermain yang dapat dilihat oleh pemain.
2. Menampilkan objek sasaran tembak untuk mulai bermain.
3. Panel Normal Arrow, menampilkan informasi *Active Skill* pemain.

4. Busur dan panah, senjata yang digunakan pemain untuk menyerang *monster*.

3.5.2. Tampilan Permainan

Tampilan permainan merupakan halaman yang muncul setelah pemain menembak objek sasaran pada tampilan awal permainan. Tampilan ini merupakan halaman dimana pemain mulai melakukan interaksi terhadap dunia virtual. Tampilan rancangan antarmuka permainan dapat dilihat pada Gambar 3.25.



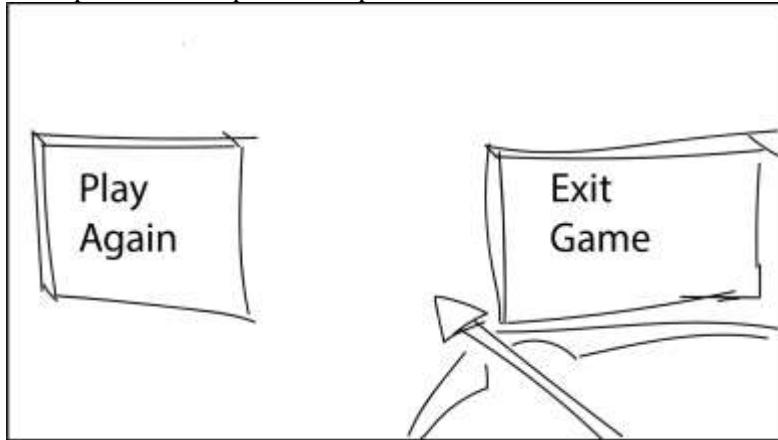
Gambar 3.25 Rancangan Antarmuka Tampilan Permainan

Berikut penjelasan Gambar 3.25:

1. Menampilkan objek *monster*.
2. Panel Enemy Count, menampilkan informasi jumlah *monster* yang ada.
3. Panel Health dan Magic Point Bar, menampilkan *health* dan *magic point* pemain.
4. Menampilkan Objek Health Potion dan Magic Potion.

3.5.3. Tampilan Akhir Permainan

Tampilan akhir permainan merupakan halaman ketika pemain mati karena *health point* habis atau ketika pemain telah menyelesaikan level terakhir. Tampilan rancangan antarmuka akhir permainan dapat dilihat pada Gambar 3.26 berikut.



Gambar 3.26 Rancangan Antarmuka Tampilan Akhir Permainan

Berikut penjelasan Gambar 3.26:

1. Objek **Play Again**, berfungsi untuk kembali ke tampilan awal permainan saat ditembak menggunakan panah.
2. Objek **Exit Game**, berfungsi untuk keluar dari permainan saat ditembak menggunakan panah.

[Halaman ini sengaja dikosongkan]

BAB IV

IMPLEMENTASI SISTEM

Bab ini membahas mengenai implementasi dari perancangan sistem yang telah dibuat. Di dalamnya mencakup proses penerapan dan pengimplementasian dalam bentuk *code*, dan antarmuka yang mengacu pada rancangan yang telah dibahas sebelumnya.

4.1. Lingkungan Implementasi

Lingkungan implementasi sistem yang digunakan untuk mengembangkan tugas akhir ini memiliki spesifikasi perangkat keras dan perangkat lunak yang ditunjukkan oleh Tabel 4.1.

Tabel 4. 1 Spesifikasi Perangkat

Perangkat	Spesifikasi
Perangkat Keras	<ul style="list-style-type: none">• Prosesor: Intel® Core™ i7-7700K CPU @ 4.20GHz (8 CPUs), ~4.2GHz• Memori: 16384MB• VGA : NVIDIA GeForce GTX 1060 6Gb• Oculus Rift S• Oculus Touch
Perangkat Lunak	<ul style="list-style-type: none">• Sistem Operasi Microsoft Windows 10 64-bit• Perangkat Pengembang Unity3D 2019.2.11f1• Perangkat Pembantu Visual Studio Community 2019, Microsoft Word 2016, Adobe Premiere, Adobe Photoshop dan Blender.

4.2. Implementasi Gameplay

Implementasi dari semua fungsi diwujukan dalam bentuk *code* dengan Bahasa pemrograman C#. Implementasi fungsi diurut berdasarkan alur yang ada pada permainan *Archer Master*.

4.2.1. Implementasi Awal Permainan

Tampilan ini merupakan halaman dimana pemain mulai melakukan interaksi terhadap dunia virtual. Pada halaman awal permainan, pemain dapat melihat ke segala arah, menembak panah dan menggunakan *Active Skill*. Tampilan awal permainan yang diimplementasikan dapat dilihat pada Gambar 4.1-4.7.

Pada tampilan awal permainan terdapat 5 obyek yaitu:

1. Objek sasaran untuk memulai permainan, berupa objek 3D *Cylinder* yang dapat dilihat pada Gambar 4.1. Pemain diharuskan menembak panah pada objek sasaran untuk dapat memulai permainan. Implementasi fungsi objek sasaran memulai permainan dapat dilihat dari Kode Sumber 4.1.



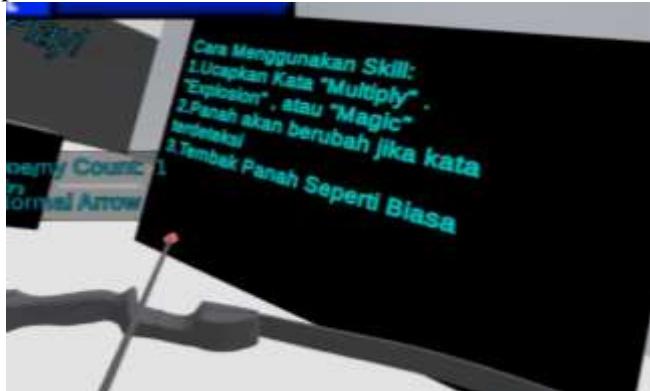
Gambar 4.1 Implementasi Sasaran Objek Awal Permainan

```
1. public class StartGame : MonoBehaviour, IDamageable
2. {
3.     public int health;
```

```
4.     public void Damage(int amount, float
5.         dmgmultiplier)
6.     {
7.         health -= amount;
8.         if (health <= 0)
9.         {
10.             SceneManager.LoadScene("level1");
11.         }
12.     }
```

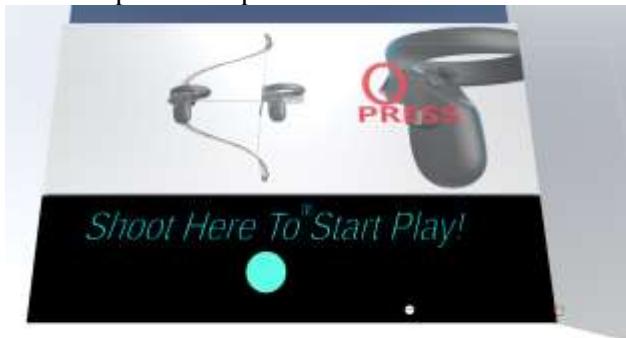
Kode Sumber 4.1 Kode Sumber Objek Sasaran Permainan

2. Objek 3D *Cube* dengan tampilan teks cara menembak panah dan menggunakan *Active Skill* yang dapat dilihat pada Gambar 4.2.



Gambar 4.2 Implementasi Tampilan Cara Bermain

3. Video peragaan menembak panah menggunakan Oculus Touch dapat dilihat pada Gambar 4.3



Gambar 4.3 Implementasi Tampilan Video Peragaan Menembak Panah

4. Obyek Busur, *Active Skill*, dan Informasi *Active Skill*.

Busur merupakan senjata pemain yang dikendalikan oleh Oculus Touch yang dipegang pemain pada tangan kiri. *Active Skill* merupakan serangan tambahan yang dapat digunakan pemain dengan cara mengucapkan kata yang mewakili sebuah *Active Skill*. *Active Skill* yang sedang digunakan dapat dilihat pada obyek Informasi *Active Skill*. Implementasi fungsi busur, fungsi untuk merubah text Informasi *Active Skill*, dan fungsi *Active Skill* dapat dilihat dari Kode Sumber 4.2. Implementasi Busur, *Active Skill*, dan Informasi *Active Skill* dapat dilihat pada Gambar 4.4-4.7.



Gambar 4.4 Implementasi Busur Dan Panah



Gambar 4.5 Implementasi Multiply Arrow Active Skill



Gambar 4.6 Implementasi Explosion Arrow Active Skill



Gambar 4.7 Implementasi Magic Arrow Active Skill

```

1. public class Bow : MonoBehaviour
2. {
3.     private void Start()
4.     {
5.         StartCoroutine(CreateArrow(0.0f));
6.     }
7.     private void Update()
8.     {
9.         if (!righthand)
10.        {
11.            righthand= GameObject.Find("RightHandAn
chor").transform;

```

```
12.        }
13.        handdistance = CalculatePull(righthand);
14.        if (m_PullingHand == null && handdistance
15.            < 0.25f &&
16.            (OVRInput.GetDown(OVRInput.Button.PrimaryIndexTrigger,
17.                m_Controller)))
18.        {
19.            m_PullingHand = GameObject.Find("RightH
20.                andAnchor").transform;
21.        }
22.        if (m_PullingHand != null)
23.        {
24.            Pull(m_OppositeController.transform);
25.            m_PullValue = CalculatePull(m_PullingHa
26.                nd);
27.            m_PullValue = Mathf.Clamp(m_PullValue,
28.                0.0f, 1.0f);
29.            m_Animator.SetFloat("Blend", m_PullValu
30.                e);
31.        }
32.        if (isSkill_1)
33.        {
34.            StartCoroutine(MultiArrow(0.2f));
35.            textku.GetComponent<TMP_Text>().text =
36.                "Multiple Arrow";
37.            isSkill_1 Released = false;
38.            isSkill_1 = false;
39.        }
40.        if (isSkill_2)
41.        {
42.            StartCoroutine(ExplodeArrow(0.2f));
43.            textku.GetComponent<TMP_Text>().text =
44.                "Explosive Arrow";
45.            isSkill_2 Released = false;
46.            isSkill_2 = false;
47.        }
48.        if (isSkill_3)
49.        {
50.            textku.GetComponent<TMP_Text>().text =
51.                "Magic Arrow";
52.            StartCoroutine(SpawnMagicArrow(0.0f));
53.            magicsound.PlayOneShot(magicclip);
54.            isSkill_3 Released = false;
```

```
45.             isSkill_3 = false;
46.         }
47.     If(OVRInput.GetUp(OVRInput.Button.PrimaryIndexTrigger, m_Controller))
48.         Release();
49.     }
50.
51.     private float CalculatePull(Transform pullHand)
52.     {
53.         Vector3 direction = m_End.position - m_Start.position;
54.         float magnitude = direction.magnitude;
55.         direction.Normalize();
56.         Vector3 difference = pullHand.position - m_Start.position;
57.         return Vector3.Dot(difference,direction)/magnitude;
58.     }
59.     IEnumerator MultiArrow(float waitTime)
60.     {
61.         GameObject arrowObject = Instantiate(m_ArrowPrefab, m_Socket);
62.         arrowObject.transform.localPosition = new Vector3(0, 0, 0.425f);
63.         arrowObject.transform.localEulerAngles = Vector3.zero;
64.         GameObject arrowObject2 = Instantiate(m_ArrowPrefab, m_Socket);
65.         arrowObject2.transform.localPosition = new Vector3(0, 0.2f, 0.425f);
66.         arrowObject2.transform.Rotate(-25, 0, 0);
67.         GameObject arrowObject3 = Instantiate(m_ArrowPrefab, m_Socket);
68.         arrowObject3.transform.localPosition = new Vector3(0, 0.4f, 0.425f);
69.         arrowObject3.transform.Rotate(-45, 0, 0);
70.         GameObject arrowObject4 = Instantiate(m_ArrowPrefab, m_Socket);
71.         arrowObject4.transform.localPosition = new Vector3(0, -0.2f, 0.425f);
72.         arrowObject4.transform.Rotate(25, 0, 0);
```

```
73.         GameObject arrowObject5 = Instantiate(m_Arr
owPrefab, m_Socket);
74.         arrowObject5.transform.localPosition = new
Vector3(0, -0.4f, 0.425f);
75.         arrowObject5.transform.Rotate(45, 0, 0);
76.         m_CurrentArrow = arrowObject.GetComponent<A
rrow>();
77.         m_CurrentArrow2 = arrowObject2.GetComponent
<Arrow>();
78.         m_CurrentArrow3 = arrowObject3.GetComponent
<Arrow>();
79.         m_CurrentArrow4 = arrowObject4.GetComponent
<Arrow>();
80.         m_CurrentArrow5 = arrowObject5.GetComponent
<Arrow>();
81.     }
82.     IEnumerator ExplodeArrow(float waitTime)
83.     {
84.         GameObject arrowObject = Instantiate(m_Expl
odeArrowPrefab, m_Socket);
85.         arrowObject.transform.localPosition = new V
ector3(0, 0, 0.425f);
86.         arrowObject.transform.localEulerAngles = Ve
ctor3.zero;
87.         m_CurrentArrow = arrowObject.GetComponent<A
rrow>();
88.     }
89.     IEnumerator CreateArrow(float waitTime)
90.     {
91.         yield return new WaitForSeconds(waitTime);

92.         GameObject arrowObject = Instantiate(m_Arr
owPrefab, m_Socket);
93.         arrowObject.transform.localPosition = new V
ector3(0, 0, 0.425f);
94.         arrowObject.transform.localEulerAngles = Ve
ctor3.zero;
95.         m_CurrentArrow = arrowObject.GetComponent<A
rrow>();
96.     }
97.
98.     IEnumerator MagicArrow(float magicpull)
99.     {
```

```
100.
101.            for (int i=0; i <15; i++)
102.            {
103.                GameObject MagicarrowObject = Instantiate
104.                    e(m_MagicArrowPrefab, m_Socket);
105.                MagicarrowObject.transform.localPosition
106.                    = new Vector3(0, 0, 0.425f);
107.                MagicarrowObject.transform.localEulerAngles
108.                    = Vector3.zero;
109.                m_CurrentArrow = MagicarrowObject.GetComponent<Arrow>();
110.                FireMagicArrow();
111.            }
112.        public void Pull(Transform hand)
113.        {
114.            float distance = Vector3.Distance(hand.position,
115.                m_Start.position);
116.            if (distance > m_GrabThreshold) { return; }
117.            m_PullingHand = hand;
118.        public void Release()
119.        {
120.            if (m_PullValue > 0.25f && isSkill_1_Releas
121.                ed && isSkill_2_Released && isSkill_3_Released)
122.                {
123.                    FireArrow();
124.                }
125.            else if(m_PullValue > 0.25f && !isSkill_1_
126.                Released && isSkill_2_Released && isSkill_3_Releas
127.                ed)
128.                {
129.                    FireMultiArrow();
130.                    isSkill_1_Released = true;
131.                }
132.            else if (m_PullValue > 0.25f && isSkill_1_R
133.                leased && !isSkill_2_Released && isSkill_3_Release
134.                d)
135.                {
136.                    FireExplodeArrow();
```

```
132.         isSkill_2_Released = true;  
133.     }  
134.     else if (135.     {  
136.         m_magicPull = m_PullValue;  
137.         m_CurrentArrow = null;  
138.         StartCoroutine(MagicArrow(magicPull));  
139.         isSkill_3_Released = true;  
140.     }  
141.     m_PullingHand = null;  
142.     m_PullValue = 0.0f;  
143.     m_Animator.SetFloat("Blend", 0);  
144.     if (!m_CurrentArrow && isSkill_1_Released && isSkill_2_Released && isSkill_3_Released)  
145.     {  
146.         StartCoroutine(CreateArrow(0.25f));  
147.     }  
148. }  
149. private void FireArrow()  
150. {  
151.     if(m_CurrentArrow == null)  
152.     {  
153.         Debug.Log("CURRENTARROW IS NULL");  
154.     }  
155.     if (isSkill_1_Released && isSkill_2_Release  
d && isSkill_3_Released) {  
156.         m_CurrentArrow.Fire(m_PullValue);  
157.         textku.GetComponent<TMP_Text>().text = "Nor  
mal Arrow";  
158.         arrowsound.PlayOneShot(arrowclip);  
159.         m_CurrentArrow = null;  
160.     }  
161.     if (!m_CurrentArrow && isSkill_1_Released &  
& isSkill_2_Released && isSkill_3_Released)  
162.     {  
163.         StartCoroutine(CreateArrow(0.25f));  
164.     }  
165. }  
166. }
```

Kode Sumber 4.2 Kode Sumber Busur Dan Active Skill

5. Obyek Panah

Panah merupakan senjata pemain yang dikendalikan oleh *Oculus Touch* yang dipegang pemain pada tangan kanan. Panah merupakan obyek yang akan ditembak oleh pemain untuk mengalahkan *monster*. Implementasi fungsi panah dapat dilihat dari Kode Sumber 4.3. Pada Kode Sumber 4.3 terdapat beberapa fungsi yaitu fungsi fisika menembak panah, fungsi untuk mendeteksi jika panah mengenai *monster*, fungsi melakukan *damage* pada *monster*, dan fungsi ledakan pada *Explosive Arrow*.

```
1. public class Arrow : MonoBehaviour
2. {
3.     private void FixedUpdate()
4.     {
5.         if (m_IsStopped)
6.             return;
7.         m_Rigidbody.MoveRotation(Quaternion.LookRotation(m_Rigidbody.velocity, transform.up));
8.         RaycastHit hit;
9.         if (Physics.Linecast(m_LastPosition, Tip.position, out hit))
10.        {
11.            Stop(hit.collider.gameObject);
12.        }
13.        m_LastPosition = Tip.position;
14.    }
15.    private void Start()
16.    {
17.        m_LastPosition = transform.position;
18.    }
19.    private void Stop(GameObject hitObject)
20.    {
21.        m_IsStopped = true;
22.        m_Rigidbody.isKinematic = true;
```

```

23.     m_Rigidbody.useGravity = false;
24.     CheckForDamage(hitObject);
25.     if(this.gameObject.CompareTag("Explosive"))
26.     {
27.         Debug.Log("EXPUROSSIONN");
28.         explodescript.StartExplode();
29.     }
30. }
31. private void CheckForDamage(GameObject hitObject)
32. {
33.     MonoBehaviour[] behaviours = hitObject.GetComponents<MonoBehaviour>();
34.     foreach(MonoBehaviour behaviour in behaviours)
35.     {
36.         if(behaviour is IDamageable)
37.         {
38.             IDamageable damageable = (IDamageable)behaviour;
39.             damageable.Damage(10, ,DmgMultiplier);
40.             break;
41.         }
42.     }
43. }
44. public void Fire(float PullValue)
45. {
46.     DmgMultiplier = PullValue;
47.     m_LastPosition = transform.position;
48.     m_IsStopped = false;
49.     transform.parent = null;
50.     m_Rigidbody.isKinematic = false;
51.     m_Rigidbody.useGravity = true;
52.     m_Rigidbody.AddForce(transform.forward * (PullValue * m_Speed));
53.     Destroy(gameObject, 5.0f);
54. }
55. }
```

Kode Sumber 4.3 Kode Sumber Panah

6. Obyek Speech Recognition

Agar permainan dapat mengenali kata yang diucapkan pemain guna mengaktifkan *Active Skill*, penulis menggunakan

library Windows.Speech, pada library ini ada beberapa fungsi yang digunakan seperti `onPhraseRecognizer`, `keyWordRecognizer`, dan lainnya. Untuk meminimalisir kesalahan pengenalan penulis memasukkan beberapa kata lain yang pengucapannya mirip dengan kata *Multiply*, *Explosion* dan *Magic*.

```
1. public class VoiceRecog : MonoBehaviour
2. {
3.     private KeywordRecognizer keywordRecognizer;
4.     private Dictionary<string, System.Action> actions
5.     = new Dictionary<string, Action>();
6.     public Bow MyBow;
7.     public PlayerManager playerManager;
8.     public void Awake()
9.     {
10.         actions.Add("magic", Magic);
11.         actions.Add("explosion", Explosion);
12.         actions.Add("multiply", Multiply);
13.         keywordRecognizer = new KeywordRecognizer(a
14.             ctions.Keys.ToArray());
15.         keywordRecognizer.OnPhraseRecognized += Rec
16.             ognizedSpeech;
17.             keywordRecognizer.Start();
18.     }
19.     void RecognizedSpeech(PhraseRecognizedEventArgs
20.         speech)
21.     {
22.         Debug.Log(speech.text);
23.         Action keywordAction;
24.         if (actions.TryGetValue(speech.text, out ke
25.             ywordAction))
26.         {
27.             Debug.Log(speech.text);
28.             keywordAction.Invoke();
29.         }
30.     }
31.     void Multiply()
32.     {
33.         if (playerManager.magic >= 20)
```

```
30.        {
31.            playerManager.magic -= 20;
32.            MyBow.isSkill_1 = true;
33.            Debug.Log("multiply");
34.        }
35.    }
36.    void Magic()
37.    {
38.        if (playerManager.magic >= 50)
39.        {
40.            playerManager.magic -= 50;
41.            MyBow.isSkill_3 = true;
42.            Debug.Log("magic");
43.        }
44.    }
45.
46.    void Explosion()
47.    {
48.        if (playerManager.magic >= 30)
49.        {
50.            playerManager.magic -= 30;
51.            MyBow.isSkill_2 = true;
52.            Debug.Log("explosion");
53.        }
54.    }
55.    void OnDestroy()
56.    {
57.        if (keywordRecognizer != null)
58.        {
59.            keywordRecognizer.Stop();
60.            keywordRecognizer.Dispose();
61.        }
62.    }
63. }
```

Kode Sumber 4.4 Kode Sumber Speech Recognition

4.2.2. Implementasi Permainan

Tampilan permainan merupakan halaman yang muncul setelah pemain menembak objek sasaran pada tampilan awal permainan. Implementasi tampilan permainan dapat dilihat pada Gambar 4.8.



Gambar 4.8 Implementasi Permainan

Terdapat beberapa obyek tambahan yang ada di halaman permainan, yaitu:

1. *Monster*
2. Informasi Jumlah *Monster*
3. *Health bar* pemain
4. *Magic bar* pemain
5. *Health Potion*
6. *Magic Potion*

Implementasi fungsi dari tiap obyek pada Gambar 4.8 dapat dilihat di Kode Sumber 4.5 – 4.14 Diantaranya yaitu,

1. Obyek *Monster*

Monster pada game ini terdiri dari 4 jenis monster yaitu *Goblin, Orc, Ogre dan Dragon*. Terdapat 3 Kode Sumber terkait perilaku *monster* yaitu:

- a. Kode Sumber *Enemy* memiliki fungsi mendeteksi *health point monster*, menampilkan teks jika panah mengenai *monster*, melakukan *Destroy* jika *health point monster* habis, mengurangi informasi jumlah *monster*, dan melakukan *damage* pada *health point* pemain. Setiap obyek *Monster* memiliki Kode Sumber *Enemy* yang sama. Implementasi fungsi dapat dilihat dari Kode Sumber 4.5.

```
2. public class Enemy : MonoBehaviour, IDamageable
3. {
4.     public void Damage(int amount)
5.     {
6.         health -= (int) (dmgmultiplier * amount);
7.         damageamount = (int) (dmgmultiplier * amount);
8.         if(health <= 0)
9.         {
10.             if (isCountReduced == false)
11.             {
12.                 countsscript.ReduceCount();
13.                 isCountReduced = true;
14.             }
15.
16.         }
17.         ShowText();
18.     }
19.     IEnumerator EnemyDead()
20.     {
21.         if (this.gameObject.GetComponent<CapsuleCo
llider>() != null) { this.gameObject.GetComponent<
CapsuleCollider>().isTrigger = true; }
22.         if (this.gameObject.GetComponent<BoxCollid
er>() != null ) { this.gameObject.GetComponent<Box
Collider>().isTrigger = true; }
23.         foreach (Renderer items in renderers)
24.             items.enabled = false;
25.         yield return new WaitForSeconds(0.1f);
```

```
26.         foreach (Renderer items in renderers)
27.             items.enabled = true;
28.         yield return new WaitForSeconds(0.1f);
29.         foreach (Renderer items in renderers)
30.             items.enabled = false;
31.         yield return new WaitForSeconds(0.1f);

32.         foreach (Renderer items in renderers)
33.             items.enabled = true;
34.         yield return new WaitForSeconds(0.5f);
35.
36.         Destroy(this.gameObject);
37.     }
38.     void Update()
39.     {
40.         if (health <=0)
41.         {
42.             health = 0;
43.         }
44.         if (health == 0)
45.         {
46.             isdead = true;
47.         }
48.         if (isdead)
49.         {
50.             StartCoroutine(EnemyDead());
51.             isdead = false;
52.         }
53.     }
54.     public void DecreaseHealth()
55.     {
56.         if(this.gameObject.name == "GOBLIN")
57.         {
58.             MyPlayer.DecreaseHealth(goblin);
59.         }
60.         if (this.gameObject.name == "ORC")
61.         {
62.             MyPlayer.DecreaseHealth(orc);
63.         }
64.         if (this.gameObject.name == "OGRE")
65.         {
66.             MyPlayer.DecreaseHealth(ogre);
```

```

67.        }
68.        if (this.gameObject.name == "DRAGON")
69.        {
70.            MyPlayer.DecreaseHealth(dragon);
71.        }
72.    }
73. }
```

Kode Sumber 4.5 Kode Sumber Enemy Monster

- b. Kode Sumber *Controller* memiliki fungsi mendekati lokasi pemain dengan menggunakan komponen NavMeshAgent, mendeteksi jarak antara pemain dan *monster*, berhenti mengejar jika jarak sudah dekat, dan mengaktifkan animasi menyerang. Kode Sumber *Controller* pada tiap *monster* berbeda karena perilaku *monster* yang berbeda-beda. Implementasi fungsi *controller* tiap *monster* dapat dilihat dari Kode Sumber 4.6-4.9.

```

1.  public class EnemyController : MonoBehaviour
2.  {
3.      void Update()
4.      {
5.          distance = Vector3.Distance(target.transfo
rm.position, this.gameObject.transform.position);
6.          if(distance < 6)
7.          {
8.              agent.isStopped = true;
9.              agent.velocity = Vector3.zero;
10.             agent.ResetPath();
11.             anim.SetInteger("condition", 0);
12.             anim.SetInteger("attack", 1);
13.         }
14.     }
15. }
```

Kode Sumber 4.6 Kode Sumber Controller Monster Goblin

```

1. public class OrcController : MonoBehaviour
2. {
3.     void Update()
4.     {
5.         distance = Vector3.Distance(target.transform
6.             .position, this.gameObject.transform.position);
7.         if (distance < 6f)
8.         {
9.             agent.isStopped = true;
10.            agent.stoppingDistance = 100f;
11.            agent.velocity = Vector3.zero;
12.            anim.ResetPath();
13.            anim.SetInteger("condition", 0);
14.            anim.SetInteger("attack", 1);
15.        }
16.    }

```

Kode Sumber 4.7 Kode Sumber Controller Monster Orc

```

1. public class OgreController : MonoBehaviour
2. {
3.     void Update()
4.     {
5.         agent.SetDestination(target.transform.posit
6.             ion);
7.         distance = Vector3.Distance(target.transform
8.             .position, this.gameObject.transform.position);
9.         if (distance < 8)
10.        {
11.            anim.SetInteger("condition", 0);
12.            anim.SetInteger("attack", 1);
13.        }
14.    }

```

Kode Sumber 4.8 Kode Sumber Controller Monster Ogre

```

1. public class DragonController : MonoBehaviour
2. {
3.     void Update()
4.     {
5.         agent.SetDestination(target.transform.position);
6.         distance = Vector3.Distance(target.transform.position, this.gameObject.transform.position);
7.         if (distance < 51)
8.         {
9.             anim.SetInteger("condition", 0);
10.            anim.SetInteger("attack", 1);
11.        }
12.    }
13.    public void Shoot()
14.    {
15.        spawned = Instantiate(fireballprefab.transform, fireballspawn.transform.position, Quaternion.identity);
16.    }
17. }
```

Kode Sumber 4.9 Kode Sumber Controller Monster Dragon

- c. Kode Sumber Generate Level memiliki fungsi untuk memunculkan monster dalam koordinat acak pada area yang ditentukan. Implementasi fungsi tersebut dapat dilihat pada Kode Sumber 4.10.

```

1. void Start()
2. {
3.     for (int i = 0; i < 5; i++)
4.     {
5.         float num = Random.Range(20, 50);
6.         if (num > 35)
7.         {
8.             num = 15 - num;
```

```

9.          }
10.         float num2 = Random.Range(20, 5
11.         0);
12.         if (num2 > 35)
13.         {
14.             num2 = 15 - num2;
15.             spawnedgoblin[i] = Instantiate(
16.                 prefabgoblin, GameObject.Find("Enemy").tran
17.                 sform);
18.                 spawnedgoblin[i].transform.Tran
19.                 slate(num, 0, num2);
20.                 Debug.Log(spawnedgoblin[i].tran
21.                 sform.position.ToString());
22.             }
23.         }
24.     }
25. }
```

Kode Sumber 4.10 Kode Sumber Generate Level

2. Obyek Infomasi Jumlah *Monster*

Informasi Jumlah *Monster* merupakan obyek berupa teks yang memberikan informasi jumlah *monster* yang masih hidup kepada pemain. Obyek ini juga memiliki fungsi untuk pindah ke level selanjutnya jika jumlah musuh habis. Implementasi fungsi informasi jumlah *monster* dapat dilihat pada Kode Sumber 4.11.

```

1. public class EnemyCount : MonoBehaviour
2. {
3.     public void ReduceCount()
4.     {
5.         enemycounts -= 1;
6.         counttext.text = enemycounts.ToString();
7.         if (enemycounts == 0)
8.         {
9.             if (string.Equals(CurrentScene, "level1"))
10.             {
11.                 StartCoroutine(LoadAsynchronous(2));
12.             }
13.             if (string.Equals(CurrentScene, "Level2New"))
```

```

14.          {
15.              StartCoroutine(LoadAsynchronous(3));
16.          }
17.      if (string.Equals(CurrentScene, "Level3New"))
18.      {
19.          StartCoroutine(LoadAsynchronous(4));
20.      }
21.      if (string.Equals(CurrentScene, "level4"))
22.      {
23.          StartCoroutine(LoadAsynchronous(5));
24.      }
25.      if (string.Equals(CurrentScene, "level5"))

26.      {
27.          StartCoroutine(LoadAsynchronous(7));

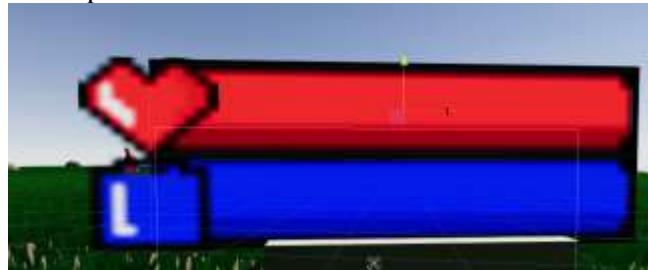
28.      }
29.  }
30. }
31. IEnumerator LoadAsynchronous(int sceneIndex)
32. {
33.     AsyncOperation operation = SceneManager.LoadSceneAsync(sceneIndex);
34.     loadingScreen.SetActive(true);
35.     while (!operation.isDone)
36.     {
37.         float progress = Mathf.Clamp01(operation.progress / .9f);
38.         slider.value = progress;
39.     }
40.     yield return null;
41. }
42. }
43. }
```

Kode Sumber 4.11 Kode Sumber Informasi Jumlah Monster

3. Health bar dan Magic Bar pemain

Obyek *health bar* dan *magic bar* pemain merupakan obyek *slider* yang memberikan informasi sisa *health point* dan *magic point* pemain yang tersisa. Obyek *health bar* dan *magic bar*

dapat dilihat pada Gambar 4.9. Implementasi fungsi untuk inisiasi *health* dan *magic point* pemain, mengkonversi nilai angka *health* dan *magic point* pemain menjadi obyek *slider* dan mengurangi *health point* pemain jika terkena serangan *monster* dapat dilihat dari Kode Sumber 4.12-4.14.



Gambar 4.9 Tampilan Obyek Health Dan Magic Bar

```

1. public class PlayerManager : MonoBehaviour
2. {
3.     void Update()
4.     {
5.         if (ishit)
6.         {
7.             currenttime -= 1 * Time.deltaTime;
8.         }
9.         if(currenttime <= 0)
10.        {
11.            currenttime = 0;
12.            ishit = false;
13.            bloodimg.SetActive(false);
14.        }
15.        hb.SetHealth(health);
16.        mb.SetMagic(magic);
17.
18.        if (health <= 0)
19.        {
20.            SceneManager.LoadScene("LoseScene");
21.        }
22.    }
23.    public void DecreaseHealth(int amount)
24.    {

```

```

25.         health -= amount;
26.         currenttime = startingtime;
27.         hb.SetHealth(health);
28.         ishit = true;
29.         bloodimg.SetActive(true);
30.         Debug.Log(health);
31.     }
32. }
```

Kode Sumber 4.12 Kode Sumber Health Dan Magic Point Manager

```

1. public class HealthBar : MonoBehaviour
2. {
3.     public Slider slider;
4.     public void SetMaxHealth(int health)
5.     {
6.         slider.maxValue = health;
7.         slider.value = health;
8.     }
9.     public void SetHealth(int health)
10.    {
11.        slider.value = health;
12.    }
13. }
```

Kode Sumber 4.13 Kode Sumber Health Bar Slider

```

1. public class MPBar : MonoBehaviour
2. {
3.     public Slider slider;
4.     public void SetMaxMagic(int health)
5.     {
6.         slider.maxValue = health;
7.         slider.value = health;
8.     }
9.     public void SetMagic(int health)
10.    {
11.        slider.value = health;
12.    }
13. }
```

*Kode Sumber 4.14 Kode Sumber Magic Bar Slider*4. *Health dan Magic Potion*

Health dan *Magic Potion* merupakan obyek yang menambah *health* dan *magic point* pemain jika terkena panah. Obyek *Health* dan *Magic Potion* dapat dilihat pada Gambar 3.12. Kode Sumber *Health* dan *Magic Potion* memiliki fungsi untuk menambah *health* dan *magic point* pemain jika terkena panah, menampilkan teks untuk memberikan pemain informasi jika telah dikenai, dan melakukan *Destroy* pada obyek jika telah dikenai. Implementasi fungsi tersebut dapat dilihat pada Kode Sumber 4.15.

```
1. public class Potions : MonoBehaviour, IDamageable
2. {
3.     void Update()
4.     {
5.         if (isdead)
6.         {
7.             if(this.gameObject.name == "Bottle_Health"
&& isdone == false)
8.             {
9.                 MyPlayer.health += 20;
10.                Debug.Log(MyPlayer.health);
11.                StartCoroutine(EnemyDead());
12.                isdone = true;
13.                isdead = false;
14.            }
15.            if(this.gameObject.name == "Bottle_Magic" &
& isdone == false)
16.            {
17.                MyPlayer.magic += 20;
18.                Debug.Log(MyPlayer.magic);
19.                StartCoroutine(EnemyDead());
20.                isdone = true;
21.                isdead = false;
22.            }
23.        }
24.    }
```

```

25.     private void ShowText()
26.     {
27.         GameObject textku = Instantiate(Floating, g
ameObject.transform.position, Quaternion.identity)
as GameObject;
28.         textku.transform.SetParent(GameObject.FindG
ameObjectWithTag("Canvas").transform, false);
29.         textku.GetComponent<TMP_Text>().text = dama
geamount.ToString();
30.     }
31.     public void Damage(int amount, float
dmgmultiplier)
32.     {
33.         health -= amount;
34.         damageamount = amount;
35.         ShowText();
36.     }
37.     IEnumerator EnemyDead()
38.     {
39.
40.         Destroy(this.gameObject);
41.     }
42. }
```

Kode Sumber 4.15 Kode Sumber Health Dan Magic Potion

4.2.3. Implementasi Akhir Permainan

Tampilan akhir permainan memiliki dua opsi, yaitu kalah dan menang. Jika pemain berhasil menyelesaikan level terakhir maka pemain akan memasuki tampilan yang berisi *credit scene*. Jika *health point* pemain mencapai 0 maka pemain akan kalah dan memasuki *lose scene*. Pada kedua *scene* tersebut pemain dapat memulai permainan kembali atau keluar dari permainan. Implementasi kedua *scene* tersebut dapat dilihat pada Gambar 4.10 dan 4.11. Implementasi fungsi yang mengendalikan kondisi kalah dan menang dapat dilihat pada Kode Sumber 4.12 dan 4.11.

Implementasi fungsi untuk memulai kembali permainan dan keluar dari permainan dapat dilihat pada Kode Sumber 4.16 dan 4.17.



Gambar 4.10 Tampilan Credit Scene



Gambar 4.11 Tampilan Lose Scene

```
1. public class PlayAgain : MonoBehaviour, IDamageable  
2. {  
3.     public int health;
```

```
4.     public void Damage(int amount, float
5.         dmgmultiplier)
6.     {
7.         health -= amount;
8.         if (health <= 0)
9.         {
10.             SceneManager.LoadScene("MainMenu");
11.         }
12.     }
```

Kode Sumber 4.16 Kode Sumber Memulai Kembali Permainan

```
1. public class ExitGame : MonoBehaviour, IDamageable
2. {
3.     public int health;
4.     public void Damage(int amount, float
dmgmultiplier)
5.     {
6.         health -= amount;
7.         if (health <= 0)
8.         {
9.             Application.Quit();
10.        }
11.    }
12. }
```

Kode Sumber 4.17 Kode Sumber Keluar Dari Permainan

[Halaman ini sengaja dikosongkan]

BAB V

PENGUJIAN DAN EVALUASI

Bab ini membahas mengenai rangkaian uji coba dan evaluasi yang dilakukan terhadap permainan *Archer Master*.

5.1. Lingkungan Pengujian

Lingkungan pengujian sistem pada penggeraan tugas ini dilakukan pada lingkungan dan alat kakas pada Tabel 5.1 berikut:

Tabel 5.1 Tabel Lingkungan Pengujian Sistem

Perangkat	Spesifikasi
Perangkat Keras	<ul style="list-style-type: none">• Prosesor: Intel® Core™ i7-7700K CPU @ 4.20GHz (8 CPUs), ~4.2GHz• Memori: 16384MB• VGA: NVIDIA GeForce GTX 1060 6Gb• Oculus Rift S• Oculus Touch
Perangkat Lunak	<ul style="list-style-type: none">• Sistem Operasi Microsoft Windows 10 64-bit• Perangkat Pengembang Unity3D 2019.2.11f1• Perangkat Pembantu Visual Studio Community 2019, Microsoft Word 2016, Adobe Premiere, Adobe Photoshop dan Blender.

5.2. Pengujian Fungsionalitas

Pengujian fungsionalitas sistem dilakukan dengan menyiapkan sejumlah skenario sebagai tolak ukur keberhasilan pengujian. Pengujian fungsionalitas yang terdapat pada permainan dijabarkan sebagai berikut.

5.2.1. Uji Coba Awal Permainan

Pada sub bab ini dijelaskan secara detil mengenai skenario yang dilakukan dan hasil yang didapatkan dari pengujian fungsionalitas perangkat lunak yang dibangun pada halaman awal permainan. Penjelasan disajikan dengan menampilkan kondisi awal, masukan, keluaran, hasil yang dicapai, dan kondisi akhir.

Pada tampilan awal permainan yang akan diuji adalah fungsionalitas Objek sasaran untuk memulai permainan, busur, panah, dan *Active Skill*. Tampilan awal permainan dapat dilihat pada Gambar 4.1-4.2. Skenario yang telah diuji terdapat pada Tabel 5.2.

Tabel 5.2 Hasil Uji Coba Awal Permainan

ID	UF-001
Nama	Uji Coba Pada Awal Permainan
Tujuan uji coba	Pengguna memahami cara menembak panah dan menggunakan <i>Active Skill</i> . Pengguna dapat memulai permainan dengan menembak objek sasaran.
Kondisi awal	Pemain berada pada halaman awal permainan
Skenario 1	Pemain menggerakkan busur dan panah
Masukan	Menggerakkan <i>Oculus Touch</i> kiri dan kanan
Keluaran yang diharapkan	Busur dan panah bergerak sesuai gerakan <i>Oculus Touch</i> yang dilakukan pemain
Hasil uji coba	Berhasil
Kondisi Akhir	Busur dan panah bergerak sesuai gerakan <i>Oculus Touch</i> yang dilakukan pemain
Skenario 2	Pemain melepaskan panah
Masukan	Menekan tombol yang ada pada jari telunjuk di <i>Oculus Touch</i> kanan
Keluaran yang diharapkan	Pemain melepaskan panah dari busur
Hasil uji coba	Berhasil

Kondisi Akhir	Pemain melepaskan panah dari busur
Skenario 3	Pemain menggunakan Active Skill
Masukan	Mengucapkan kata yang mewakili sebuah Active Skill
Keluaran yang diharapkan	Active skill telah diaktifkan dan dapat digunakan oleh pemain
Hasil uji coba	Berhasil
Kondisi Akhir	Active skill telah diaktifkan dan dapat digunakan oleh pemain
Skenario 4	Pemain dapat melihat Informasi Active Skill
Masukan	Pemain mengaktifkan Active Skill
Keluaran yang diharapkan	Informasi Active Skill berubah dan dapat dilihat oleh pemain
Hasil uji coba	Berhasil
Kondisi Akhir	Informasi Active Skill berubah dan dapat dilihat oleh pemain
Skenario 5	Pemain dapat memulai permainan
Masukan	Pemain menembak objek sasaran menggunakan panah atau Active Skill
Keluaran yang diharapkan	Pemain dipindahkan ke halaman permainan
Hasil uji coba	Berhasil
Kondisi Akhir	Pemain dipindahkan ke halaman permainan

5.2.2. Uji Coba Permainan

Pada sub bab ini dijelaskan mengenai skenario yang dilakukan dan hasil yang didapatkan dari pengujian fungsionalitas pada permainan. Penjelasan disajikan dengan menampilkan

kondisi awal, masukan, keluaran, hasil yang dicapai, dan kondisi akhir. Skenario yang telah diuji terdapat pada Tabel 5.3.

Tabel 5.3 Hasil Uji Coba Permainan

ID	UF-002
Nama	Uji Coba Pada Permainan
Tujuan uji coba	Pengguna mengetahui fungsionalitas interaksi yang ada pada permainan
Kondisi awal	Pemain berada pada halaman permainan
<i>Skenario 1</i>	<i>Panah yang dilepaskan pemain dapat menyerang monster</i>
Masukan	Menembak panah pada <i>monster</i>
Keluaran yang diharapkan	<i>Health point monster</i> berkurang
Hasil uji coba	Berhasil
Kondisi Akhir	<i>Health point monster</i> berkurang
<i>Skenario 2</i>	<i>Pemain dapat menyerang monster menggunakan Active Skill</i>
Masukan	Menembak <i>Active Skill</i> pada <i>monster</i>
Keluaran yang diharapkan	<i>Health point monster</i> berkurang
Hasil uji coba	Berhasil
Kondisi Akhir	<i>Health point monster</i> berkurang
<i>Skenario 3</i>	<i>Pemain dapat mengalahkan monster</i>
Masukan	Pemain menyerang monster hingga <i>health point monster</i> mencapai 0
Keluaran yang diharapkan	Musuh dikalahkan dan informasi jumlah musuh berkurang
Hasil uji coba	Berhasil

Kondisi Akhir	Musuh dikalahkan dan informasi jumlah musuh berkurang
Skenario 4	Pemain dapat menambah health point dan magic point
Masukan	Menembak <i>health potion</i> dan <i>magic potion</i>
Keluaran yang diharapkan	<i>Health point</i> dan <i>magic point</i> pemain bertambah
Hasil uji coba	Berhasil
Kondisi Akhir	<i>Health point</i> dan <i>magic point</i> pemain bertambah
Skenario 5	Pemain dapat melihat jumlah monster yang masih hidup
Masukan	Mengalahkan <i>monster</i>
Keluaran yang diharapkan	Informasi jumlah <i>monster</i> sesuai dengan jumlah <i>monster</i> yang masih hidup
Hasil uji coba	Berhasil
Kondisi Akhir	Informasi jumlah <i>monster</i> sesuai dengan jumlah <i>monster</i> yang masih hidup
Skenario 6	Pemain memiliki health point dan magic point yang dapat berkurang
Masukan	Diserang oleh monster atau menggunakan <i>Active Skill</i>
Keluaran yang diharapkan	<i>Health point</i> pemain berkurang setelah diserang oleh monster dan <i>magic point</i> pemain berkurang setelah menggunakan <i>Active Skill</i>
Hasil uji coba	Berhasil
Kondisi Akhir	<i>Health point</i> pemain berkurang setelah diserang oleh monster dan <i>magic point</i> pemain berkurang setelah menggunakan <i>Active Skill</i>
Skenario 7	Pemain kalah saat health point mencapai 0
Masukan	<i>Health point</i> pemain mencapai 0

Keluaran yang diharapkan	Pemain dipindahkan ke halaman <i>Lose Scene</i>
Hasil uji coba	Berhasil
Kondisi Akhir	Pemain dipindahkan ke halaman <i>Lose Scene</i>
Skenario 8	<i>Monster dapat menyerang pemain</i>
Masukan	Pemain berada dalam jarak serang <i>monster</i>
Keluaran yang diharapkan	<i>Monster</i> melakukan serangan kepada pemain
Hasil uji coba	Berhasil
Kondisi Akhir	<i>Monster</i> melakukan serangan kepada pemain
Skenario 9	<i>Monster dapat berjalan mendekati pemain</i>
Masukan	Level Permainan dimulai
Keluaran yang diharapkan	<i>Monster</i> berjalan mendekati pemain
Hasil uji coba	Berhasil
Kondisi Akhir	<i>Monster</i> berjalan mendekati pemain
Skenario 10	<i>Pemain dapat memenangkan permainan</i>
Masukan	Mengalahkan semua <i>monster</i> pada level 5
Keluaran yang diharapkan	Pemain dipindahkan ke halaman <i>Credit Scene</i>
Hasil uji coba	Berhasil
Kondisi Akhir	Pemain dipindahkan ke halaman <i>Credit Scene</i>

5.2.3. Uji Coba Akhir Permainan

Pada sub bab ini dijelaskan secara detil mengenai skenario yang dilakukan dan hasil yang didapatkan dari pengujian fungsionalitas perangkat lunak yang dibangun pada halaman akhir permainan. Penjelasan disajikan dengan menampilkan kondisi awal, masukan, keluaran, hasil yang dicapai, dan kondisi akhir.

Pada menu permainan yang akan diuji adalah fungsionalitas objek yang terdapat di akhir permainan, yaitu objek yang dapat ditembak untuk memulai kembali permainan dan keluar dari permainan. Tampilan akhir permainan dapat dilihat pada Gambar 4.10 dan Gambar 4.11. Skenario yang telah diuji terdapat pada Tabel 5.4.

Tabel 5.4 Hasil Uji Coba Akhir Permainan

ID	UF-003
Nama	Uji Coba Pada Akhir Permainan
Tujuan uji coba	Pengguna mengetahui fungsionalitas objek yang ada pada akhir permainan
Kondisi awal	Pemain berada pada halaman akhir permainan
Skenario 1	<i>Pemain memulai kembali permainan</i>
Masukan	Menembak objek untuk memulai kembali permainan
Keluaran yang diharapkan	Pemain berpindah ke halaman awal permainan
Hasil uji coba	Berhasil
Kondisi Akhir	Pemain berada pada halaman permainan
Skenario 2	<i>Pemain keluar dari permainan</i>
Masukan	Menembak objek untuk keluar dari permainan
Keluaran yang diharapkan	Pemain keluar dari permainan
Hasil uji coba	Berhasil

Kondisi Akhir	Pemain keluar dari permainan
---------------	------------------------------

5.2.4. Hasil Uji Coba

Pada sub bab ini diberikan hasil evaluasi dari pengujian yang dilakukan pada permainan. Hasil evaluasi dapat dilihat pada Tabel 5.5

Tabel 5.5 Hasil Evaluasi

ID	Deskripsi	Kemungkinan / Skenario	Perilaku Terlaksana
UF-001	Uji Coba Menu Permainan	Skenario 1	Ya
		Skenario 2	Ya
		Skenario 3	Ya
		Skenario 4	Ya
		Skenario 5	Ya
UF-002	Uji Coba Permainan	Skenario 1	Ya
		Skenario 2	Ya
		Skenario 3	Ya
		Skenario 4	Ya
		Skenario 5	Ya
		Skenario 6	Ya
		Skenario 7	Ya
		Skenario 8	Ya
		Skenario 9	Ya
		Skenario 10	Ya
UF-003	Uji Coba Akhir Permainan	Skenario 1	Ya
		Skenario 2	Ya

5.3. Pengujian Pengguna

Pengujian pada permainan yang dibangun tidak hanya dilakukan pada fungsionalitas yang dimiliki, tetapi juga ditujukan kepada pengguna untuk mencoba secara langsung. Pengujian ini

berfungsi sebagai pengujian subjektif yang bertujuan untuk mengetahui tingkat keberhasilan permainan yang dibangun dari sisi pengguna. Hal ini dapat dicapai dengan meminta penilaian dan tanggapan dari pengguna terhadap sejumlah aspek permainan yang ada.

5.3.1. Skenario Pengujian Pengguna

Dalam melakukan pengujian permainan, pengguna diminta mencoba memainkan permainan untuk mencoba semua fungsionalitas dan fitur yang ada. Pengujian permainan oleh pengguna dilakukan dengan sebelumnya memberikan informasi seputar permainan, kegunaan, dan fitur-fitur yang dimiliki. Setelah informasi tersampaikan, pengguna kemudian diarahkan untuk langsung mencoba permainan dengan spesifikasi lingkungan yang sama dengan yang telah diuraikan pada Tabel 5.1 Lingkungan Uji Coba.

Jumlah pengguna yang terlibat dalam pengujian perangkat sebanyak 10 orang. Dalam melakukan pengujian pengguna melakukan percobaan lebih dari satu kali penggunaan untuk masing-masing pengguna.

Dalam memberikan penilaian dan tanggapan, pengguna diberikan kuesioner pengujian permainan. . Kuesioner pengujian ini memiliki beberapa aspek penilaian seputar desain antarmuka, *immersivity*, dan tingkat kenyamanan permainan. Nilai yang diberikan rentang nilai 1 hingga 5 dengan rincian pada Tabel 5.6 . pada bagian akhir terdapat saran untuk perbaikan fitur. Detil kuesioner pengguna dapat dilihat pada Tabel 5.7.

Tabel 5.6 Rentang Nilai

No	Keterangan	Nilai
1	Sangat Tidak Setuju (STS)	1
2	Tidak Setuju (TS)	2
3	Netral (N)	3
4	Setuju (S)	4
5	Sangat Setuju (SS)	5

Tabel 5.7 Format Kuesioner

No	Parameter	STS	TS	N	S	SS
1	Permainan memiliki objek dan latar belakang yang sesuai					
2	Permainan memiliki tampilan, warna, dan desain yang menarik					
3	Permainan memiliki antarmuka yang mudah dilihat/dikenali					
4	Saya merasa seperti pemanah					
5	Permainan dapat menangkap suara saya dengan baik					
6	Saya merasakan sensasi mengucapkan <i>Active Skill</i>					
7	Permainan dapat berjalan lancar tanpa adanya lag dan atau crash					
8	Saya merasa terbantu dengan adanya petunjuk yang ada					
9	Saya merasa nyaman selama memainkan permainan ini					
10	Saya merasa tertarik untuk memainkan permainan ini untuk selanjutnya					

5.3.2. Daftar Penguji Permainan

Pada sub bab ini ditunjukkan daftar pengguna yang bertindak sebagai penguji coba permainan yang dibangun. Dalam pengujian ini tidak terdapat kriteria atau keahlian khusus yang harus dimiliki pengguna karena permainan ini ditunjukkan kepada berbagai kalangan pengguna baik yang suka bermain permainan ataupun tidak. Daftar nama penguji permainan ini dapat dilihat pada Tabel 5.8.

Tabel 5.8 Daftar Penguji Permainan

No	Nama	Pekerjaan	Usia
1	Taritno	Supir	39
2	M Ronald	Swasta	48
3	Mulyanto Hartono	Karyawan	52
4	Guruh Caesar G	Pelajar	7
5	Sutrisno	Keamanan	40
6	Siwi Peni	Karyawan	50
7	Muhammad Irfan S.R.W.	Mahasiswa	21
8	Kurniawan Adjie S.	Mahasiswa	23
9	Siwi Handayani	Wirausaha	57
10	Fariz Ardin A.	Mahasiswa	22

5.3.3. Hasil Pengujian Pengguna

Sistem penilaian didasarkan pada skala perhitungan satu sampai lima dimana skala satu menunjukkan nilai terendah dan skala lima menunjukkan skala tertinggi. Penilaian akhir kemudian dilakukan dengan menghitung berapa banyak penguji yang memilih suatu skala tertentu dan kemudian dicari nilai rata-ratanya. Hasil uji coba dipaparkan secara lengkap dengan disertai Tabel yang dapat dilihat pada Tabel 5.9 dan Tabel 5.10.

Tabel 5.9 Hasil Pengujian Pengguna

No	Pernyataan	Penilaian					Rata-Rata
		1	2	3	4	5	
1	Permainan memiliki objek dan latar belakang yang sesuai	0	0	2	5	3	4.1
2	Permainan memiliki tampilan, warna, dan desain yang menarik	0	0	3	6	1	3.8
3	Permainan memiliki antarmuka yang mudah dilihat/dikenali	0	0	2	7	1	3.9
4	Saya merasa seperti pemanah	0	0	0	3	7	4.7
5	Permainan dapat menangkap suara saya dengan baik	0	0	4	6	0	3.6
6	Saya merasakan sensasi mengucapkan <i>Active Skill</i>	0	0	1	8	1	4.0
7	Permainan dapat berjalan lancar tanpa adanya lag dan atau crash	0	0	1	5	4	4.3
8	Saya merasa terbantu dengan adanya petunjuk yang ada	0	0	3	6	1	3.8
9	Saya merasa nyaman selama memainkan permainan ini	0	0	2	5	3	4.1
10	Saya merasa tertarik untuk memainkan permainan ini untuk selanjutnya	0	0	0	5	5	4.5

Tabel 5.10 Hasil Akhir Pengujian Pengguna

No	Pernyataan	Rata-Rata	Total	Total (%)
Parameter Antarmuka				
1	Permainan memiliki objek dan latar belakang yang sesuai	4.1	3.93	78.6%
2	Permainan memiliki tampilan, warna, dan desain yang menarik	3.8		
3	Permainan memiliki antarmuka yang mudah dilihat/dikenali	3.9		
Parameter Immersivity				
4	Saya merasa seperti pemanah	4.7	4.10	82%
5	Permainan dapat menangkap suara saya dengan baik	3.6		
6	Saya merasakan sensasi mengucapkan <i>Active Skill</i>	4.0		
Parameter Kenyamanan				
7	Permainan dapat berjalan lancar tanpa adanya lag dan atau crash	4.3	4.18	83.5%
8	Saya merasa terbantu dengan adanya petunjuk yang ada	3.8		
9	Saya merasa nyaman selama memainkan permainan ini	4.1		
10	Saya merasa tertarik untuk memainkan permainan ini untuk selanjutnya	4.5		

5.3.4. Kritik dan Saran Pengguna

Dalam memberikan penilaian dan tanggapan, pengguna diberikan kuesioner pengujian permainan. Kuesioner pengujian permainan ini terdapat bagian kritik dan saran untuk perbaikan fitur

kedepannya. Kritik dan sara pengguna dapat dilihat pada Tabel 5.11.

Tabel 5.11 Kritik dan Saran Pengguna

No	Nama	Kritik dan Saran
1	Taritno	Level 3 terlalu gelap
2	M Ronald	Perlu nyawa tambahan lebih
3	Mulyanto Hartono	Permainan mudah dimengerti
4	Guruh Caesar G	Game nya bagus seperti main panah beneran
5	Sutrisno	Sangat baik untuk kesehatan
6	Siwi Peni	<i>Monster</i> perlu diberi suara agar lebih menegangkan
7	Muhammad Irfan S.R.W.	<i>UI Health Point</i> terlalu besar menghalangi
8	Kurniawan Adjji S.	Game menyenangkan
9	Siwi Handayani	Sudah bagus dan mudah dipahami bisa untuk olahraga
10	Fariz Ardin A.	Kalau bisa setiap musuh dikasih <i>HP Bar</i>

5.4. Evaluasi Pengujian

Sub bab ini membahas mengenai evaluasi terhadap pengujian-pengujian yang telah dilakukan. Dalam hal ini, evaluasi menunjukkan data rekapitulasi dari hasil pengujian fungsionalitas. Rekapitulasi disusun dalam bentuk Tabel yang dapat dilihat pada Tabel 5.5. Dari data yang terdapat pada Tabel tersebut, diketahui bahwa aplikasi yang dibuat telah berjalan sesuai dengan skenario yang diharapkan.

BAB VI

KESIMPULAN DAN SARAN

Bab ini membahas tentang kesimpulan yang diperoleh selama penggerjaan tugas akhir dan saran mengenai pengembangan yang dapat dilakukan terhadap tugas akhir ini di masa yang akan datang.

6.1. Kesimpulan

Dari hasil pengamatan selama proses perancangan, implementasi, dan pengujian yang dilakukan, kesimpulan dapat diambil sebagaimana berikut.

1. Permainan *Archer Master* berhasil dibuat serta memiliki tingkat kesulitan dan skenario yang sesuai dengan rancangan permainan.
2. Permainan berhasil menampilkan lingkungan realitas virtual menggunakan *Oculus Rift* serta menerapkan aturan main yang sesuai dengan rancangan permainan.
3. Permainan dapat mendengar suara pemain dan mengaktifkan *Active Skill* berdasarkan kata yang dikenali oleh *speech recognition*.
4. Berdasarkan hasil dari pengujian fungsionalitas dan pengujian oleh pengguna dapat disimpulkan permainan *Archer Master* telah mengimplementasikan rancangan dengan baik.

6.2. Saran

Berikut merupakan beberapa saran untuk pengembangan sistem di masa yang akan datang. Saran-saran ini didasarkan pada hasil perancangan, implementasi, dan pengujian yang telah dilakukan. Di antaranya adalah sebagai berikut:

1. Menambahkan lebih banyak *Active Skill* dan kata yang dapat dikenali.

2. Memberi suara pada *monster* agar permainan lebih terasa nyata
3. *User Interface* disesuaikan dengan lingkungan realitas virtual agar tidak menghalangi pandangan pemain.

DAFTAR PUSTAKA

- [1] Wikipedia, “Shooter game - Wikipedia,” [Online]. Available: https://en.wikipedia.org/wiki/Shooter_game. [Diakses 4 Desember 2019].
- [2] Wikipedia, “Speech Recognition - Wikipedia,” [Online]. Available: https://en.wikipedia.org/wiki/Speech_recognition. [Diakses 12 Desember 2019].
- [3] YellRobot.com, “Speech recognition in video games - Yell Robot,” [Online]. Available: <https://yellrobot.com/speech-recognition-in-video-games/>. [Diakses 11 Desember 2019].
- [4] Wikipedia, “C Sharp - Wikipedia Bahasa Indonesia, ensiklopedia bebas,” [Online]. Available: https://id.wikipedia.org/wiki/C_sharp. [Diakses 1 Desember 2019].
- [5] Wikipedia, “Unity (game engine) - Wikipedia,” [Online]. Available: [https://en.wikipedia.org/wiki/Unity_\(game_engine\)](https://en.wikipedia.org/wiki/Unity_(game_engine)). [Diakses 1 Desember 2019].
- [6] Wikipedia, “Virtual reality - Wikipedia,” [Online]. Available: https://en.wikipedia.org/wiki/Virtual_reality. [Diakses 1 Desember 2019].
- [7] T. Maya, “PEMODELAN MODEL OBJEK 3D,” [Online]. Available: <http://www.teorimaya.com/2018/09/pemodelan-model-objek-3d.html>. [Diakses 1 Desember 2019].
- [8] D. Gonzalez, “Speech Recognition and VR - Unity Technologies Blog,” [Online]. Available: <https://blogs.unity3d.com/2016/08/02/speech-recognition-and-vr/>. [Diakses 12 Desember 2019].
- [9] E. Barber, “Get Started in Archery Through Virtual Reality,” [Online]. Available:

- <https://www.archery360.com/2019/02/14/get-started-in-archery-through-virtual-reality/>. [Diakses 19 Juni 2020].
- [10] Wikipedia, “Immersion (virtual reality) - Wikipedia,” [Online]. Available: [https://en.wikipedia.org/wiki/Immersion_\(virtual_reality\)](https://en.wikipedia.org/wiki/Immersion_(virtual_reality)). [Diakses 7 Juli 2020].

LAMPIRAN



Lampiran 1 Penguji Coba (1)



Lampiran 2 Penguji Coba (2)



Lampiran 3 Penguji Coba (3)



Lampiran 4 Penguji Coba (4)



Lampiran 5 Penguji Coba (5)



Lampiran 6 Penguji Coba (6)



Lampiran 7 Penguji Coba (7)



Lampiran 8 Penguji Coba (8)



Lampiran 9 Penguji Coba (9)



Lampiran 10 Penguji Coba (10)

KUSKUSER TUGAS ANAK
002249000000 - Ani Sutrisna

Persentase Kepelitas Virtual Andher Muster Menggunakan Teknologi Digital
Rif dan Speech Recognition

Identitas Respondee Surabaya, 1 Juli 2020

Nama Lengkap : JATINNO
Pekerjaan : Sopir
Usia : 33 thn.

B. KAMANTIKSI RESPONDEE
Jawaban pertanyaan di bawah ini dengan menggunakan skala centring 5.
 1. Persepsi anda memerlukan teknologi digital atau gerak berbasis teknologi informasi?
 ✓ Personil Publik
 2. Persepsi anda memerlukan teknologi digital atau gerak berbasis teknologi informasi?
 ✓ Personil Publik
 3. Persepsi anda memerlukan teknologi digital atau gerak berbasis teknologi informasi?
 ✓ Personil Publik
 4. Apakah anda pernah mendengar tentang?
 ✓ Personil Publik

C. PERILAKU TEPATGAB PERNIAGAAN
Jawaban ditukarkan kepada dalam menggunakan skala centring 5.
 11 = Sangat Sama 2 = Sedang 3 = Rata-rata 4 = Berbeza
 12 = Sangat Berbeza 13 = Sangat Berbeza

No.	Parameter Antara mode	0%	1%	2%	3%	4%
1.	Persepsi anda memerlukan teknologi digital atau gerak berbasis teknologi informasi?					

D. KETAHUI DAN SARAN
Lihat di selanjutnya

KUSKUSER TUGAS ANAK
002249000000 - Ani Sutrisna

Persentase Kepelitas Virtual Andher Muster Menggunakan Teknologi Digital
Rif dan Speech Recognition

Identitas Respondee Surabaya, 1 Juli 2020

Nama Lengkap : M. RONALD
Pekerjaan : driver
Usia : 31 thn.

B. KAMANTIKSI RESPONDEE
Jawaban pertanyaan di bawah ini dengan menggunakan skala centring 5.
 1. Persepsi anda memerlukan teknologi digital atau gerak berbasis teknologi informasi?
 ✓ Personil Publik
 2. Persepsi anda memerlukan teknologi digital atau gerak berbasis teknologi informasi?
 ✓ Personil Publik
 3. Persepsi anda memerlukan teknologi digital atau gerak berbasis teknologi informasi?
 ✓ Personil Publik
 4. Apakah anda pernah mendengar tentang?
 ✓ Personil Publik

C. PERILAKU TEPATGAB PERNIAGAAN
Jawaban ditukarkan kepada dalam menggunakan skala centring 5.
 11 = Sangat Sama 2 = Sedang 3 = Rata-rata 4 = Berbeza
 12 = Sangat Berbeza 13 = Sangat Berbeza

No.	Parameter Antara mode	0%	1%	2%	3%	4%
1.	Persepsi anda memerlukan teknologi digital atau gerak berbasis teknologi informasi?					

D. KETAHUI DAN SARAN
- Dik M. RONALD, terima kasih

 KONSENTRASI TUGAS AKHIR ITS (Institut Teknologi Sepuluh Nopember)						
Penilaian Realitas Virtual Archer Master Menggunakan Teknologi Optical Rift dan Speech Recognition						
Waktu Responsum:		Senin, 16 Juli 2018				
Nama Lengkap:		<u>Angga Pratama</u>				
Pekerjaan:		<u>Archer Master</u>				
Telepon:		<u>0812 94 24 24</u>				
Email:		<u>angga.pratama@its.ac.id</u>				
G. KANSTANTINUS RESPONSMEN Assistent pertama di bawah ini dengan menggunakan teknologi AI						
<ol style="list-style-type: none"> 1. Penilaian anda memahami teknologi pengenalan suara dengan baik? Oke/Baik <input checked="" type="checkbox"/> 2. Penilaian anda memahami teknologi pengenalan wajah dengan baik? Oke/Baik <input checked="" type="checkbox"/> 3. Penilaian anda memahami teknologi pengenalan gerak dengan baik? Oke/Baik <input checked="" type="checkbox"/> 4. Apakah anda pernah mendengar tentang teknologi AI? Oke/Baik <input checked="" type="checkbox"/> 						
B. PENILAIAN TERHADAP PERMAJUAN Kode responsum untuk menilai perkembangan teknologi AI SI = Sangat Baik 2 = Baik A = Rata-Rata 3 = Sangat Buruk Buruk						
No	Persentase Kemajuan	0%	1%	5%	10%	25%
1	Penilaian memiliki teknologi AI dengan teknologi yang masih belum diketahui	<input checked="" type="checkbox"/>				
C. KETUA DAN PEMERIKSA Setelah melihat hasil jadwal						
 KONSENTRASI TUGAS AKHIR ITS (Institut Teknologi Sepuluh Nopember)						
Penilaian Realitas Virtual Archer Master Menggunakan Teknologi Optical Rift dan Speech Recognition						
Waktu Responsum:		Senin, 16 Juli 2018				
Nama Lengkap:		<u>Budi Cahya D.</u>				
Pekerjaan:		<u>Archer</u>				
Telepon:		<u>0812 94 24 24</u>				
Email:		<u>budi.cahya@its.ac.id</u>				
G. KANSTANTINUS RESPONSMEN Assistent pertama di bawah ini dengan menggunakan teknologi AI						
<ol style="list-style-type: none"> 1. Penilaian anda memahami teknologi pengenalan suara dengan baik? Oke/Baik <input checked="" type="checkbox"/> 2. Penilaian anda memahami teknologi pengenalan wajah dengan baik? Oke/Baik <input checked="" type="checkbox"/> 3. Penilaian anda memahami teknologi pengenalan gerak dengan baik? Oke/Baik <input checked="" type="checkbox"/> 4. Apakah anda pernah mendengar tentang teknologi AI? Oke/Baik <input checked="" type="checkbox"/> 						
B. PENILAIAN TERHADAP PERMAJUAN Kode responsum untuk menilai perkembangan teknologi AI SI = Sangat Baik 2 = Baik A = Rata-Rata 3 = Sangat Buruk Buruk						
No	Persentase Kemajuan	0%	1%	5%	10%	25%
1	Penilaian memiliki teknologi AI dengan teknologi yang masih belum diketahui	<input checked="" type="checkbox"/>				
C. KETUA DAN PEMERIKSA Setelah melihat hasil jadwal						
 KONSENTRASI TUGAS AKHIR ITS (Institut Teknologi Sepuluh Nopember)						
Penilaian Realitas Virtual Archer Master Menggunakan Teknologi Optical Rift dan Speech Recognition						
Waktu Responsum:		Senin, 16 Juli 2018				
Nama Lengkap:		<u>Budi Cahya D.</u>				
Pekerjaan:		<u>Archer</u>				
Telepon:		<u>0812 94 24 24</u>				
Email:		<u>budi.cahya@its.ac.id</u>				
G. KANSTANTINUS RESPONSMEN Assistent pertama di bawah ini dengan menggunakan teknologi AI						
<ol style="list-style-type: none"> 1. Penilaian anda memahami teknologi pengenalan suara dengan baik? Oke/Baik <input checked="" type="checkbox"/> 2. Penilaian anda memahami teknologi pengenalan wajah dengan baik? Oke/Baik <input checked="" type="checkbox"/> 3. Penilaian anda memahami teknologi pengenalan gerak dengan baik? Oke/Baik <input checked="" type="checkbox"/> 4. Apakah anda pernah mendengar tentang teknologi AI? Oke/Baik <input checked="" type="checkbox"/> 						
B. PENILAIAN TERHADAP PERMAJUAN Kode responsum untuk menilai perkembangan teknologi AI SI = Sangat Baik 2 = Baik A = Rata-Rata 3 = Sangat Buruk Buruk						
No	Persentase Kemajuan	0%	1%	5%	10%	25%
1	Penilaian memiliki teknologi AI dengan teknologi yang masih belum diketahui	<input checked="" type="checkbox"/>				
C. KETUA DAN PEMERIKSA Setelah melihat hasil jadwal						

KEMENTERIAN TUGAS RAKER ITS (Institut Teknologi Sepuluh Nopember)															
<p align="center">Perbaikan Realitas Virtual Archiver Master Menggunakan Teknologi Occlusion API dan Speech Recognition</p>															
Identitas Responden Nama Lengkap : <u>SARINDA</u> Kode : <u>1611011402101000</u> Perguruan : <u>STMIK ITB MADIUN</u> Tgl. : <u>40.01.2020</u> Unit : <u>ITB</u>															
A. KARAKTERISTIK RESPONDEN Jumlah pertemuan di bantuan AI dengan menggunakan teknologi Occlusion API 1. Apakah anda memerlukan teknologi perangkat keras generasi terbaru untuk membuat AI? ✓Tidak Perlu ✓Diperlukan ✓Tidak Penting 2. Pertemuan anda memerlukan pertemuan secara online dengan teknologi AI? ✓Tidak Penting ✓Diperlukan 3. Pertemuan anda memerlukan pertemuan secara offline dengan teknologi AI? ✓Tidak Penting ✓Diperlukan 4. Apakah anda pernah mendengar tentang AI? ✓Tidak Penting ✓Diperlukan															
B. PERILAKU TERHADAP PENGETAHUAN Soal nilai ditentukan berdasarkan pengalaman teknologi AI 1 = Sangat Baik 5 = Sedang 9 = Buruk 10 = Sangat Buruk 11 = Sangat Tidak Baik															
<table border="1" style="width: 100%;"> <thead> <tr> <th>No.</th> <th>Pertemuan Awalnya</th> <th>1/5</th> <th>2/5</th> <th>3/5</th> <th>4/5</th> <th>5/5</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Pertemuan membutuhkan teknologi AI dengan teknologi Occlusion API</td> <td></td> <td></td> <td><input checked="" type="checkbox"/></td> <td></td> <td></td> </tr> </tbody> </table>		No.	Pertemuan Awalnya	1/5	2/5	3/5	4/5	5/5	1	Pertemuan membutuhkan teknologi AI dengan teknologi Occlusion API			<input checked="" type="checkbox"/>		
No.	Pertemuan Awalnya	1/5	2/5	3/5	4/5	5/5									
1	Pertemuan membutuhkan teknologi AI dengan teknologi Occlusion API			<input checked="" type="checkbox"/>											
C. KETIKA DAN SARANA Sarana teknologi AI yang dibutuhkan untuk mendukung kegiatan															
KELAS DARING ITS (Institut Teknologi Sepuluh Nopember)															
<p align="center">Perbaikan Realitas Virtual Archiver Master Menggunakan Teknologi Occlusion API dan Speech Recognition</p>															
Identitas Responden Nama Lengkap : <u>SWI PENTI</u> Kode : <u>1611011402101000</u> Perguruan : <u>Politeknik BUMN</u> Tgl. : <u>51.01.2020</u> Unit : <u>ITB</u>															
A. KARAKTERISTIK RESPONDEN Jumlah pertemuan di bantuan AI dengan menggunakan teknologi Occlusion API 1. Apakah anda memerlukan teknologi perangkat keras generasi terbaru untuk membuat AI? ✓Tidak Penting ✓Diperlukan 2. Apakah anda memerlukan pertemuan secara online dengan teknologi AI? ✓Tidak Penting ✓Diperlukan 3. Pertemuan anda memerlukan pertemuan secara offline dengan teknologi AI? ✓Tidak Penting ✓Diperlukan 4. Apakah anda pernah mendengar tentang AI? ✓Tidak Penting ✓Diperlukan															
B. PERILAKU TERHADAP PENGETAHUAN Soal nilai ditentukan berdasarkan pengalaman teknologi AI 1 = Sangat Baik 5 = Sedang 9 = Buruk 10 = Sangat Buruk 11 = Sangat Tidak Baik															
<table border="1" style="width: 100%;"> <thead> <tr> <th>No.</th> <th>Pertemuan Awalnya</th> <th>1/5</th> <th>2/5</th> <th>3/5</th> <th>4/5</th> <th>5/5</th> </tr> </thead> <tbody> <tr> <td>1</td> <td>Pertemuan membutuhkan teknologi AI dengan teknologi Occlusion API</td> <td></td> <td></td> <td><input checked="" type="checkbox"/></td> <td></td> <td></td> </tr> </tbody> </table>		No.	Pertemuan Awalnya	1/5	2/5	3/5	4/5	5/5	1	Pertemuan membutuhkan teknologi AI dengan teknologi Occlusion API			<input checked="" type="checkbox"/>		
No.	Pertemuan Awalnya	1/5	2/5	3/5	4/5	5/5									
1	Pertemuan membutuhkan teknologi AI dengan teknologi Occlusion API			<input checked="" type="checkbox"/>											
C. KETIKA DAN SARANA Sarana teknologi AI yang dibutuhkan untuk mendukung kegiatan															
KELAS DARING ITS (Institut Teknologi Sepuluh Nopember)															

KUESIONER TES ALAIR
ATAK: Kuesioner Tes Alair

Pernyataan Realitas Virtual-Archer Master Menggunakan Teknologi-Orakul
Rif dan Speach Recognition

Mitra Responsum: Mukhammad Idris / Birthdate: 10-10-1996

Name: Mukhammad Idris /
Nik: 1101010101010101 /
Sex: Male /
Signature: [Signature]

A. SAKARANTIK RESPONSUM
Jawaban pernyataan di bawah ini dengan menggunakan teknologi-Orakul

- Pernyataan anda mempunyai teknologi yang mampu berkomunikasi dengan manusia.
 Benar Salah
- Pernyataan anda mempunyai teknologi yang mampu berkomunikasi dengan manusia.
 Benar Salah
- Pernyataan anda mempunyai teknologi yang mampu berkomunikasi dengan manusia.
 Benar Salah
- Pernyataan anda mempunyai teknologi yang mampu berkomunikasi dengan manusia.
 Benar Salah

B. PERNYATAAN TERHADAP PEMERINTAH
Jawaban pernyataan di bawah ini dengan menggunakan teknologi-Orakul

SI = Sangat Setuju T = Sesuai A = Rasa-rasa
D = Sedikit Setuju E = Sangat Tidak Setuju

No	Pernyataan Administrasi	SI	T	A	D	E
1.	Pernyataan memudahkan dan mempermudah yang cukup				<input checked="" type="checkbox"/>	

C. IDENTITAS DAN SUMBER
[Lainnya, Sumber, Sumber, sumber]

KUESIONER TES ALAIR
ATAK: Kuesioner Tes Alair

Pernyataan Realitas Virtual-Archer Master Menggunakan Teknologi-Orakul
Rif dan Speach Recognition

Mitra Responsum: Xaverius Alfi / Birthdate: 10-10-1996

Name: Xaverius Alfi /
Nik: 1101010101010101 /
Sex: Male /
Signature: [Signature]

A. SAKARANTIK RESPONSUM
Jawaban pernyataan di bawah ini dengan menggunakan teknologi-Orakul

- Pernyataan anda mempunyai teknologi yang mampu berkomunikasi dengan manusia.
 Benar Salah
- Pernyataan anda mempunyai teknologi yang mampu berkomunikasi dengan manusia.
 Benar Salah
- Pernyataan anda mempunyai teknologi yang mampu berkomunikasi dengan manusia.
 Benar Salah
- Pernyataan anda mempunyai teknologi yang mampu berkomunikasi dengan manusia.
 Benar Salah

B. PERNYATAAN TERHADAP PEMERINTAH
Jawaban pernyataan di bawah ini dengan menggunakan teknologi-Orakul

SI = Sangat Setuju T = Sesuai A = Rasa-rasa
D = Sedikit Setuju E = Sangat Tidak Setuju

No	Pernyataan Administrasi	SI	T	A	D	E
1.	Pernyataan memudahkan dan mempermudah yang cukup				<input checked="" type="checkbox"/>	

C. IDENTITAS DAN SUMBER
[Lainnya, Sumber, Sumber, sumber]

KONSEPUEK TIKALAKH KONSEPUEK TIKALAKH Penerapan Realitas Virtual Archer Marker Menggunakan Teknologi Optical ARF dan Speech Recognition						
Identitas Responden Nama Lengkap : <u>Siti HANIFAH</u> Tanggal : <u>June 2020</u> Pekerjaan : <u>Widya Mahasiswa</u> Ima : <u>STP-J000</u>						
A. KARAKTERISTIK RESPONDEK Identitas penerapan di bawah menggunakan teknologi ARF 1. Perangkat untuk memerlukan perangkat digital atau perangkat teknologi informasi? ✓ 2. Perangkat untuk memerlukan perangkat teknologi informasi? ✓ 3. Perangkat untuk memerlukan perangkat teknologi informasi? ✓ 4. Apakah anda pernah melakukan penelitian? ✓ 5. Apakah anda pernah melakukan penelitian? ✓						
B. PENILAIAN TERADAP PERMANASI skala dalam bantuan dengan menggunakan teknologi ARF 1 = sangat buruk 2 = buruk 3 = cukup 4 = baik 5 = sangat baik. Bantu. 10 = sangat baik						
No	Pertanda Kemeriahan	5	4	3	2	1
1	Perangkat memerlukan teknologi informasi yang tidak relevan					
C. BANTUAN DAN SARAN Bantu dengan <u>Setiap kali mengalami kesulitan teknologi ARF</u>						
2. Perangkat memerlukan teknologi informasi yang relevan dengan tujuan / tujuan ✓ 3. Perangkat memerlukan teknologi informasi yang relevan dengan tujuan / tujuan ✓ 4. Perangkat memerlukan teknologi informasi yang relevan dengan tujuan / tujuan ✓						
No	Pertanda Kemeriahan	5	4	3	2	1
5	Perangkat memerlukan teknologi informasi yang relevan dengan tujuan / tujuan					
6	Perangkat memerlukan teknologi informasi yang relevan dengan tujuan / tujuan					
7	Perangkat dapat meningkatkan hasil kerja dengan baik					
8	Perangkat memerlukan teknologi informasi yang relevan dengan tujuan / tujuan					
No	Pertanda Kemeriahan	5	4	3	2	1
10	Perangkat dapat memberikan hasil kerja yang akurasi masih					
11	Perangkat memberikan hasil kerja yang akurasi masih					
12	Perangkat memberikan hasil kerja yang akurasi masih					
13	Perangkat memberikan hasil kerja yang akurasi masih					
C. BANTUAN DAN SARAN Bantu dengan <u>Setiap kali mengalami kesulitan teknologi ARF</u>						

BIODATA PENULIS



Ariq Sudibyo, lahir di Jakarta pada tanggal 21 November 1998. Lulus dari SMAN 05 Bekasi pada tahun 2016 dan melanjutkan studinya di Departemen Teknik Informatika, Fakultas Teknologi Elektro dan Informatika Cerdas, Institut Teknologi Sepuluh Nopember di Surabaya.

Selama menjadi mahasiswa, penulis ikut dalam staff NST Schematics 2017 pada tahun pertama, dan dilanjutkan staff ahli NST Schematics 2018. Dalam menyelesaikan pendidikan sarjananya, penulis mengambil bidang minat Interaksi, Grafika, dan Seni (IGS). Penulis dapat dihubungi melalui alamat *e-mail*: ariqsudibyo21@gmail.com.