

PENGEMBANGAN GAME BRIDGE OF BALANCE (BOB) DENGAN MENGGUNAKAN KINECT DAN OCULUS RIFT

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Muhammad Faisal
NIM: 125150200111110



PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2016

PENGESAHAN

PENGEMBANGAN GAME BRIDGE OF BALANCE (BOB) DENGAN MENGGUNAKAN
KINECT DAN OCULUS RIFT

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Muhammad Faisal

NIM: 125150200111110

Skripsi ini telah diuji dan dinyatakan lulus pada
18 Agustus 2016

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Wibisono Sukmo Wardhono, S.T, M.T

NIK: 201008 820404 1 001

Issa Arwani, S.Kom, M.Sc

NIP: 19830922 201212 1 003

Mengetahui
Ketua Jurusan Teknik Informatika

Tri Astoto Kurniawan, S.T, M.T, Ph.D

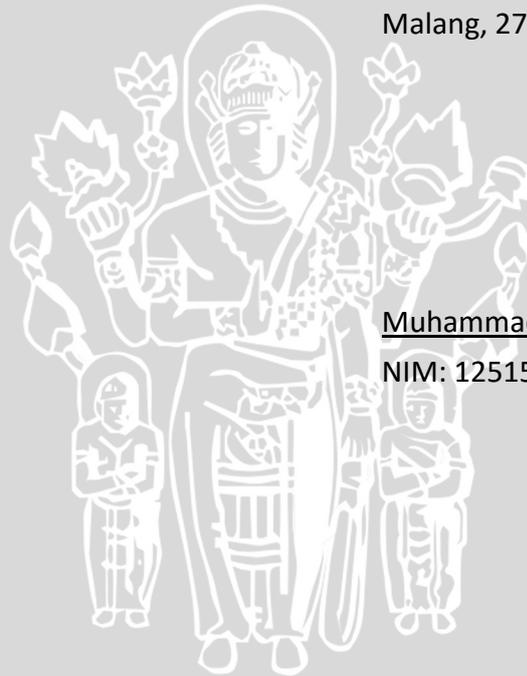
NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 27 Juli 2016



Muhammad Faisal

NIM: 125150200111110

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa karena hanya dengan rahmat dan karunia-Nya, penulis dapat menyelesaikan penyusunan tugas akhir yang berjudul “Pengembangan Game Bridge Of Balance (Bob) dengan Menggunakan Kinect Dan Oculus Rift”.

Penulisan skripsi ini bertujuan untuk memenuhi sebagian syarat memperoleh gelar sarjana pendidikan bagi mahasiswa program S1 pada Fakultas Ilmu Komputer Jurusan Teknik Informatika Universitas Brawijaya Malang. Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan, oleh sebab itu penulis mengharapkan kritik dan saran yang bersifat membangun dari semua pihak demi kesempurnaan skripsi ini.

Oleh karena itu, pada kesempatan ini penulis menyampaikan rasa terima kasih atas segala bimbingan dan bantuan kepada:

1. Allah Swt yang telah memberikan kelancaran dan ilmu yang barokah
2. Zahra, S.Pd., selaku ibu tercinta yang telah menyemangati untuk menyelesaikan skripsi ini.
3. Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D., selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya.
4. Heru Nurwarsito, Ir., M.Kom., selaku Wakil Dekan Akademik Fakultas Ilmu Komputer Universitas Brawijaya.
5. Wibisono Sukmo Wardhono, S.T, M.T., selaku Dosen Pembimbing I Skripsi.
6. Issa Arwani, S.Kom, M.Sc., selaku Dosen Pembimbing II Skripsi.
7. Seluruh Staf dan Karyawan Fakultas Ilmu Komputer Universitas Brawijaya yang tidak bisa saya sebutkan satu persatu.

Penyusun menyadari bahwa masih perlu adanya perbaikan dalam penulisan laporan ini. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun demi perbaikan selanjutnya

Malang, 21 Juli 2016

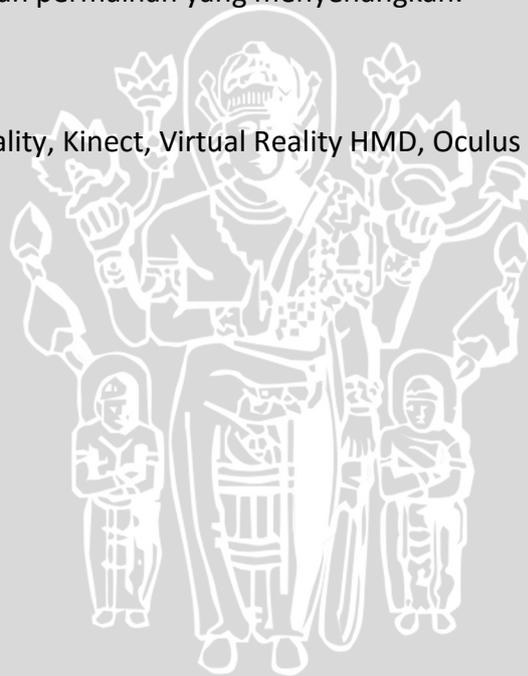
Penulis

muhammad.faisal.3d@gmail.com

ABSTRAK

Virtual Reality adalah teknologi yang cukup baru dan masih berkembang secara pesat. Dengan Virtual Reality, pemain dapat merasakan sensasi seakan-akan dia sedang berada di dalam permainan tersebut. Sayangnya, Virtual Reality yang ada saat ini kebanyakan menggunakan controller konvensional sebagai pengendali permainannya. Game Bridge of Balance ini menggunakan Kinect yang dapat mendeteksi gerakan pemain sebagai pengendali permainan, sehingga tingkat realisme bertambah. Permainan ini memiliki gameplay yang simple, dimana pemain diharuskan berjalan melewati jembatan di atas tebing yang tinggi. Gameplay yang simple ini bertujuan untuk memudahkan pemain yang masih awam dengan Virtual Reality dapat menikmati permainan tersebut, serta memaksimalkan fungsi dari Kinect dalam mendeteksi gerakan pemain. Pada pengujian playtesting yang dilakukan pada 25 orang, 24 orang menjawab bahwa Bridge of Balance adalah permainan yang menyenangkan.

Kata Kunci: Virtual Reality, Kinect, Virtual Reality HMD, Oculus Rift



ABSTRACT

Virtual Reality is a pretty new technology that is developing really quickly. With Virtual Reality, player can feel as if they are really inside the game. Unfortunately, most Virtual Reality games currently only used conventional controller to control the game. Bridge of Balance game used Kinect to detect the movement of the player to control the game, so the realism value will increase. This game used simple gameplay, where the player must walk through a bridge on top of a tall cliff. The gameplay is simple so a new player in the field of Virtual Reality can enjoy the game, and maximize the function of Kinect in tracking player's movement. From the playtesting that had been done with 25 respondent, 24 answered that Bridge of Balance is a fun game.

Keywords: Virtual Reality, Kinect, Virtual Reality HMD, Oculus Rift



DAFTAR ISI

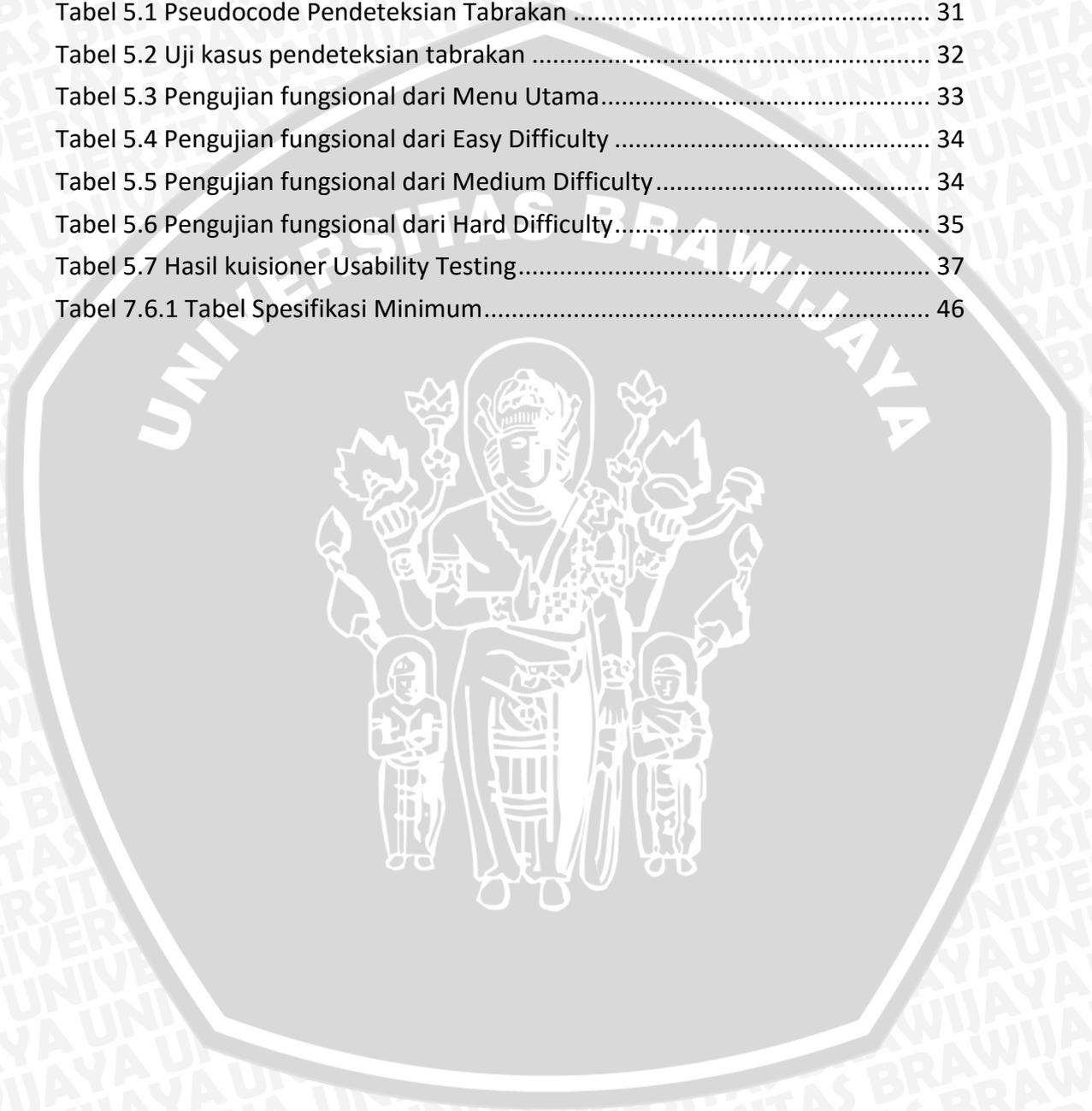
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	v
ABSTRACT.....	vi
DAFTAR ISI.....	vii
DAFTAR TABEL.....	ix
DAFTAR GAMBAR.....	x
DAFTAR LAMPIRAN	xi
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	1
1.3 Tujuan	2
1.4 Manfaat.....	2
1.5 Batasan masalah	2
1.6 Sistematika pembahasan.....	2
BAB 2 LANDASAN KEPUSTAKAAN	4
2.1 Permainan Virtual Reality	4
2.2 Virtual Reality Head Mounted Display.....	4
2.2.1 Cara Kerja Virtual Reality Head Mounted Display	4
2.2.2 Vendor Virtual Reality Head Mounted Display	4
2.3 Kinect	7
2.3.1 Teknologi.....	7
2.3.2 Kinect for Windows.....	7
2.4 Unity.....	7
2.5 Blender.....	8
2.6 Iterative with Rapid Prototyping	9
2.7 Pengujian Game	9
2.7.1 White-box Testing.....	9
2.7.2 Black-box Testing	9
2.7.3 Playtesting.....	10
2.8 MDA Framework.....	11
2.9 Obyek 3D.....	12
2.9.1 <i>Polygon</i>	12

BAB 3 METODOLOGI	13
3.1 Studi Literatur	13
3.2 Perancangan Game Bridge of Balance.....	13
3.3 Implementasi Game Bridge of Balance	14
3.4 Pengujian	14
BAB 4 PERANCANGAN DAN IMPLEMENTASI	15
4.1 Perancangan Game.....	15
4.1.1 Formal Elements	15
4.1.2 Iterative with Rapid Prototyping	16
4.2 Implementasi	22
4.2.1 Implementasi Game Bridge of Balance.....	22
4.2.2 Implementasi Kinect	26
4.2.3 Implementasi Oculus Rift.....	29
BAB 5 PEMBAHASAN.....	31
5.1 White Box Testing.....	31
5.1.1 Basis Path Testing.....	31
5.2 Black Box Testing	33
5.2.1 Pengujian Menu Utama	33
5.2.2 Pengujian Easy Difficulty.....	33
5.2.3 Pengujian Medium Difficulty.....	34
5.2.4 Pengujian Hard Difficulty	35
5.3 Usability Testing.....	35
5.3.1 Hasil Pengujian Usability.....	36
5.3.2 Pembahasan Pengujian Permainan Bridge of Balance	37
BAB 6 Penutup	43
6.1 Kesimpulan.....	43
6.2 Saran	43
DAFTAR PUSTAKA.....	44
LAMPIRAN A TUTORIAL PENGGUNAAN GAME BRIDGE OF BALANCE (BOB) DENGAN MENGGUNAKAN KINECT DAN OCULUS RIFT	46



DAFTAR TABEL

Tabel 3.1 Penggunaan Program dalam Implementasi	14
Tabel 4.1 <i>Source Code</i> Sistem Pengecekan Kondisi Tabrakan	25
Tabel 5.1 Pseudocode Pendeteksian Tabrakan	31
Tabel 5.2 Uji kasus pendeteksian tabrakan	32
Tabel 5.3 Pengujian fungsional dari Menu Utama.....	33
Tabel 5.4 Pengujian fungsional dari Easy Difficulty	34
Tabel 5.5 Pengujian fungsional dari Medium Difficulty	34
Tabel 5.6 Pengujian fungsional dari Hard Difficulty.....	35
Tabel 5.7 Hasil kuisoner Usability Testing.....	37
Tabel 7.6.1 Tabel Spesifikasi Minimum.....	46



DAFTAR GAMBAR

Gambar 2.1 Headset Oculus Rift	5
Gambar 2.2 Headset HTC Vive	6
Gambar 2.3 Komponen MDA Framework.....	11
Gambar 3.1 Blok Diagram Runtutan Pengerjaan Skripsi	13
Gambar 4.1 <i>Iterative with Rapid Prototyping</i>	16
Gambar 4.2 Dadu	17
Gambar 4.3 Desain board dari permainan.....	17
Gambar 4.4 Desain awal Easy Difficulty	18
Gambar 4.5 Desain awal Medium Difficulty	18
Gambar 4.6 Desain awal Hard Difficulty	19
Gambar 4.7 Prototype Easy Difficulty	20
Gambar 4.8 Prototype Medium Difficulty	21
Gambar 4.9 Prototype Hard Difficulty	21
Gambar 4.10 Implementasi Aktor.....	22
Gambar 4.11 Implementasi Easy Difficulty.....	23
Gambar 4.12 Implementasi Medium Difficulty	24
Gambar 4.13 Implementasi Hard Difficulty	24
Gambar 4.14 Daftar Bone yang di deteksi oleh Kinect	27
Gambar 4.15 Contoh tampilan script Kinect Manager	28
Gambar 4.16 Contoh tampilan script Avatar Controller.....	29
Gambar 5.1 <i>Flow Graph</i> algoritma pendeteksian tabrakan	32

DAFTAR LAMPIRAN

LAMPIRAN A TUTORIAL PENGGUNAAN GAME BRIDGE OF BALANCE (BOB)
DENGAN MENGGUNAKAN KINECT DAN OCULUS RIFT 46



BAB 1 PENDAHULUAN

1.1 Latar belakang

Virtual reality adalah teknologi yang memungkinkan seseorang melakukan simulasi terhadap suatu objek nyata dengan menggunakan komputer yang mampu membangkitkan suasana tiga dimensi (3-D) sehingga membuat pemakai seolah-olah terlibat secara fisik. Sistem seperti ini dapat digunakan untuk peramu obat, arsitek, pekerja medis, dan bahkan orang awam untuk melakukan aktivitas-aktivitas yang meniru dunia nyata. Sebagai contoh, pilot dapat menggunakan sistem *virtual reality* untuk melakukan simulasi penerbangan sebelum melakukan penerbangan yang sesungguhnya (Jackson, 2015).

Kelebihan VR dibandingkan dengan sistem visual dengan menggunakan monitor adalah player dapat merasakan sensasi seakan berada di dalam permainan tersebut. Akan tetapi, sistem pengendali yang ada saat ini kebanyakan masih menggunakan tombol untuk melakukan gerakan, sehingga permainan menjadi kurang realistis dan player kurang merasakan sensasi seakan berada di dalam permainan tersebut. Oleh karena itu, penulis ingin membuat sebuah permainan *virtual reality* yang dinamakan Bridge of Balance (BOB) dengan memanfaatkan Oculus Rift sebagai perangkat *virtual reality* dan Kinect yang dapat mengenali gerakan user sebagai pengendalinya.

Oculus Rift adalah sebuah perangkat *Virtual Reality Head Mounted Display* (VR-HMD) yang dikembangkan oleh Oculus VR. Pada permainan ini, Oculus Rift digunakan sebagai penampil dari dunia permainan. Kemudian gerakan pemain dideteksi dengan menggunakan sensor Kinect, sehingga pemain dapat menggunakan gerakan tubuhnya untuk mengendalikan permainan, dan tidak perlu menggunakan tombol untuk melakukan pergerakan pada permainan. Permainan ini dinamakan Bridge of Balance dikarenakan pada permainan ini, pemain akan diharuskan untuk berjalan melewati jembatan tanpa terjatuh. Pada tingkat kesulitan yang lebih tinggi, jembatan yang ada akan menjadi semakin sempit, sehingga pengguna harus menjaga keseimbangannya agar tidak terjatuh.

1.2 Rumusan masalah

Dengan latar belakang yang telah dipaparkan di atas, maka rumusan masalah adalah:

1. Bagaimana rancangan dan implementasi permainan “BOB”?
2. Bagaimana mengintegrasikan Oculus Rift sebagai *virtual reality display* pada permainan “BOB”?
3. Bagaimana mengintegrasikan Kinect sebagai pengendali karakter pada permainan “BOB”?
4. Bagaimana minat pengguna terhadap permainan “BOB” yang menggunakan Oculus Rift dan Kinect?

1.3 Tujuan

Berdasarkan rumusan masalah yang telah dipaparkan di atas, maka tujuan dari skripsi ini adalah:

1. Merancang dan mengimplementasikan permainan "BOB".
2. Mengimplementasikan Oculus Rift sebagai *virtual reality display* pada permainan "BOB".
3. Mengimplementasikan Kinect sebagai pengendali karakter pada permainan "BOB".
4. Mengetahui minat terhadap permainan "BOB" yang menggunakan Oculus Rift dan Kinect.

1.4 Manfaat

Manfaat yang dapat diperoleh dari penelitian ini adalah:

1. Memahami cara membuat sebuah permainan.
2. Memahami cara mengintegrasikan permainan yang ada dengan teknologi virtual reality oculus rift.
3. Memahami cara memanfaatkan kinect untuk mengendalikan gerakan karakter berdasarkan gerakan player.

1.5 Batasan masalah

Agar tidak menyimpang dari maksud dan tujuan, maka ruang lingkup penelitian dibatasi pada pengembangan permainan yang memanfaatkan teknologi Virtual Reality. Dengan ruang lingkup tersebut, maka diperlukan adanya batasan pokok-pokok permasalahan sebagai berikut:

1. Permainan bersifat sudut pandang pertama.
2. Permainan menggunakan gameplay yang simple.
3. Permainan memiliki sistem *difficulty* sebanyak 3 jenis *difficulty* tanpa menggunakan level.

1.6 Sistematika pembahasan

Bab I Pendahuluan

Memuat latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dan sistematika penulisan.

Bab II Dasar Teori

Memaparkan teori dasar dan teori pendukung yang berhubungan dengan pembuatan permainan virtual reality dengan menggunakan kinect sebagai pengendali.

Bab III Metodologi

Berisi tentang langkah-langkah dalam melakukan observasi, pengumpulan, dan pengolahan data. Data yang dicari terkait dengan kapabilitas dari teknologi virtual reality dan kinect.

Bab IV Perancangan dan Implementasi

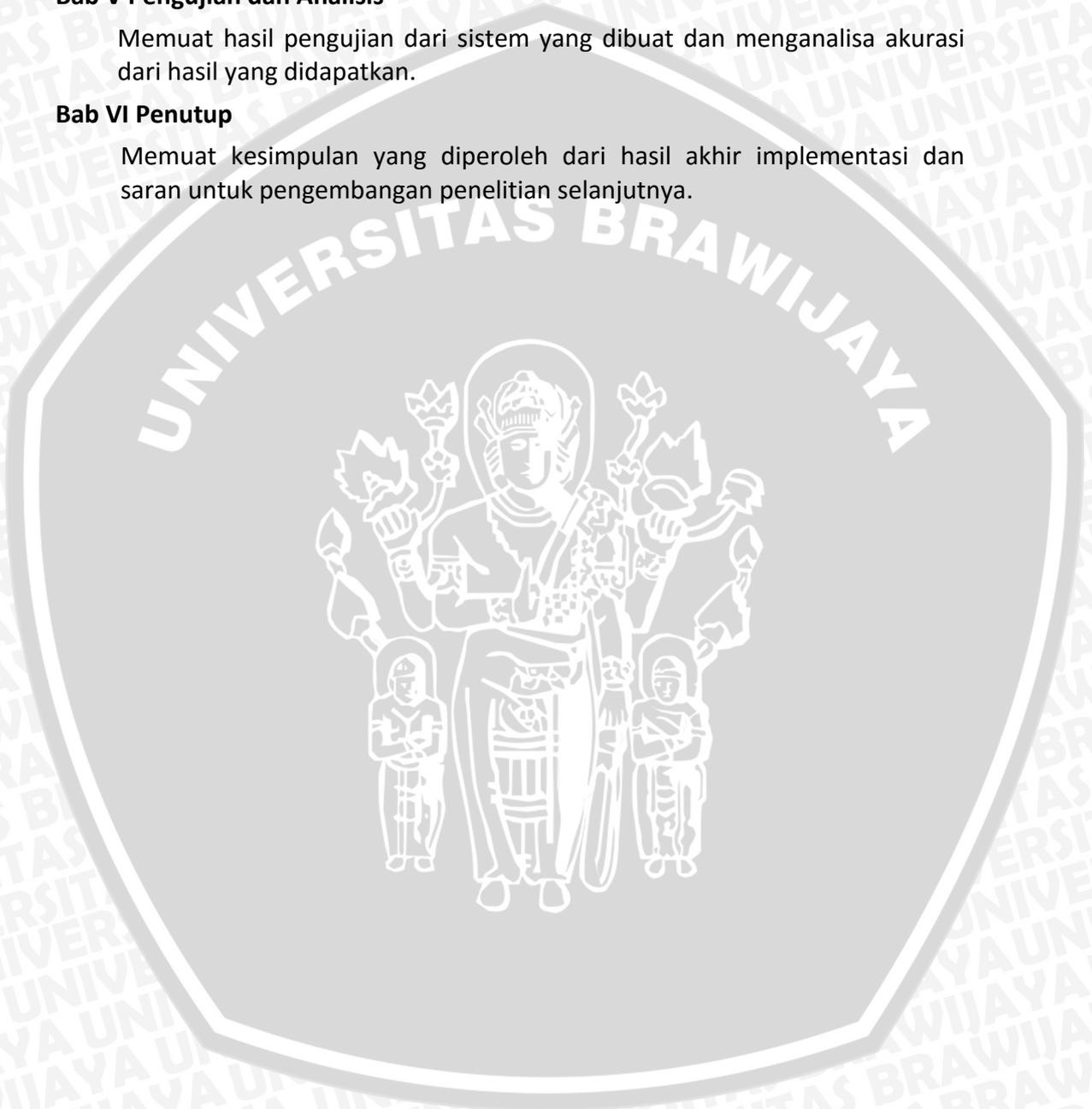
Membahas mengenai langkah-langkah perancangan permainan berdasarkan analisa data yang telah dikumpulkan. Selain itu akan dibahas tentang implementasi dari pembuatan permainan balance bridge dengan menggunakan kinect dan *Virtual Reality Head Mounted Display*.

Bab V Pengujian dan Analisis

Memuat hasil pengujian dari sistem yang dibuat dan menganalisa akurasi dari hasil yang didapatkan.

Bab VI Penutup

Memuat kesimpulan yang diperoleh dari hasil akhir implementasi dan saran untuk pengembangan penelitian selanjutnya.



BAB 2 LANDASAN KEPUSTAKAAN

2.1 Permainan Virtual Reality

Permainan merupakan suatu sistem yang memiliki aturan-aturan tertentu dimana pemain akan terlibat di dalam suatu permasalahan sehingga dapat menghasilkan suatu hasil yang dapat diukur yaitu menang atau kalah (salen, 2003).

Virtual reality atau realitas maya adalah teknologi yang membuat pengguna dapat berinteraksi dengan suatu lingkungan yang disimulasikan oleh komputer (computer-simulated environment), suatu lingkungan sebenarnya yang ditiru atau benar-benar suatu lingkungan yang hanya ada dalam imajinasi (Sihite, 2013).

Permainan *virtual reality* adalah sebuah jenis permainan dimana pengguna dapat berinteraksi dengan suatu lingkungan yang disimulasikan oleh komputer untuk menjadi terlibat dalam suatu permasalahan sehingga dapat menghasilkan suatu hasil yang dapat diukur yaitu menang atau kalah.

2.2 Virtual Reality Head Mounted Display

Virtual reality adalah teknologi yang membuat pengguna dapat berinteraksi dengan suatu lingkungan yang disimulasikan oleh komputer (computer-simulated environment), suatu lingkungan sebenarnya yang ditiru atau benar-benar suatu lingkungan yang hanya ada dalam imajinasi (Sihite, 2013). Lingkungan realitas maya terkini umumnya menyajikan pengalaman visual, yang ditampilkan pada sebuah layar komputer atau melalui sebuah penampil stereoskopik, tapi beberapa simulasi mengikutsertakan tambahan informasi hasil pengindraan, seperti suara melalui speaker atau headphone (McKay, 2008).

2.2.1 Cara Kerja Virtual Reality Head Mounted Display

Virtual Reality Head Mounted Display bekerja dengan cara memproyeksikan gambar yang berbeda ke dalam setiap mata, hal ini mensimulasikan pandangan mata kita pada dunia nyata yang melihat pemandangan dengan sudut yang sedikit berbeda dikarenakan perbedaan posisi pada mata (Citra, 2008). *Virtual Reality Head Mounted Display* juga memiliki berbagai macam sensor seperti *gyroscope* dan *accelerometer* yang digunakan untuk mendeteksi gerakan kepala dari pemain, sehingga pemain dapat menggunakan gerakan kepala untuk merubah pandangan dari headset.

2.2.2 Vendor Virtual Reality Head Mounted Display

Virtual reality sedang mengalami masa perkembangan yang sangat cepat, oleh karena itu saat ini banyak perusahaan raksasa yang ikut membuat *Virtual Reality Head Mounted Display* sendiri. Berikut beberapa contohnya:

1. Oculus Rift

Oculus Rift adalah *Virtual Reality Head Mounted Display* yang dikembangkan oleh Oculus VR. Pada awalnya, Oculus Rift ditawarkan melalui Kickstarter, dimana Oculus VR mendapatkan bantuan sebesar 2,5 Juta Dolar untuk pengembangan produk tersebut (Kickstarter, 2012).

Oculus Rift dijadwalkan untuk meluncur pada tanggal 28 Maret 2016, membuatnya menjadi salah satu *headset virtual reality* pertama yang di targetkan untuk level konsumen. Headset ini memiliki resolusi 1080x1200 untuk tiap mata, refresh rate sebesar 90hz, dan field of view yang lebar. (Binstock, 2015) Oculus Rift memiliki headphone yang terintegrasi yang memberikan effect suara 3D. headset ini juga memiliki pelacak rotasi dan posisi. Pelacak posisi dilakukan oleh sebuah sensor IR yang di hubungkan dengan USB, yang biasanya diletakkan pada meja pengguna, hal ini memungkinkan penggunaan headset pada saat duduk, berdiri, atau berjalan di sekitar ruangan (Oculus VR, 2015). Tampilan dari perangkat Oculus Rift ditunjukkan pada gambar 2.1.



Gambar 2.1 Headset Oculus Rift

2. Google Cardboard

Google Cardboard adalah headset *Virtual Reality* (VR) dan *Augmented Reality* (AR) yang dikembangkan oleh Google untuk digunakan dengan perangkat mobile. Headset tersebut dinamakan Cardboard dikarenakan bahan pembuatannya yang merupakan karton yang dilipat, sehingga headset ini dapat didapatkan dengan murah. Headset ini dibuat untuk mengenalkan dan meningkatkan minat terhadap VR dan AR (Branstetter, 2015). Pengguna dapat membuat headset mereka sendiri dengan menggunakan komponen yang murah sesuai dengan spesifikasi yang telah dipublikasikan oleh Google, atau membeli salah satu headset yang telah dibuat oleh pihak ketiga.

3. Samsung Gear VR

Samsung Gear VR adalah *headset virtual reality* yang dikembangkan oleh Samsung Electronics dan bekerjasama dengan Oculus VR. Headset ini membutuhkan perangkat mobile Samsung Galaxy yang kompatibel (Galaxy Note 5 or Galaxy S6/S6 Edge) dalam penggunaannya. Perangkat mobile tersebut

digunakan dalam *display* dan *processor* dari *headset*, sementara unit Gear VR itu sendiri berisi lensa dengan field of view yang tinggi serta sebuah inertial measurement unit (IMU) yang digunakan untuk melacak rotasi yang dihubungkan pada ponsel cerdas melalui micro-USB. IMU ini lebih akurat dan terkalibrasi dengan *latency* yang lebih kecil dibandingkan dengan IMU yang berada pada ponsel cerdas biasa seperti yang digunakan pada Google Cardboard (VRS Group, 2016).

Headset Gear VR juga memiliki *touchpad* dan tombol kembali pada bagian sisinya, serta sebuah sensor *proximity* yang digunakan untuk mendeteksi ketika *headset* digunakan. *Touchpad* dan tombol memungkinkan adanya kapabilitas *input* standar minimal untuk pengguna dalam berinteraksi dengan lingkungan virtual, dimana Google Cardboard hanya memiliki sebuah tombol (VRS Group, 2016).

4. HTC Vive

HTC Vive adalah headset virtual reality yang sedang dikembangkan oleh kolaborasi antara HTC dan Valve Corporation. HTC Vive memiliki *refresh rate* sebesar 90hz, memungkinkan konten untuk di *render* dengan kecepatan 90 fps. Perangkat ini menggunakan 2 layar, satu untuk tiap mata, masing-masing memiliki resolusi sebesar 1080x1200. Perangkat ini menggunakan lebih dari 70 sensor diantaranya yaitu sebuah MEMS *gyroscope*, *accelerometer* dan sensor posisi laser, dan dikatakan dapat beroperasi dalam ruang gerak sebesar 4,5 x 4,5 meter apabila digunakan dengan stasiun dasar "Lighthouse". (Kelion, 2015) Sistem Lighthouse didesain oleh Alan Yates dan menggunakan *photosensors* simple pada tiap object yang perlu di capture.

Adanya kamera di depan memungkinkan software untuk mengenali object diam atau bergerak pada ruangan, fungsi ini digunakan sebagai bagian dari sistem keamanan "Chaperone", yang akan menampilkan secara otomatis data dari kamera untuk memperingatkan user mengenai rintangan. Gambar 2.2 menunjukkan tampilan dari perangkat HTC Vive.



Gambar 2.2 Headset HTC Vive

2.3 Kinect

Kinect for Xbox 360 atau biasa disebut Kinect (Dulunya di kenal sebagai Project Natal), adalah "*controller-free gaming* dan pengalaman hiburan" yang dibuat oleh Microsoft pada platform video permainan Xbox 360.

2.3.1 Teknologi

Sensor Kinect adalah batang horizontal yang terhubung dengan alas kecil yang memiliki poros yang dapat berputar. Sensor Kinect dirancang untuk diletakkan diatas maupun di bawah TV. Perangkat ini memiliki kamera RGB, sensor kedalaman dan mikrofon yang berjalan di perangkat software khusus, yang menyediakan kemampuan untuk menangkap gerak secara 3D, mengenali wajah dan mengenali suara (Salah, 2011).

Sensor kedalaman Kinect terdiri dari proyektor laser *infrared* dikombinasikan dengan sensor CMOS monokromatik, yang merekam data video 3D dalam kondisi pencahayaan apapun. Area pengenalan sensor kedalaman dapat diatur, dan perangkat lunak Kinect secara otomatis mampu mengkalibrasikan sensor berdasarkan permainan dan kondisi lingkungan pemain, mengenali keberadaan furnitur maupun halangan lain. Kinect secara simultan mampu mengenali hingga enam orang, termasuk dua pemain aktif untuk dianalisis gerakannya pada 20 titik sendi tiap orang (Salah, 2011).

2.3.2 Kinect for Windows

Pada tanggal 21 Februari 2011, Microsoft mengumumkan bahwa mereka akan meluncurkan Kinect software development kit (SDK) yang bersifat non komersil untuk Windows pada musim semi 2011, yang diluncurkan untuk Windows 7 pada tanggal 16 Juni 2011 di 12 negara. SDK tersebut memiliki driver untuk perangkat Kinect yang kompatibel dengan Windows 7. Hal ini memberikan kapabilitas Kinect untuk para pengembang sehingga mereka dapat mengembangkan aplikasi dengan menggunakan bahasa C++, C#, atau Visual Basic dengan menggunakan Microsoft Visual Studio 2010 dan memiliki fitur berikut ini:

1. Data sensor mentah: Mengakses hasil dari sensor kedalaman, sensor warna kamera, dan rangkaian 4 elemen mikrofon dari Kinect.
2. *Skeletal tracking*: Kapabilitas untuk melacak dan mengikuti gambar kerangka dari satu atau dua orang yang bergerak di dalam jarak pandang Kinect untuk digunakan pada aplikasi yang menggunakan gestur.
3. Kapabilitas suara yang canggih: Kapabilitas pemrosesan suara termasuk peredam suara akustik yang canggih dan peredam gema, beam formation untuk mengenali sumber suara yang ada, dan integrasi dengan API pengenalan suara milik Windows (Microsoft, 2011).

2.4 Unity

Unity adalah game engine cross-platform yang dikembangkan oleh Unity Technologies dan digunakan untuk mengembangkan permainan untuk PC,

konsol, perangkat *mobile* dan *website*. Pada saat pertama kali diumumkan pada tahun 2005, Unity hanya ditujukan untuk OS X, sekarang Unity telah diperluas untuk menarget lebih dari 15 *platform*. Sekarang Unity adalah Software Development Kit (SDK) utama untuk pengembangan pada Wii U.

Unity memiliki berbagai fitur yang dapat memudahkan dalam pembuatan permainan, antara lain:

1. *Physically-Based Shading*: Memungkinkan pembuatan shader yang mensimulasikan interaksi antara material dan cahaya seperti dalam dunia nyata, sehingga material menjadi lebih realistis.
2. Sistem Partikel Shuriken: Memungkinkan pembuatan efek partikel yang canggih dan mudah untuk dikonfigurasi.
3. *Inverse Kinematics*: Berfungsi untuk menggerakkan rig dari karakter secara natural sehingga dapat memudahkan pembuatan animasi dan gerakan karakter.
4. *Level of Detail*: Membuat obyek yang dapat terlihat semakin detail pada saat obyek tersebut berada dekat dengan kamera sehingga dapat menambah performa dari permainan.
5. *Super-Accurate Collision Detection*: Adanya berbagai mode Collision Detection yang akurat dan cepat untuk berbagai macam kebutuhan dan tipe permainan (Helgason, 2012).

2.5 Blender

Blender adalah aplikasi komputer grafis 3D profesional yang bersifat *open source* dan gratis. Blender dapat digunakan untuk membuat film animasi, *visual effect*, seni, aplikasi 3D interaktif, dan game. Fitur dari Blender antara lain adalah 3D *modelling*, *UV Unwrapping*, *texturing*, pengeditan gambar raster, *rigging*, *skinning*, simulasi cairan dan asap, simulasi partikel, *sculpting*, animasi, *camera tracking*, *rendering*, hingga pengeditan video dan pembuatan game (Blender Foundation, 2016).

Pada awalnya, Blender dikembangkan oleh studio animasi belanda yang bernama Neo Geo untuk digunakan dalam studionya pada tahun 1995. Ketika Neo Geo dibeli oleh perusahaan lain, Ton Roosendaal dan Frank van Beek membuat perusahaan bernama Not a Number Technologies (NaN) pada tahun 1998 untuk mengembangkan Blender lebih jauh. Pada awalnya, NaN mendistribusikan Blender sebagai *Shareware*, hingga NaN mengalami bangkrut pada tahun 2002.

Pada juli 2002, Roosendaal membuat kampanye *crowdfunding* yang diberi name "*Free Blender*". Kampanye tersebut dibuat untuk menjadikan Blender menjadi *open source* dengan biaya sebesar €100,000 yang didapatkan dari komunitas. Pada september 2002, Roosendaal mengumumkan bahwa mereka telah mendapatkan cukup dana dan akan meluncurkan *source code* dari Blender (Blender Foundation, 2013).

2.6 Iterative with Rapid Prototyping

Iterative with rapid prototyping adalah tahap perancangan dimana dibuat sebuah *prototype* dari permainan yang akan dibuat untuk kemudian dilakukan pengujian langsung kepada orang-orang yang sesuai dengan target *demographic* permainan tersebut dengan menggunakan MDA Framework untuk mendapatkan timbal balik dari gameplay yang sudah dirancang, setelah di dapatkan sebuah timbal balik dari pengguna, pengembang dapat menerapkannya ke dalam permainan yang telah dibuat, untuk kemudian kembali dibuat menjadi *prototype* yang akan diuji. Metode ini digunakan untuk mengurangi atau menghilangkan sebanyak mungkin bug ataupun menambah fitur berdasarkan keinginan pengguna sehingga di dapatkan permainan yang menarik.

2.7 Pengujian Game

Dalam pembuatan game diperlukan pengujian untuk memenuhi persyaratan yang menuntut stabilitas, kompatibilitas dan kinerja yang baik.

Tujuan dari pengujian game adalah untuk menyampaikan informasi kepada pengembang mengenai kesalahan yang terjadi dalam game. Berikut adalah proses dari pengujian game:

1. Merencanakan dan mendesain pengujian
2. Menyiapkan pengujian
3. Menjalankan pengujian
4. Melaporkan hasil dari pengujian
5. Memperbaiki kesalahan dari game
6. Kembali ke langkah pertama dan melakukan pengujian ulang hingga tidak terdapat kesalahan.

2.7.1 White-box Testing

White-box testing atau pengujian struktural adalah pengujian terhadap *source code*. Pada pengujian ini, penguji harus memiliki pengetahuan penuh mengenai struktur *internal* dari aplikasi yang akan diuji. Pengujian struktural biasanya diaplikasikan pada bagian kecil dari aplikasi seperti subrutin, atau operasi yang diasosiasikan dengan sebuah obyek (Agarwal, 2009).

2.7.2 Black-box Testing

Black-box testing adalah pengujian terhadap fungsionalitas. Pada pengujian fungsionalitas, struktur dari aplikasi tidak diperhatikan. Kasus uji ditentukan berdasarkan basis dari kebutuhan atau spesifikasi dari aplikasi.

Pengujian fungsional mengacu pada pengujian, yang hanya meliputi observasi keluaran dari nilai masukan tertentu, dan tidak terdapat percobaan untuk menganalisa kode yang menghasilkan keluaran tersebut. Struktur internal dari aplikasi diacuhkan (Agarwal, 2009).

2.7.3 Playtesting

Playtesting merupakan proses mempelajari bagaimana reaksi dan kesenangan pemain mengenai aktifitas yang telah dikerjakan. Proses ini melibatkan seseorang untuk memainkan game tanpa tujuan untuk menemukan kesalahan dari game. Playtesting digunakan untuk mengetahui pandangan terhadap unsur *balance*, kesulitan, dan minat pemain terhadap game. Tidak seperti pengujian game sebelumnya, *playtesting* mengarah ke pertanyaan opini dan penilaian setiap pemain terhadap fakta-fakta game tersebut. Playtesting menjawab pertanyaan-pertanyaan penting seperti:

1. Apakah game mudah dimainkan?
2. Apakah game menyenangkan?

Untuk pengujian playtesting pada skripsi ini, digunakan sepuluh heuristik usability untuk desain antarmuka pengguna yang dirancang oleh Jakob Nielsen pada tahun 1995. Disebut heuristik dikarenakan hal ini hanyalah merupakan panduan umum, dan bukan merupakan panduan usability yang spesifik.

Berikut adalah heuristik usability rancangan Jakob Nielsen:

1. Keterlihatan status sistem
Sistem harus selalu menginformasikan pengguna mengenai apa yang sedang terjadi, melalui *feedback* yang sesuai pada waktu yang tepat.
2. Kesesuaian antara sistem dan dunia nyata
Sistem harus berbicara bahasa pengguna, dengan kata, kalimat, dan konsep yang familiar dengan pengguna. Menggunakan konvensi dunia nyata, membuat informasi tampil dengan urutan yang logis dan natural.
3. Kebebasan pengguna
Pengguna sering salah memilih fungsi dari sistem dan akan membutuhkan jalur keluar yang jelas untuk keluar dari kondisi yang tidak diinginkan tanpa perlu melewati dialog yang panjang.
4. Konsistensi dan standar
Pengguna tidak perlu memikirkan apakah kata, situasi, atau aksi yang berbeda memiliki arti yang sama.
5. Pencegahan kerusakan
Hal yang lebih baik daripada pesan kesalahan yang didesain dengan baik adalah rancangan yang dibuat dengan hati-hati sehingga mencegah terjadinya kerusakan.
6. Pengenalan dibandingkan dengan pengingatan
Kurangi beban kerja memori dari pengguna dengan cara membuat obyek, aksi, dan pilihan terlihat. Pengguna tidak perlu mengingat informasi dari satu bagian dialog ke bagian lainnya. Instruksi mengenai penggunaan sistem harus terlihat dan mudah didapatkan kapanpun dibutuhkan.

7. Fleksibilitas dan efisiensi penggunaan

Percepatan yang tidak terlihat oleh pengguna pemula dapat digunakan untuk mempercepat interaksi oleh pengguna yang lebih ahli sehingga sistem dapat digunakan dengan baik oleh pengguna pemula dan ahli.

8. Estetika dan desain yang minimalis

Dialog sebaiknya tidak memiliki informasi yang tidak relevan atau jarang dibutuhkan.

9. Membantu pengguna mengenali, mendiagnosa, dan kembali dari kerusakan

Pesan kesalahan seharusnya dinyatakan dalam bahasa pengguna (tidak dalam kode), menunjukkan kesalahan dengan jelas, dan menyarankan solusi.

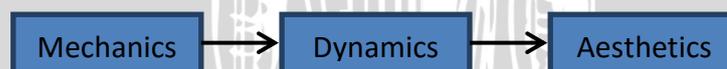
10. Bantuan dan dokumentasi

Meskipun sistem yang baik sebaiknya dapat digunakan tanpa dokumentasi, bantuan dan dokumentasi mungkin perlu disediakan. Informasi tersebut sebaiknya mudah dicari, terfokus pada pekerjaan pengguna, menunjukkan langkah-langkah yang harus dikerjakan, dan tidak terlalu besar.

2.8 MDA Framework

Model pengembangan yang digunakan adalah MDA Framework, alasan menggunakan MDA Framework karena MDA adalah pendekatan formal untuk menjembatani kesenjangan antara desain game dan pengembangan (Hunicke, 2004).

MDA Framework terbagi menjadi beberapa komponen, seperti pada gambar 2.3.



Gambar 2.3 Komponen MDA Framework

1. *Mechanics*

Menggambarkan komponen tertentu dari game, pada tingkat representasi data dan algoritma.

2. *Dynamics*

Menggambarkan perilaku run-time dari mekanik yang bekerja pada input *player* dan setiap output dari waktu ke waktu

3. *Aesthetics*

Menggambarkan dan membangkitkan respon emosional yang diharapkan di pemain, ketika ia berinteraksi dengan sistem permainan (Hunicke, 2004).

2.9 Obyek 3D

Benda 3 dimensi (3D) adalah sebuah objek / ruang yang memiliki panjang, lebar, dan tinggi. Grafik tiga dimensi merupakan teknik penggambaran yang mengacu pada titik koordinat sumbu x (datar), sumbu y (tegak), dan sumbu z (miring). Representasi dari data geometrik tiga dimensi sebagai hasil dari pemrosesan dan pemberian efek cahaya terhadap grafika komputer dua dimensi (2D). Tiga dimensi, biasanya digunakan dalam penanganan grafik.

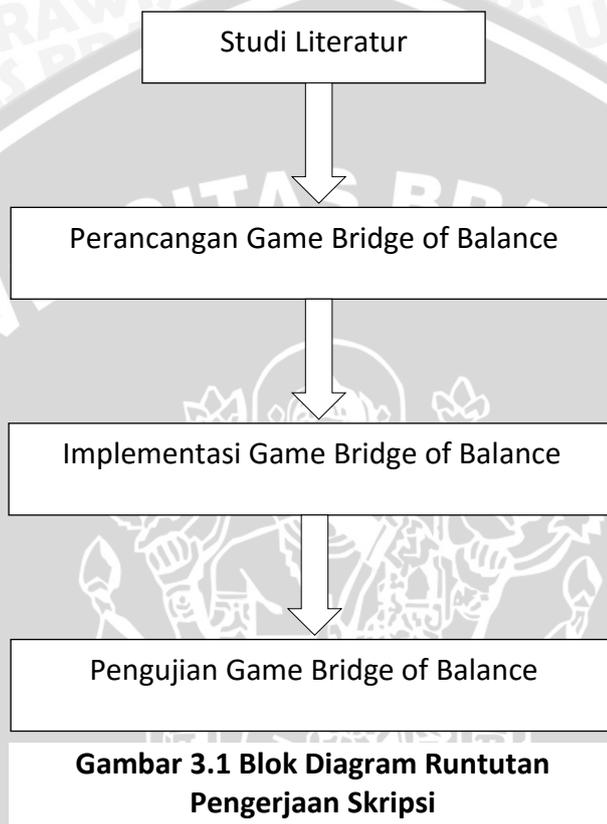
2.9.1 Polygon

Polygon merupakan sebuah bangun datar yang memiliki banyak sudut. Namun dalam pemodelan tiga dimensi, *polygon* merujuk pada kumpulan dari segitiga – segitiga yang membentuk bangun datar lain, atau bahkan bangun ruang yang kasar (Alias, 2004). *Polygon* yang lebih kompleks dapat membuat objek yang terdiri lebih dari tiga simpul. Sekelompok *polygon*, terhubung satu sama lain dengan simpul bersama, umumnya disebut sebagai elemen. Setiap *polygon* yang membentuk elemen disebut wajah (face) atau permukaan. *Polygon* adalah kumpulan dari permukaan tersebut. Terdapat dua (2) macam jenis dari *polygon* yaitu *high poly* dan *low poly*. Objek *low poly* adalah objek yang menggunakan *polygon* dengan jumlah yang relative kecil atau sedikit. Hal ini mengakibatkan proses kalkulasi yang dilakukan oleh *Graphic Card* untuk melakukan *rendering* menjadi lebih mudah dan tidak menghabiskan banyak sumber daya, sehingga permainan dapat berjalan dengan lebih cepat dan lancar.



BAB 3 METODOLOGI

Penelitian ini merupakan salah satu bentuk penelitian implementatif, dimana penelitian ini akan menghasilkan sebuah permainan yang memiliki kapabilitas untuk mengenali gerakan user sebagai pengendalinya. Untuk mendukung kelancaran penelitian ini, maka akan dibuat rancangan penelitian berupa langkah-langkah yang ditunjukkan pada gambar 3.1.



3.1 Studi Literatur

Pada tahap ini, dikumpulkan bahan, informasi, keterangan dan teori yang berhubungan dengan Virtual Reality dan Kinect, khususnya pada sistem operasi Windows dan menggunakan Unity dalam implementasinya. Pencarian dilakukan dalam buku dan konsultasi dengan para ahli atau narasumber serta rujukan dari artikel.

3.2 Perancangan Game Bridge of Balance

Perancangan game dilakukan setelah dilakukan studi literatur. Pada tahap ini, dilakukan penentuan formal element dari game yang akan menentukan identitas dan tujuan dari permainan beserta peraturan-peraturan yang berada dalam permainan tersebut. Setelah itu, dilakukan prototyping untuk menentukan desain awal tampilan dan alur dari permainan.

3.3 Implementasi Game Bridge of Balance

Pada tahap ini dilakukan implementasi berdasarkan rancangan yang telah dibuat pada tahap perancangan game. Implementasi pada skripsi ini dibagi menjadi 3 bagian yaitu implementasi permainan, implementasi Kinect, dan implementasi Oculus Rift. Tabel 3.1 menunjukkan program yang digunakan dalam implementasi:

Tabel 3.1 Penggunaan Program dalam Implementasi

Jenis Implementasi	Nama Program
Permainan	Unity
Level	Blender, Unity

3.4 Pengujian

Pengujian game dilakukan untuk menguji kinerja game sesuai spesifikasi yang sudah ditetapkan. Pengujian dilakukan dengan 2 metode yaitu dengan cara pengujian fungsional (Blackbox Testing) dan *playtesting*.



BAB 4 PERANCANGAN DAN IMPLEMENTASI

4.1 Perancangan Game

Perancangan game terdiri dari 2 tahap yaitu formal element dan prototyping. Formal element adalah rincian deskripsi elemen-elemen dalam game yang menjadi dasar dalam proses implementasi pembuatan permainan. Sedangkan Prototyping adalah tahap perancangan untuk membuat permainan dengan melakukan testing secara berulang sehingga didapatkan permainan yang baik.

4.1.1 Formal Elements

Elemen-elemen pada permainan ini adalah sebagai berikut:

A. Players

Permainan ini bertipe single player, sehingga hanya dapat dimainkan seorang diri.

B. Objectives

Pemain diharuskan untuk berjalan melewati jembatan yang telah disediakan pada level. Pemain akan menang apabila berhasil melewati jembatan tanpa terjatuh.

C. Rules

Peraturan dari permainan ini adalah:

1. Pemain harus berjalan melewati jembatan secepat mungkin
2. Pemain menang apabila berhasil melewati jembatan
3. Pemain kalah apabila terjatuh dari jembatan

D. Resources and Resource Management

Dalam permainan ini hanya terdapat resource waktu yang digunakan untuk menghitung waktu pemain dalam melewati jembatan. Resource tersebut tidak dapat di atur oleh pemain. Waktu berjalan secara otomatis saat pemain menginjak jembatan dan berhenti saat pemain berhasil melewati jembatan atau kalah. Waktu tidak dapat dilihat kecuali setelah pemain berhasil melewati jembatan.

E. Game State

Game state dari permainan ini adalah:

1. Pemain belum menginjak jembatan sehingga waktu belum berjalan
2. Pemain telah menginjak jembatan sehingga waktu berjalan
3. Pemain terjatuh dari jembatan sehingga pemain kalah
4. Pemain Berhasil melewati jembatan sehingga pemain menang

F. Information

Permainan ini hampir tidak menggunakan HUD (Heads-Up Display) sama sekali agar tidak mengganggu pemain dan menambah tingkat realisme. Satu-

satunya informasi yang tersedia untuk player adalah waktu yang akan di tunjukkan pada saat player berhasil menang.

G. Sequencing

Sequencing dari permainan ini adalah pemain memilih level yang akan dimainkan, kemudian pemain menginjak jembatan pada level tersebut. Apabila pemain terjatuh, maka akan ditampilkan layar Game Over. Apabila pemain berhasil melewati jembatan maka akan ditampilkan waktu yang dibutuhkan untuk melewati jembatan tersebut.

H. Player Interaction

Dalam permainan ini tidak ada interaksi antar pemain dikarenakan sifat dari permainan ini yang berupa single player

I. Theme (Narrative, Backstory, Setting)

Permainan ini memiliki 3 level dengan setting yang berbeda-beda.

Pada level pertama, setting yang digunakan adalah perbukitan dengan sebuah sungai yang mengalir di tengahnya. Pemain melewati jembatan untuk mencapai rumahnya.

Pada level kedua, setting yang digunakan adalah sebuah kuil kuno. Pemain melewati jembatan untuk mencari harta karun yang tersembunyi pada kuil tersebut.

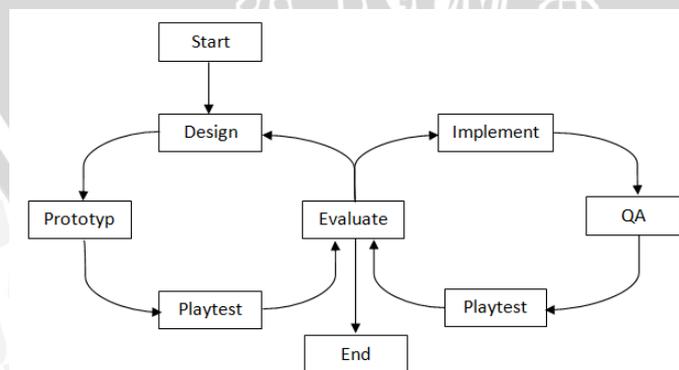
Pada level ketiga, setting yang digunakan adalah neraka. Pemain melewati jembatan untuk menuju ke portal agar bisa keluar dari tempat tersebut.

J. Games as Systems

Game as system adalah permainan sebagai sistem dalam artian sekumpulan formal elemen yang telah disusun sebelumnya saling berhubungan dan membentuk sebuah sistem. Apabila salah satu formal elemen dirubah maka game akan berubah secara signifikan.

4.1.2 Iterative with Rapid Prototyping

Pada gambar 4.1 terdapat alur dari *Iterative with Rapid Prototyping*.



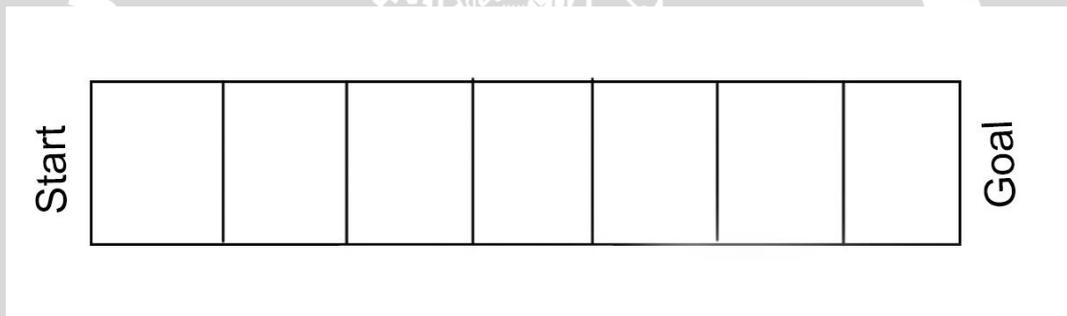
Gambar 4.1 *Iterative with Rapid Prototyping*

A. Paper Prototyping

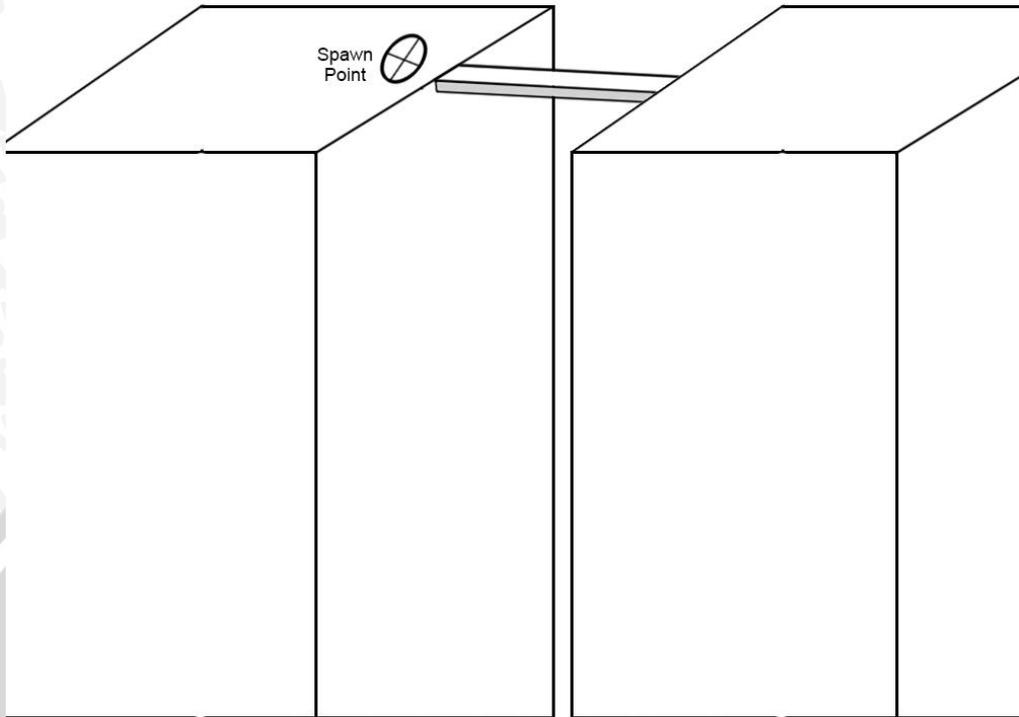
Paper prototype digunakan untuk mengetahui proses interaksi langsung antara pemain dan sistem yang ditampilkan melalui pembuatan permainan nyata berbentuk board game berdasarkan rancangan dari permainan yang telah dibuat. Dalam proses ini pengembang dapat melihat dan mengevaluasi rancangan sistem permainan yang akan dibuat. Berikut adalah gambar dari Paper Prototype dari permainan ini.



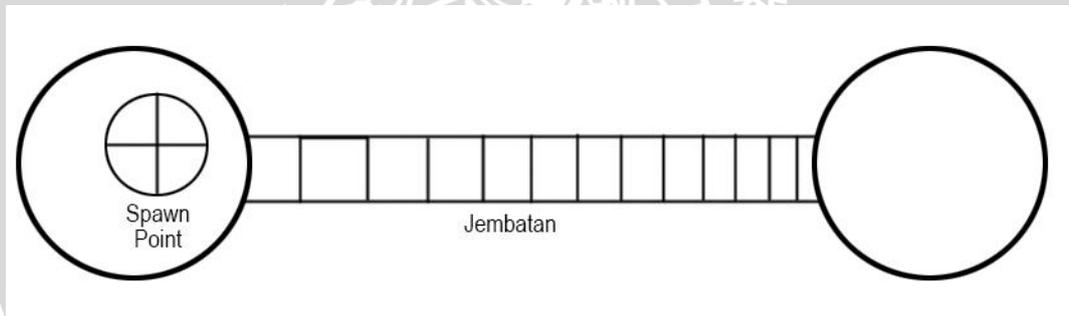
Gambar 4.2 Dadu



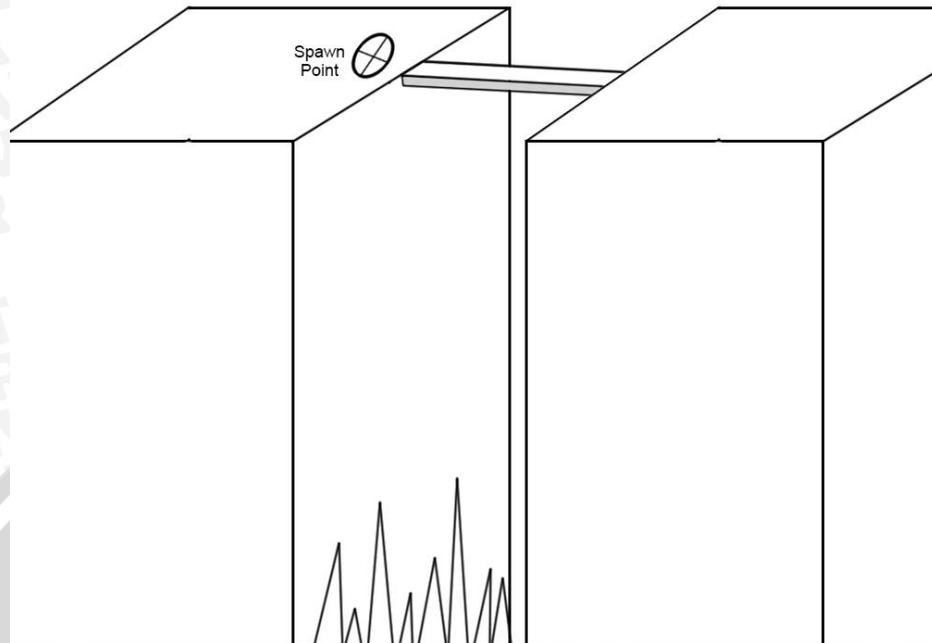
Gambar 4.3 Desain board dari permainan



Gambar 4.4 Desain awal Easy Difficulty



Gambar 4.5 Desain awal Medium Difficulty



Gambar 4.6 Desain awal Hard Difficulty

Dadu pada Gambar 4.2 digunakan untuk mensimulasikan gerakan pemain. Apabila pemain mendapatkan angka genap, maka pemain dapat melangkah maju sebanyak 1 tile, apabila pemain mendapatkan angka ganjil, maka pemain akan terjatuh dari jembatan dan kalah.

Pada gambar 4.3 ditunjukkan board permainan yang terdiri dari tile berukuran 1x7. Pemain dapat melewati tile tersebut tanpa ada batasan selama dadu menunjukkan angka genap.

Pada gambar 4.4 ditunjukkan desain awal dari easy difficulty. Desain awal dari easy difficulty meliputi pembuatan desain tebing dan jembatan. Hal ini dilakukan agar dapat diketahui rasio antara tinggi tebing dari masing-masing difficulty dan juga lebar dari jembatan yang dibutuhkan.

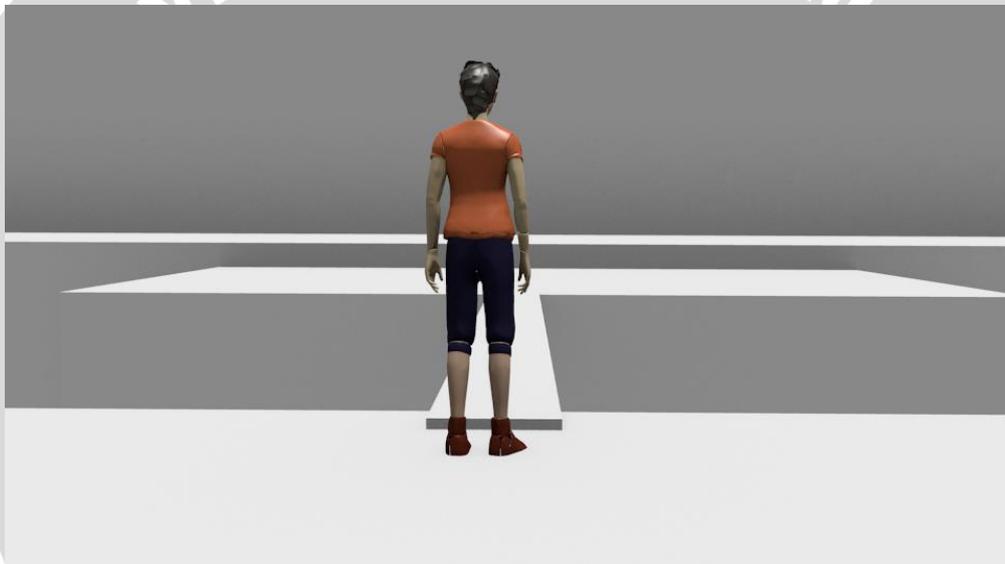
Pada gambar 4.5 ditunjukkan desain awal dari medium difficulty. Desain awal dari medium difficulty meliputi pembuatan desain gantungan dan jembatan yang ada di antaranya. Hal ini dilakukan agar dapat diketahui jarak antara gantungan dan posisi spawn dari pemain.

Pada gambar 4.6 ditunjukkan desain awal dari hard difficulty. Desain awal dari hard difficulty meliputi pembuatan desain tebing dan jembatan. Selain itu terdapat benda tajam yang berada di bawah jembatan. Hal ini dilakukan untuk mencari tahu rasio ketinggian dari tebing dan rasio tingginya benda tajam di dasar tebing tersebut.

B. Digital Prototyping

Tahapan ini adalah proses implementasi secara digital dari paper prototype yang telah dibuat. Prototype ini dibangun untuk memberikan gambaran umum dari gameplay permainan yang telah dibuat.

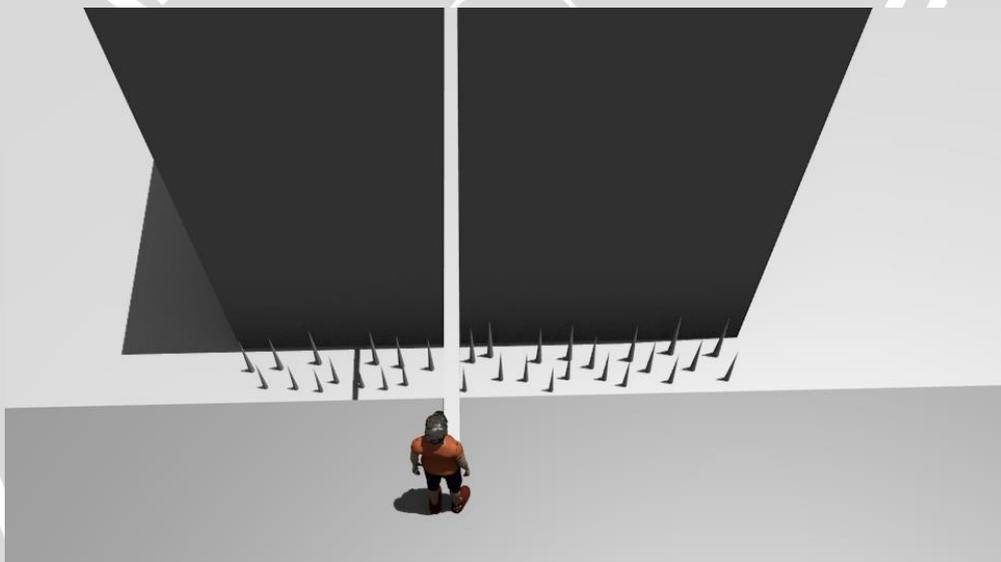
Pada digital prototype ini dibuat rancangan dari level design yang dibangun dengan menggunakan balok untuk mengatur perspektif dan ukuran pemain. Prototype ini juga berguna sebagai dasar lokasi object dalam pembuatan level design nantinya. Dasar dari pemain juga akan dibuat dalam prototype ini agar dapat dipastikan rasio ukuran antara pemain dan level. Selain itu adanya pemain juga dapat berguna untuk mencari bug dan kesalahan pada level. Hasil dari digital prototype ditunjukkan pada gambar berikut ini :



Gambar 4.7 Prototype Easy Difficulty



Gambar 4.8 Prototype Medium Difficulty



Gambar 4.9 Prototype Hard Difficulty

Gambar 4.7 menunjukkan prototype dari Easy Difficulty. Pada prototype ini terdapat 2 balok yang menunjukkan lokasi bukit dengan sebuah papan diantaranya yang menunjukkan lokasi awal jembatan.

Gambar 4.8 menunjukkan prototype dari Medium Difficulty. Pada prototype ini terdapat 2 lingkaran yang menunjukkan lokasi gantungan pada kuil yang akan dibuat dengan sebuah papan diantaranya yang menunjukkan lokasi awal jembatan.

Gambar 4.9 menunjukkan prototype dari Hard Difficulty. Pada prototype ini terdapat 2 balok yang menunjukkan lokasi dataran pada neraka dengan sebuah papan diantaranya yang menunjukkan lokasi awal jembatan. Selain itu, terdapat duri dibawah jembatan yang menunjukkan lokasi awal benda tajam.

4.2 Implementasi

Pada subbab implementasi akan dibahas mengenai penerapan dari perancangan yang telah dibangun sebelumnya dimana pokok bahasan yang akan dibahas pada bab ini adalah penerapan dalam pembuatan level dan mengimplementasikan Oculus rift dan Kinect ke dalam permainan yang telah dibuat.

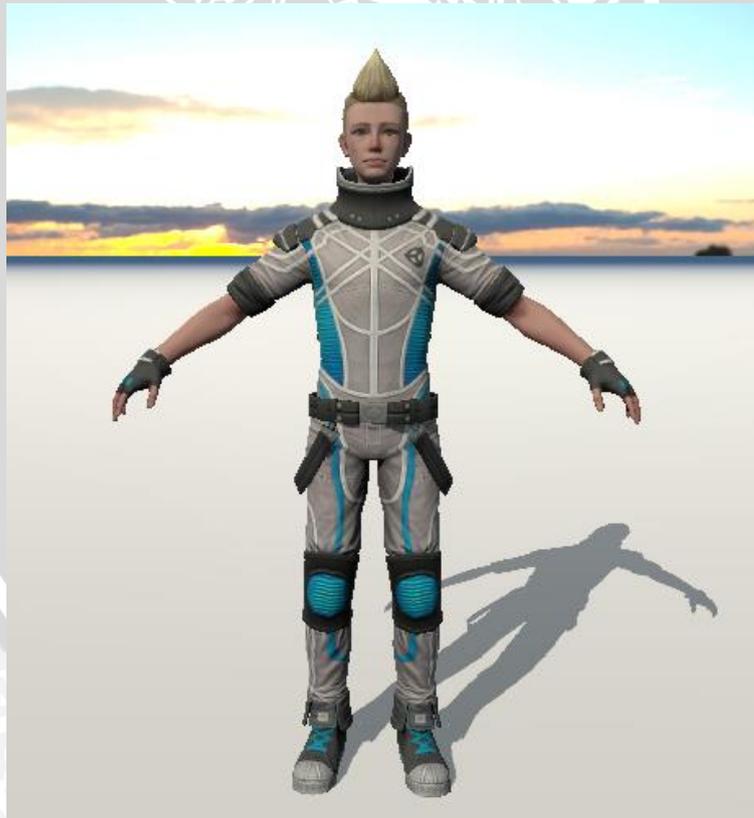
4.2.1 Implementasi Game Bridge of Balance

Pada tahap ini, dibuat permainan berdasarkan perancangan yang telah dibangun sebelumnya. Pada subbab ini akan membahas tentang 3 topik bahasan yaitu:

1. Aktor
2. Level
3. Sistem

A. Implementasi Aktor

Karakter adalah model laki-laki yang digunakan untuk menambah kesan realitas pada saat pengguna bergerak. Kinect digunakan untuk mendeteksi gerakan dan lokasi dari player. Gambar 4.10 menunjukkan hasil dari implementasi aktor:



Gambar 4.10 Implementasi Aktor

Permainan ini memiliki sudut pandang pertama (First Person View). Oleh karena itu, kamera pada permainan diatur menjadi child object dari kepala aktor, sehingga kamera akan mengikuti gerakan dan posisi kepala dari aktor. Adanya implementasi Kinect dalam permainan ini memungkinkan adanya sinkronisasi gerakan dari pemain dengan aktor.

B. Implementasi Level

Pada permainan ini, terdapat 3 difficulty yang memiliki setting dan tempat yang berbeda-beda. Hal ini ditujukan untuk meningkatkan keberagaman dari permainan sehingga pemain tidak mudah merasa jenuh. Masing-masing level memiliki tingkat kesulitan yang berbeda-beda, mulai dari easy difficulty yang paling mudah hingga yang paling sulit yaitu hard difficulty. Gambar 4.11, 4.12, dan 4.13 menunjukkan hasil dari implementasi masing-masing difficulty.



Gambar 4.11 Implementasi Easy Difficulty



Gambar 4.12 Implementasi Medium Difficulty



Gambar 4.13 Implementasi Hard Difficulty

Easy Difficulty yang ditunjukkan pada gambar 4.11 memiliki setting pegunungan dengan sebuah sungai di antara 2 buah tebing. Terdapat sebuah jembatan yang menghubungkan kedua tebing tersebut. Pada ujung dari jembatan tersebut terdapat sebuah rumah gubuk. Di bawah jembatan tersebut terdapat sungai. Pemain diharuskan melewati jembatan tersebut tanpa terjatuh ke sungai.

Medium Difficulty yang ditunjukkan pada gambar 4.12 memiliki setting sebuah kuil raksasa. Terdapat dua buah gantungan pada langit-langit kuil tersebut, dan ada sebuah jembatan yang menghubungkan kedua gantungan tersebut. Di bawah jembatan tersebut terdapat api yang menyala dari obor raksasa. Pemain diharuskan melewati jembatan tersebut tanpa terjatuh ke api.

Hard Difficulty yang ditunjukkan pada gambar 4.13 memiliki setting neraka. Pada level ini terdapat lautan lava yang memiliki banyak bebatuan tajam. Selain itu juga terdapat 2 buah bukit yang cukup datar. Terdapat sebuah jembatan yang menghubungkan kedua bukit tersebut. Di bawah jembatan tersebut terdapat lautan lava dan bebatuan. Pemain diharuskan melewati jembatan tersebut tanpa terjatuh ke lava.

C. Implementasi Sistem

Pada subbab ini dijelaskan tentang implementasi berbagai sistem pendukung permainan. Pada permainan ini terdapat timer yang digunakan untuk melihat kecepatan pemain dalam melewati jembatan, collider yang digunakan untuk memulai timer, collider yang digunakan untuk menghentikan timer dan menampilkan teks menang, dan collider yang digunakan untuk menampilkan teks Game Over ketika player terjatuh.

Tabel 4.1 Source Code Sistem Pengecekan Kondisi Tabrakan

No	Source Code
1	<code>using UnityEngine;</code>
2	<code>using System.Collections;</code>
3	
4	<code>public class SystemCollider : MonoBehaviour {</code>
5	<code>public static float timer;</code>
6	<code>public static bool menang = false;</code>
7	<code>public static bool kalah = false;</code>
8	<code>public static int state = 0;</code>
9	<code>public bool mulaiWaktu = false;</code>
10	<code>public static string waktuAkhir;</code>
11	
12	<code>void Update () {</code>
13	<code>if (mulaiWaktu == true) {</code>
14	<code>timer += Time.deltaTime;</code>
15	<code>int menit = Mathf.FloorToInt(timer / 60F);</code>
16	<code>int detik = Mathf.FloorToInt(timer - menit * 60);</code>
17	<code>string waktu = string.Format("{0:0}:{1:00}", menit, det</code>
18	<code>ik);</code>
19	<code>waktuAkhir = waktu;</code>
20	<code>Debug.Log (waktuAkhir);</code>
21	<code>}</code>
22	<code>}</code>
23	<code>void OnTriggerEnter(Collider other) {</code>
24	<code>if (state == 0 && other.tag != "kalah") {</code>
25	<code>mulaiWaktu = true;</code>
26	<code>state = 1;</code>
27	<code>}</code>
28	<code>else if (other.tag == "kalah") {</code>

```
29         mulaiWaktu = false;
30         kalah = true;
31         state = 0;
32     }
33     else if (state == 1) {
34         mulaiWaktu = false;
35         //waktuAkhir = waktuAkhir;
36         menang = true;
37         state = 0;
38     }
39 }
40 void OnGUI(){
41     if (menang) {
42         GUI.Label (new Rect (10, 10, 250, 100), "Waktu anda ada
43 lah: " + waktuAkhir);
44         mulaiWaktu = false;
45     }
46     else if (kalah) {
47         GUI.Label (new Rect (10, 10, 250, 100), "ANDA KALAH!");
48     }
49 }
50 }
```

4.2.2 Implementasi Kinect

Pada subbab ini dijelaskan tentang implementasi dari kinect ke dalam permainan yang telah dibuat sebelumnya. Pada permainan ini, kinect digunakan untuk mensinkronisasikan gerakan player di dunia nyata ke dalam avatar aktor di dalam permainan. Pada subbab ini terdapat 2 bagian yang menjelaskan tentang setup dan implementasi.

A. Setup

Kinect yang digunakan dalam permainan ini adalah Kinect V2 yang dihubungkan ke komputer dengan menggunakan Kinect Adapter for Windows. Kinect V2 membutuhkan sistem operasi Windows 8 dalam pengoperasiannya.

Berikut adalah *Recommended System Requirement* dari Kinect V2:

1. Windows 8 (64 Bit)
2. 4GB Memory
3. Intel I7 3.1 GHZ
4. Slot USB 3.0
5. Graphic Card yang mendukung Direct X 11

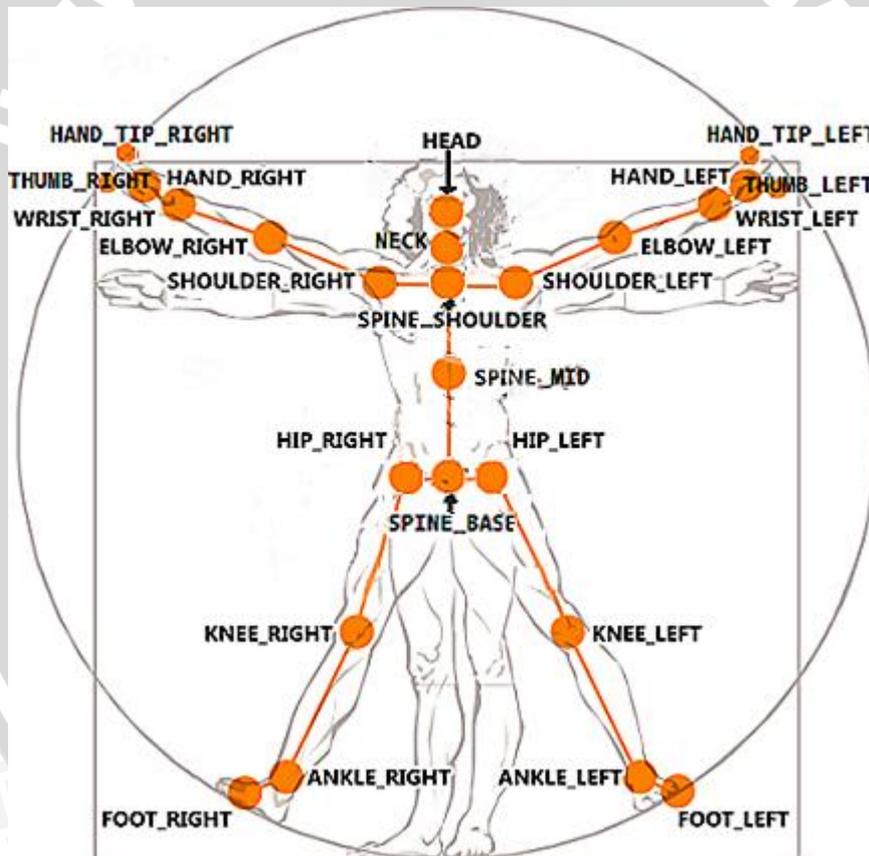
Kinect dapat langsung dihubungkan dengan komputer melalui Kinect Adapter for Windows. Setelah Kinect terhubung kita perlu menginstall Kinect SDK versi 2.0 yang dapat di unduh melalui situs Microsoft untuk menginstall driver yang diperlukan.

Setelah Kinect terhubung dengan komputer, kita perlu menghubungkan Kinect dengan Unity untuk membuat permainan yang memanfaatkan Kinect. Microsoft telah menyediakan Asset Package untuk memungkinkan Unity

memanfaatkan fitur-fitur dari Kinect yang tersedia untuk di unduh pada websitenya. Sayangnya, Asset Package yang disediakan oleh Microsoft tidak menyediakan kapabilitas untuk menghubungkan fitur bone tracking Kinect dengan model avatar aktor pada Unity. Oleh karena itu, penulis menggunakan 3rd party asset buatan Rumen Filkov yang bernama “Kinect v2 Examples with MS-SDK”. Adanya asset ini memungkinkan penulis untuk membuat menghubungkan gerakan pemain dengan avatar aktor pada unity dengan mudah.

B. Implementasi

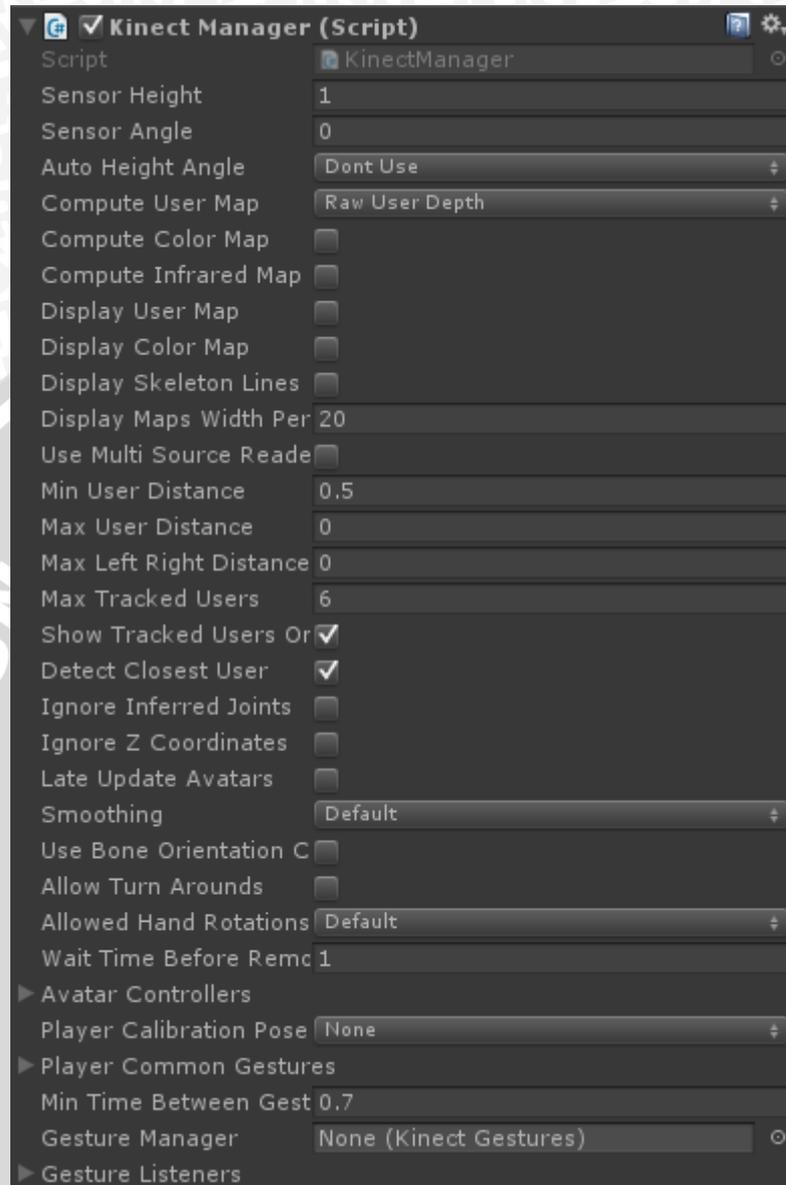
Setelah Kinect terhubung dengan komputer dan Unity, kita dapat melakukan implementasi fitur dari Kinect tersebut dengan permainan yang telah di buat. Fitur dari kinect yang dibutuhkan untuk permainan ini adalah fitur Bone Tracking yang dapat digunakan untuk mendeteksi gerakan pemain. Gambar 4.14 menunjukkan daftar Bone yang di deteksi oleh Kinect.



Gambar 4.14 Daftar Bone yang di deteksi oleh Kinect

Untuk menghubungkan hasil dari Bone Tracking kinect ke model avatar Unity, digunakan script yang telah disediakan oleh Rumen Filkov yaitu script Avatar Controller yang ditulis dengan menggunakan bahasa C#. Pertama, kita perlu memasukkan script Kinect Manager ke dalam salah satu Game Object yang ada.

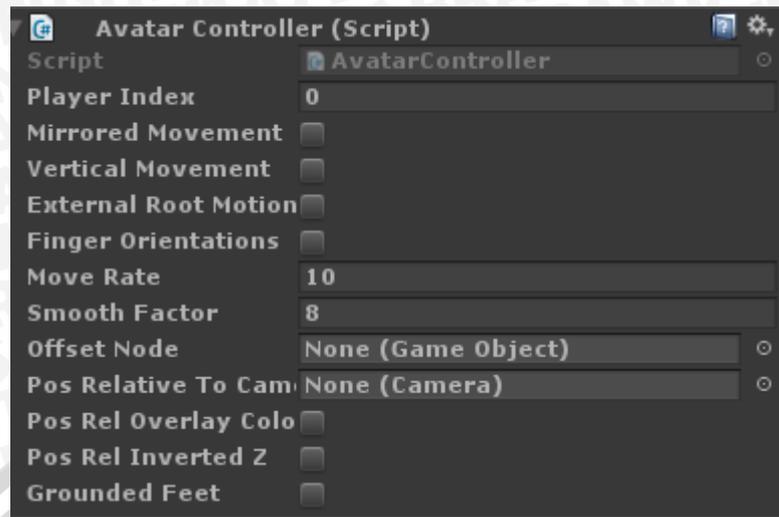
Gambar 4.15 menunjukkan contoh tampilan komponen dari script Kinect Manager.



Gambar 4.15 Contoh tampilan script Kinect Manager

Kinect Manager berguna untuk mengatur segala fitur yang berada pada Kinect. Kita dapat mengatur sudut dan ketinggian sensor, berapa banyak orang yang dapat di deteksi, dan sebagainya.

Setelah Kinect Manager berhasil di implementasikan pada scene, kita perlu menginputkan script Avatar Controller ke dalam model avatar yang telah kita siapkan sebelumnya. Gambar 4.16 menunjukkan contoh tampilan komponen dari script Avatar Controller.



Gambar 4.16 Contoh tampilan script Avatar Controller

Avatar Controller berguna untuk menghubungkan Bone yang telah di deteksi oleh Kinect dengan Rig dari model avatar aktor. Posisi avatar di deteksi melalui Bone Spine Base yang dihubungkan dengan rig bagian Hip dari model avatar aktor. Pada komponen ini kita dapat mengatur berbagai macam hal yang berhubungan dengan pergerakan avatar seperti: posisi kaki, kecepatan gerakan, gerakan vertikal, dan smoothing gerakan.

4.2.3 Implementasi Oculus Rift

Pada subbab ini dijelaskan tentang implementasi dari Oculus Rift ke dalam permainan yang telah dibuat sebelumnya. Pada permainan ini, Oculus Rift digunakan untuk menampilkan permainan secara Virtual Reality dengan sudut pandang First Person View. Pada subbab ini terdapat 2 bagian yang menjelaskan tentang setup dan implementasi.

A. Setup

Virtual Reality merupakan teknologi yang cukup baru pada saat tulisan ini dibuat dan membutuhkan spesifikasi yang cukup tinggi dalam pengoperasiannya. Head Mounted Display (HMD) yang digunakan disini adalah Oculus Rift DK2 (*Development Kit 2*). Berikut adalah Minimum System Requirement dari Oculus Rift DK2:

1. Windows: 7, 8, or 8.1
2. 2.5+ GHz processor
3. 4 GB system RAM
4. DirectX 10 or OpenGL 3 compatible video card.

Oculus Rift DK2 dapat dengan mudah di install dengan cara mengunduh software Oculus Home. Software ini akan mengunduh semua driver dan file yang dibutuhkan untuk menggunakan Oculus Rift pada komputer. Setelah proses mengunduh selesai, maka kita akan dipandu mengenai cara mengatur Oculus Rift dan Gamepad Xbox Wireless yang dibutuhkan untuk menjalankan Oculus Rift tersebut.

B. Implementasi

Oculus Rift dapat langsung digunakan pada Unity versi terbaru tanpa perlu mengunduh file tambahan, akan tetapi kita juga dapat mengunduh Oculus Utilities for Unity 5 pada website Oculus Developer untuk menggunakan beberapa fitur tambahan yang telah disediakan oleh Oculus. Berikut adalah cara mengimplementasikan Oculus Rift pada Unity 5:

Klik menu **Edit** pada toolbar Unity, kemudian klik **Project Settings** ->**Player**. Kemudian pada bagian **Other Settings**, beri tanda pada Checkbox **Virtual Reality Supported**.

Setelah Checkbox tersebut di beri tanda, maka Oculus Rift dapat langsung digunakan pada Unity. Ketika kita melakukan play testing, maka HMD Oculus Rift akan langsung menampilkan permainan kita melalui Main Camera yang ada.

Pada permainan ini, kita tidak membutuhkan fitur Positional Tracking yang terdapat pada Oculus Rift DK2 dikarenakan kamera merupakan Child Object dari Head aktor yang di sinkronisasikan dengan player melalui Kinect. Oleh karena itu, kita menggunakan Oculus Utilities yang di sediakan oleh Oculus sehingga kita dapat menggunakan Camera Rig yang telah di desain khusus untuk VR dengan beberapa opsi yang dapat kita atur. Cara menggunakan Oculus Utilities tersebut adalah dengan mengunduh Asset tersebut dari website Oculus Rift kemudian di import pada Unity dengan cara klik menu **Assets** pada toolbar, kemudian klik **Import Package** ->**Custom Package**. Kemudian pilih Oculus Utilities yang telah di unduh sebelumnya.

Fitur dari Oculus Utilities yang dipakai dalam permainan ini adalah OVRCameraRig yang merupakan kamera yang di desain khusus untuk Virtual Reality. Kamera ini akan menggantikan posisi Main Camera sebagai Child Object dari model avatar aktor bagian Head.

BAB 5 PEMBAHASAN

Pada bab ini akan dibahas proses pengujian terhadap game yang telah dibuat. Pengujian tersebut menggunakan 3 jenis pengujian yaitu: *White Box Testing*, *Black Box Testing* dan *Usability Testing*.

5.1 White Box Testing

5.1.1 Basis Path Testing

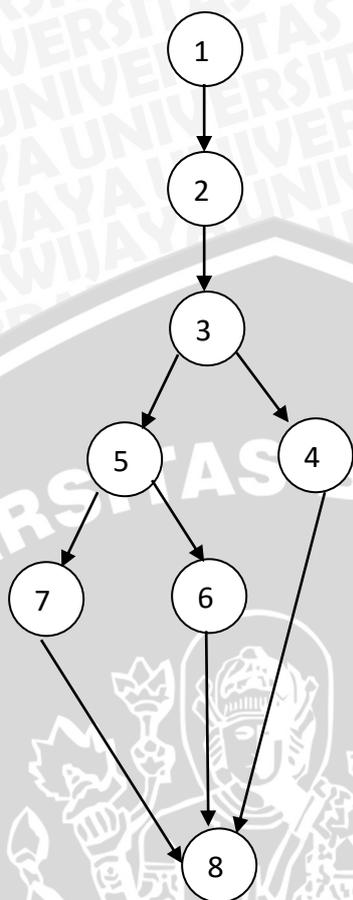
Untuk melakukan tahapan pengujian ini algoritma metode diubah menjadi flow graph, kemudian dihitung jalur eksekusi berdasarkan jumlah cyclometric complexity, jalur eksekusi yang diharapkan adalah jalur yang dapat mendeteksi tabrakan, memulai waktu, menghentikan waktu, menampilkan text kalah apabila pemain kalah, menampilkan text menang apabila pemain menang. Algoritma dijelaskan pada tabel 5.1.

Tabel 5.1 Pseudocode Pendeteksian Tabrakan

```

float timer;
bool menang = false;
bool kalah = false;
int state = 0;
bool mulaiWaktu = false;
string waktuAkhir;
cektabrakan; → 2
IF state == 0 && tagTrigger != "kalah" 3
mulaiWaktu = true; } 4
state = 1;
ELSE IF tagTrigger == "kalah" 5
mulaiWaktu = false; } 6
kalah = true;
state = 0;
ELSE
mulaiWaktu = false; } 7
menang = true;
state = 0;
SELESAI → 8
    
```





Gambar 5.1 Flow Graph algoritma pendeteksian tabrakan

Pada gambar 5.1 ditunjukkan *flow graph* dari algoritma pendeteksian tabrakan dengan jumlah *cyclometric complexity* berdasarkan persamaan $V(G) = \text{Edge} - \text{Node} + 2$

$$V(G) = 9 \text{ Edge} - 8 \text{ Node} + 2 = 3$$

Perhitungan ini menunjukkan bahwa *flow graph* dari algoritma pendeteksian tabrakan memiliki 3 jalur independen sebanyak 3 jalur, yaitu:

1. Jalur 1: 1-2-3-4-8
2. Jalur 2: 1-2-3-5-6-8
3. Jalur 3: 1-2-3-5-7-8

Kemudian dilakukan uji kasus yang dieksekusi sesuai jalur independen. Adapun uji kasus dijelaskan pada tabel 5.2.

Tabel 5.2 Uji kasus pendeteksian tabrakan

Uji Kasus	Hasil yang diharapkan	Hasil yang didapat
Pemain menginjak jembatan	Waktu berjalan, <i>state</i> berubah	Waktu berjalan, <i>state</i> berubah
Pemain berhasil	Waktu berhenti, UI	Waktu berhenti, UI

melewati jembatan	menang ditampilkan, UI menampilkan waktu akhir, UI menampilkan cara kembali ke menu utama	menang ditampilkan, UI menampilkan waktu akhir, UI menampilkan cara kembali ke menu utama
Pemain terjatuh dari jembatan	Waktu berhenti, UI kalah ditampilkan, UI menampilkan cara kembali ke menu utama	Waktu berhenti, UI kalah ditampilkan, UI menampilkan cara kembali ke menu utama

5.2 Black Box Testing

Pengujian black box dilakukan dengan cara membuat skenario uji dan hasil yang diharapkan dari masing-masing skenario tersebut. Kemudian dilakukan pengujian fungsi dari masing-masing skenario tanpa melihat *source code* yang ada.

5.2.1 Pengujian Menu Utama

Skenario uji dan hasil dari pengujian fungsional Menu Utama di tunjukkan pada Tabel 5.1.

Tabel 5.3 Pengujian fungsional dari Menu Utama

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
Menu Utama	Level selesai dimuat	Pemain dapat melihat level melalui Oculus Rift	[v] Berhasil [] Tidak Berhasil
	Pemain menggerakkan kepala	Kamera bergerak sesuai gerakan kepala pemain	
	Pemain menggerakkan kepala	Pemain dapat menggunakan gerakan kepala untuk memilih tombol yang diinginkan	
	Memilih Easy	Menampilkan area Easy	
	Memilih Medium	Menampilkan area Medium	
	Memilih Hard	Menampilkan area Hard	

5.2.2 Pengujian Easy Difficulty

Skenario uji dan hasil dari pengujian fungsional Easy Difficulty di tunjukkan pada Tabel 5.2.

Tabel 5.4 Pengujian fungsional dari Easy Difficulty

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
Easy Difficulty	Level selesai dimuat	Pemain dapat melihat level melalui Oculus Rift	[V] Berhasil [] Tidak Berhasil
	Pemain menggerakkan kepala	Kamera bergerak sesuai gerakan kepala pemain	
	Pemain menggerakkan badan	Avatar bergerak sesuai gerakan badan pemain	
	Pemain menginjak jembatan	Timer dimulai	
	Pemain jatuh dari jembatan	Timer berhenti dan menampilkan UI kalah	
	Pemain mencapai akhir jembatan	Timer berhenti dan menampilkan UI menang	
	Pemain menang	Pemain dapat menggunakan gerakan kepala untuk memilih tombol yang diinginkan	

5.2.3 Pengujian Medium Difficulty

Skenario uji dan hasil dari pengujian fungsional Medium Difficulty di tunjukkan pada Tabel 5.3.

Tabel 5.5 Pengujian fungsional dari Medium Difficulty

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
Medium Difficulty	Level selesai dimuat	Pemain dapat melihat level melalui Oculus Rift	[V] Berhasil [] Tidak Berhasil
	Pemain menggerakkan kepala	Kamera bergerak sesuai gerakan kepala pemain	
	Pemain menggerakkan badan	Avatar bergerak sesuai gerakan badan pemain	
	Pemain menginjak jembatan	Timer dimulai	

	Pemain jatuh dari jembatan	Timer berhenti dan menampilkan UI kalah
	Pemain mencapai akhir jembatan	Timer berhenti dan menampilkan UI menang
	Pemain menang	Pemain dapat menggunakan gerakan kepala untuk memilih tombol yang diinginkan

5.2.4 Pengujian Hard Difficulty

Skenario uji dan hasil dari pengujian fungsional Hard Difficulty di tunjukkan pada Tabel 5.4.

Tabel 5.6 Pengujian fungsional dari Hard Difficulty

Kelas Uji	Skenario Uji	Hasil yang diharapkan	Kesimpulan
Hard Difficulty	Level selesai dimuat	Pemain dapat melihat level melalui Oculus Rift	[V] Berhasil [] Tidak Berhasil
	Pemain menggerakkan kepala	Kamera bergerak sesuai gerakan kepala pemain	
	Pemain menggerakkan badan	Avatar bergerak sesuai gerakan badan pemain	
	Pemain menginjak jembatan	Timer dimulai	
	Pemain jatuh dari jembatan	Timer berhenti dan menampilkan UI kalah	
	Pemain mencapai akhir jembatan	Timer berhenti dan menampilkan UI menang	
	Pemain menang	Pemain dapat menggunakan gerakan kepala untuk memilih tombol yang diinginkan	

5.3 Usability Testing

Dalam pengujian ini, diambil dua puluh lima (25) responden untuk mencoba kemudian memberikan tanggapan terhadap permainan Bridge of Balance. Dari tanggapan tersebut didapatkan hasil data uji yang digunakan untuk

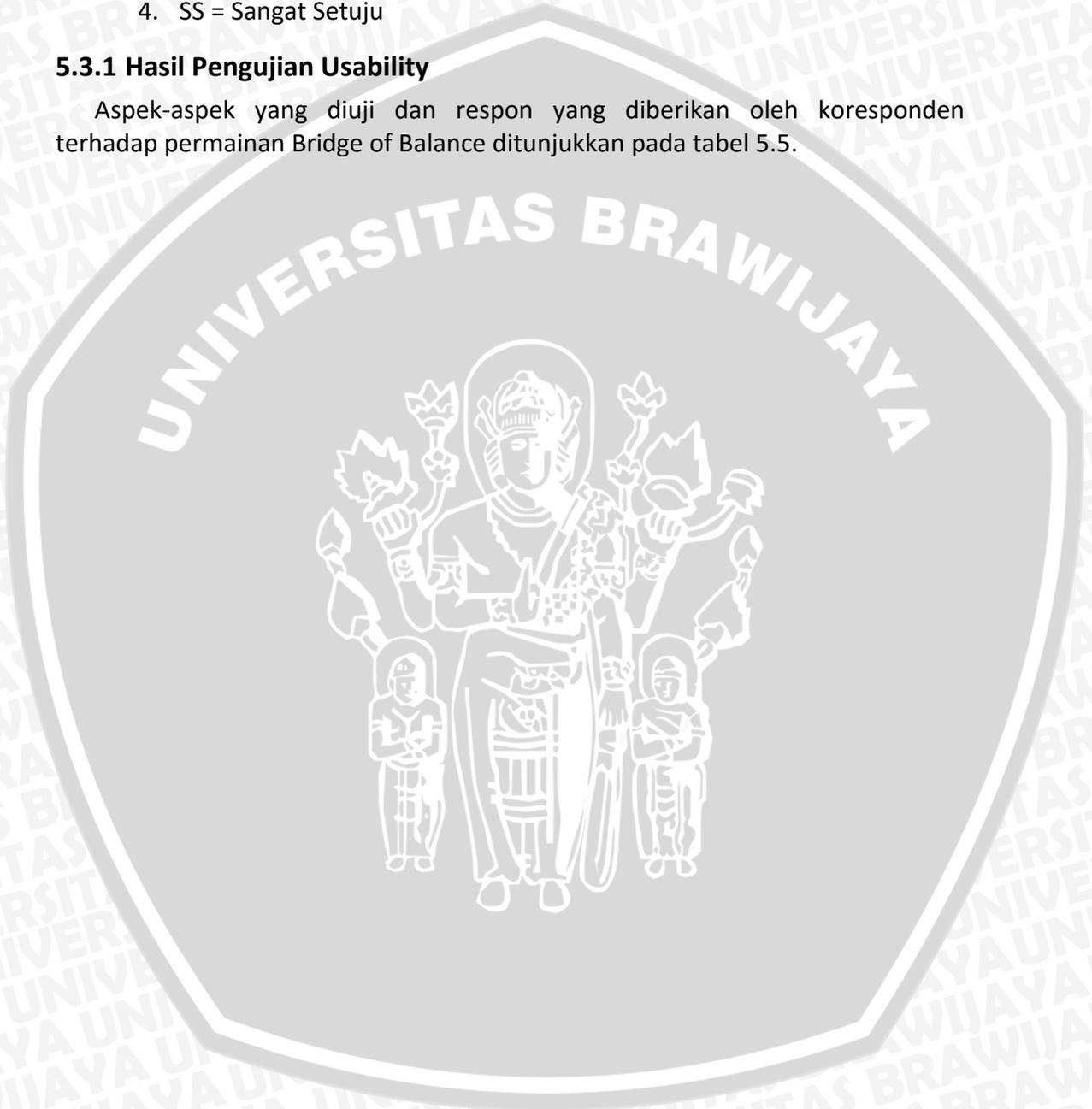
menyimpulkan tentang permainan Bridge of Balance ini. Pilihan yang diberikan pada koresponden adalah skala Likert dengan 4 pilihan tanpa ada pilihan netral.

Masing-masing pilihan tersebut adalah:

1. STS = Sangat Tidak Setuju
2. TS = Tidak Setuju
3. S = Setuju
4. SS = Sangat Setuju

5.3.1 Hasil Pengujian Usability

Aspek-aspek yang diuji dan respon yang diberikan oleh koresponden terhadap permainan Bridge of Balance ditunjukkan pada tabel 5.5.



Tabel 5.7 Hasil kuisioner Usability Testing

No	Aspek Pengujian	Nilai				Nilai Terbanyak
		STS	TS	S	SS	
1	Kejelasan petunjuk	0	2	13	10	Setuju
2	Keterbacaan warna, <i>font</i> , dan teks	0	1	12	12	Setuju dan Sangat Setuju
3	Keakuratan dan Kelengkapan informasi	0	5	13	7	Setuju
4	Kejelasan Antarmuka pada lingkungan <i>Virtual Reality</i>	0	0	13	12	Setuju
5	Kesesuaian penggunaan multimedia dengan konteks	0	2	14	9	Setuju
6	Ketepatan penggunaan multimedia dalam menyampaikan informasi	0	3	13	9	Setuju
7	Tidak berlebihan dalam penggunaan elemen multimedia	0	1	16	8	Setuju
8	Kombinasi antar elemen multimedia dapat dimengerti	0	1	17	7	Setuju
9	Oculus Rift dapat menampilkan visual dengan baik dan akurat	0	3	10	12	Sangat Setuju
10	Oculus Rift mendeteksi gerakan kepala pemain dengan akurat	0	3	6	16	Sangat Setuju
11	Kinect dapat menggerakkan karakter dengan mudah	0	2	13	10	Setuju
12	Kinect dapat menggerakkan karakter dengan akurat	0	3	12	10	Setuju
13	Bridge of Balance adalah permainan yang menyenangkan	0	1	14	10	Setuju
14	Permainan yang menggunakan Oculus Rift lebih menyenangkan dibandingkan dengan permainan yang menggunakan layar biasa	0	2	7	16	Sangat Setuju
15	Permainan yang menggunakan Kinect lebih menyenangkan dibandingkan dengan permainan yang menggunakan <i>Controller</i> Konvensional sebagai pengendali	0	2	6	17	Sangat Setuju

5.3.2 Pembahasan Pengujian Permainan Bridge of Balance

Pada subbab ini akan membahas mengenai pengujian yang telah dilakukan. Dari 15 pertanyaan yang telah disediakan, dapat di generalisasikan menjadi beberapa aspek, yaitu:

1. Aspek Visual

Pertanyaan nomor 2, 5, dan 7 dapat di kelompokkan menjadi aspek visual dikarenakan pertanyaan tersebut berhubungan dengan penggunaan grafik pada permainan Bridge of Balance.

2. Aspek Controller

Pertanyaan nomor 10 dan 11 dapat di kelompokkan menjadi aspek controller dikarenakan pertanyaan tersebut membahas mengenai penggunaan Oculus Rift dan Kinect sebagai pengendali karakter.

3. Aspek Navigasi

Pertanyaan nomor 9 dan 12 dapat di kelompokkan menjadi aspek navigasi dikarenakan pertanyaan tersebut membahas mengenai penggunaan Oculus Rift dan Kinect sebagai alat navigasi pada permainan Bridge of Balance.

4. Aspek Efisiensi

Pertanyaan nomor 1, 3, 4, 6, dan 8 dapat di kelompokkan menjadi aspek efisiensi dikarenakan pertanyaan tersebut membahas mengenai tingkat efisiensi dalam penggunaan multimedia dan penyajian petunjuk.

5. Aspek Kelayakan

Pertanyaan nomor 13 dapat di sebut sebagai aspek kelayakan dikarenakan dari pertanyaan tersebut dapat dilihat apakah permainan Bridge of Balance sudah layak untuk digunakan.

6. Aspek Keminatan

Pertanyaan nomor 14 dan 15 dapat dikelompokkan menjadi aspek keminatan dikarenakan dari pertanyaan tersebut dapat dilihat tingkat keminatan pengguna terhadap teknologi Oculus Rift dan Kinect

Disini akan dijelaskan mengenai hasil dari analisis pengujian permainan Bridge of Balance berdasarkan data yang didapatkan dengan mengacu pada aspek-aspek yang telah di uji.

1. Aspek Visual

Pada aspek visual, terdapat 3 pertanyaan yang ditanyakan, yaitu pertanyaan nomor 2, 5, dan 7. Berikut adalah pembahasan dari masing-masing pertanyaan tersebut:

Pertanyaan nomor 2 menanyakan mengenai keterbacaan warna, font, dan teks. Pada pertanyaan ini, 1 responden menjawab tidak setuju, 12 dari responden menjawab setuju dan 12 yang lain menjawab sangat setuju. Responden yang menjawab setuju dan sangat setuju sama banyaknya, sehingga

dapat ditarik kesimpulan bahwa keterbacaan warna, font, dan teks dari permainan Bridge of Balance sudah cukup baik.

Pertanyaan nomor 5 menanyakan mengenai kesesuaian penggunaan multimedia dengan konteks. Pada pertanyaan ini, 2 responden menjawab tidak setuju, 14 responden menjawab setuju, dan 9 responden menjawab sangat setuju. Jawaban paling banyak adalah setuju, sehingga dapat disimpulkan bahwa penggunaan multimedia dalam permainan Bridge of Balance sudah sesuai dengan konteks.

Pertanyaan nomor 7 menanyakan mengenai berlebihan atau tidaknya penggunaan elemen multimedia dalam permainan Bridge of Balance. 1 responden menjawab tidak setuju, 16 menjawab setuju, dan 8 menjawab sangat setuju, sehingga dapat disimpulkan bahwa permainan ini tidak berlebihan dalam menggunakan elemen multimedia.

Dari pertanyaan-pertanyaan tersebut dapat ditarik kesimpulan bahwa permainan Bridge of Balance secara visual sudah baik, meskipun ada beberapa responden yang menjawab tidak setuju, akan tetapi jumlahnya sangat sedikit, sehingga masih dapat ditoleransi.

2. Aspek Controller

Pada aspek Controller, terdapat 2 pertanyaan yang ditanyakan, yaitu pertanyaan nomor 10 dan 11. Berikut adalah pembahasan dari masing-masing pertanyaan tersebut:

Pertanyaan nomor 10 menanyakan mengenai keakuratan Oculus Rift dalam mendeteksi gerakan kepala pengguna. Pada pertanyaan ini, 3 responden menjawab tidak setuju, 6 menjawab setuju, dan 16 menjawab sangat setuju. Sehingga dapat ditarik kesimpulan bahwa Oculus Rift sangat baik dalam mendeteksi gerakan kepala player. Adanya 3 responden yang menjawab tidak setuju mungkin dikarenakan adanya kesalahan tracking dari Kinect yang mengakibatkan keakuratan pendeteksian gerakan kepala player berkurang, karena kamera pada permainan Bridge of Balance merupakan *child object* dari avatar kepala pada player avatar yang dikendalikan dengan menggunakan Kinect.

Pertanyaan nomor 11 menanyakan mengenai kemudahan penggunaan Kinect dalam menggerakkan karakter. 2 responden menjawab tidak setuju, 13 menjawab setuju, dan 10 menjawab sangat setuju sehingga dapat ditarik kesimpulan bahwa Kinect sudah cukup mudah dalam menggerakkan karakter.

Dari pertanyaan-pertanyaan tersebut dapat ditarik kesimpulan bahwa penggunaan Oculus Rift dan Kinect sebagai controller pada permainan Bridge of Balance sudah baik dan mudah.

3. Aspek Navigasi

Pada aspek navigasi, terdapat 2 pertanyaan yang ditanyakan, yaitu pertanyaan nomor 9 dan 12. Berikut adalah pembahasan dari masing-masing pertanyaan tersebut:

Pertanyaan nomor 9 menanyakan mengenai keakuratan dan baik tidaknya Oculus Rift dalam menampilkan dunia permainan. Pada pertanyaan ini, ada 3 responden yang menjawab tidak setuju, hal ini dikarenakan perangkat Oculus Rift yang digunakan dalam pengujian adalah versi DK2 yang memiliki resolusi dan jarak pandang yang kecil, sehingga terkadang gambar pada layar nampak buram. Kemudian 10 responden menjawab setuju, dan 12 menjawab sangat setuju. Sehingga dapat ditarik kesimpulan bahwa Oculus Rift dapat menampilkan dunia permainan dengan baik meskipun terkadang gambar pada layar nampak buram apabila pemasangan perangkat kurang tepat.

Pertanyaan nomor 12 menanyakan keakuratan Kinect dalam menggerakkan karakter. Pada pertanyaan ini, 3 responden menjawab tidak setuju, 12 menjawab setuju, 10 menjawab sangat setuju. Responden yang menjawab tidak setuju mengatakan bahwa Kinect kurang akurat dalam mendeteksi gerakan tangan dari player, terutama pada saat memutar tangan. Responden yang menjawab setuju dan sangat setuju menyukai keakuratan Kinect dalam mendeteksi gerakan kaki dan posisi player. Pada saat pengujian dengan salah satu responden, Kinect tidak dapat mendeteksi gerakan kaki responden dikarenakan responden tersebut menggunakan rok, akan tetapi responden tersebut memberikan nilai sangat setuju dikarenakan Kinect masih dapat mendeteksi gerakan tangan dan posisi player. Sehingga dapat disimpulkan bahwa Kinect dapat menggerakkan karakter dengan cukup akurat, selama tidak ada objek yang menghalangi Kinect dalam mendeteksi posisi bagian tubuh.

Dari pertanyaan-pertanyaan tersebut dapat ditarik kesimpulan bahwa penggunaan Oculus Rift dan Kinect sebagai alat navigasi pada permainan Bridge of Balance sudah cukup baik, akan tetapi hal ini masih dapat diperbaiki lagi dengan menggunakan versi Oculus Rift yang lebih baru sehingga gambar tidak menjadi buram dan menambahkan perangkat lain untuk mendeteksi gerakan tangan pemain dengan lebih akurat.

4. Aspek Efisiensi

Pada aspek efisiensi, terdapat 5 pertanyaan yang ditanyakan, yaitu pertanyaan nomor 1, 3, 4, 6, dan 8. Berikut adalah pembahasan dari masing-masing pertanyaan tersebut:

Pertanyaan nomor 1 menanyakan mengenai jelas atau tidaknya petunjuk yang ada pada permainan Bridge of Balance. Pada pertanyaan ini, ada 2 responden yang menjawab tidak setuju, 13 responden menjawab setuju, 10 responden menjawab sangat setuju. Sehingga dapat ditarik kesimpulan bahwa petunjuk yang berada pada permainan Bridge of Balance sudah cukup jelas.

Pertanyaan nomor 3 menanyakan mengenai keakuratan dan kelengkapan informasi yang terdapat pada antarmuka permainan Bridge of Balance. Pada pertanyaan ini, 5 responden menjawab tidak setuju, 13 responden menjawab setuju, dan 7 responden menjawab sangat setuju. Dari jawaban responden tersebut, dapat disimpulkan bahwa informasi yang terdapat pada permainan Bridge of Balance sudah cukup akurat dan lengkap, akan tetapi masih dapat diperbaiki untuk lebih membantu pemain yang masih awam.

Pertanyaan nomor 4 menanyakan apakah antarmuka yang terdapat pada permainan Bridge of Balance mudah dipahami dan digunakan pada lingkungan *Virtual Reality*. Pada pertanyaan ini, 13 responden menjawab setuju, dan 12 responden menjawab sangat setuju. Tidak ada satupun responden yang menjawab tidak setuju atau sangat tidak setuju, sehingga dapat disimpulkan bahwa antarmuka yang terdapat pada permainan Bridge of Balance mudah dipahami dan digunakan pada lingkungan *Virtual Reality*.

Pertanyaan nomor 6 menanyakan mengenai ketepatan penggunaan konten multimedia dalam menyampaikan informasi. Pada pertanyaan ini, 3 responden menjawab tidak setuju, 13 responden menjawab setuju, dan 9 responden menjawab sangat setuju, sehingga dapat disimpulkan bahwa penggunaan konten multimedia dalam menyampaikan informasi pada permainan Bridge of Balance sudah cukup baik, akan tetapi masih dapat diperbaiki untuk memperjelas informasi yang ingin disampaikan.

Pertanyaan nomor 8 menanyakan mengenai dapat dimengerti atau tidaknya kombinasi antar elemen multimedia pada permainan Bridge of Balance. Pada pertanyaan ini, 1 responden menjawab tidak setuju, 16 responden menjawab setuju, dan 8 responden menjawab sangat setuju. Sehingga dapat disimpulkan bahwa kombinasi antar elemen multimedia pada permainan Bridge of Balance sudah cukup baik.

Dari pertanyaan-pertanyaan tersebut dapat ditarik kesimpulan bahwa permainan Bridge of Balance sudah cukup efisien dalam menyampaikan informasi dan penggunaan konten multimedia, akan tetapi masih dapat diperbaiki untuk memperjelas informasi bagi pemain yang masih awam.

5. Aspek Kelayakan

Pada aspek kelayakan, hanya terdapat satu pertanyaan, yaitu pertanyaan nomor 13. Pertanyaan nomor 13 menanyakan apakah permainan Bridge of Balance menyenangkan untuk dimainkan. Pada pertanyaan ini, 1 responden menjawab tidak setuju, 14 menjawab setuju, dan 10 menjawab sangat setuju, dari jawaban responden tersebut, dapat disimpulkan bahwa permainan Bridge of Balance sudah cukup layak untuk digunakan dan cukup menyenangkan untuk dimainkan.

6. Aspek Keminatan

Pada aspek keminatan, terdapat 2 pertanyaan yang ditanyakan, yaitu pertanyaan nomor 14 dan 15. Berikut adalah pembahasan dari masing-masing pertanyaan tersebut:

Pertanyaan nomor 14 menanyakan mengenai apakah permainan yang menggunakan Oculus Rift sebagai display lebih menyenangkan dibandingkan dengan permainan yang menggunakan layar biasa sebagai display. Dari pertanyaan ini dapat disimpulkan minat responden terhadap perangkat *Virtual Reality* Oculus Rift. Pada pertanyaan ini, terdapat 2 responden yang menjawab tidak setuju, 7 responden menjawab setuju, 16 responden menjawab sangat

setuju. Dari jawaban responden tersebut, dapat disimpulkan bahwa Oculus Rift lebih menyenangkan dibandingkan permainan yang menggunakan layar biasa sebagai display dan minat terhadap Oculus Rift cukup tinggi.

Pertanyaan nomor 15 menanyakan apakah permainan yang menggunakan Kinect sebagai pengendali lebih menyenangkan dibandingkan dengan permainan yang menggunakan *Controller* konvensional sebagai pengendali. Dari pertanyaan ini dapat disimpulkan minat responden terhadap perangkat Kinect. Pada pertanyaan ini terdapat 2 responden yang menjawab tidak setuju, 6 responden menjawab setuju, 17 responden menjawab sangat setuju. Dari jawaban responden tersebut, dapat disimpulkan bahwa permainan yang menggunakan perangkat Kinect sebagai *Controller* lebih menyenangkan dibandingkan dengan permainan yang menggunakan *Controller* biasa sebagai pengendali, dan minat terhadap Kinect cukup tinggi.

Dari pertanyaan-pertanyaan tersebut, dapat ditarik kesimpulan bahwa minat responden terhadap teknologi *Virtual Reality* menggunakan Oculus Rift dan penggunaan Kinect sebagai *Controller* cukup tinggi.



BAB 6 PENUTUP

6.1 Kesimpulan

Berdasarkan hasil implementasi dan pengujian permainan Bridge of Balance dapat diambil kesimpulan yaitu:

1. Untuk merancang permainan “BOB”, diperlukan penyusunan metode perancangan yang dimulai dari menyusun arsitektur aplikasi serta representasi *virtual world* dan aktor. Kemudian level dan aktor diimplementasikan dengan jumlah *poly* yang rendah dikarenakan permainan *virtual reality* lebih berat dibandingkan permainan biasa.
2. Pengintegrasian Oculus Rift pada permainan “BOB” sangat mudah dikarenakan adanya fitur Unity yang dapat membuat perubahan sistem dan kamera secara otomatis agar permainan dapat digunakan pada sistem *virtual reality*.
3. Pengintegrasian Kinect pada permainan “BOB” memerlukan penggunaan SDK yang telah dibuat oleh Microsoft. Akan tetapi, SDK tersebut tidak mendukung fungsi untuk menghubungkan *bone* hasil deteksi Kinect kepada *avatar* aktor yang terdapat pada Unity, sehingga diperlukan *asset* tambahan yang dibuat oleh Rumen Filkov.
4. Hasil pengujian yang dilakukan pada permainan Bridge of Balance menunjukkan bahwa permainan dapat berjalan dengan baik sesuai dengan perancangan. Pada pengujian playtesting yang dilakukan pada 25 orang, 24 orang menjawab bahwa “BOB” adalah permainan yang menyenangkan, 23 orang menjawab bahwa penggunaan Oculus Rift pada permainan lebih menyenangkan dibandingkan dengan permainan yang menggunakan layar biasa, dan 23 orang menjawab bahwa penggunaan Kinect pada permainan lebih menyenangkan dibandingkan dengan permainan yang menggunakan *controller* biasa. Sehingga dapat disimpulkan bahwa minat pengguna pada permainan “BOB” yang menggunakan Kinect dan Oculus Rift sangat tinggi.

6.2 Saran

Untuk pengembangan permainan Bridge of Balance, ada beberapa hal yang dapat dikembangkan seperti penggunaan sensor tambahan untuk meningkatkan efisiensi pengenalan gerakan pemain. Dengan demikian maka mode permainan dapat dikembangkan dengan gerakan yang lebih berat seperti meloncati lubang atau meraih tali.

DAFTAR PUSTAKA

- Agarwal, Bharat Bhushan., Tayal, Sumit Prakash. 2009. Software Engineering. Laxmi Publications
- Alias. 2004. *"Polygonal Modeling Version 6"*. King Street East. Toronto. Canada.
- Binstock, Atman, 2015. Powering the Rift. Tersedia di: <<https://www.oculus.com/en-us/blog/powering-the-rift/>> [Diakses 10 Februari 2016]
- Blender Foundation. 2013. The History of Blender 3D. <<https://www.blender.org/foundation/history/>> [Diakses 10 Juli 2016]
- Blender Foundation. 2016. Features of Blender 3D. <<https://www.blender.org/features/>> [Diakses 10 Juli 2016]
- Branstetter, Gillian. 2015. The reason Google Glass failed is why Cardboard will change VR forever. <<http://www.dailydot.com/via/google-glass-cardboard-virtual-reality-devices/>> [Diakses 10 Juli 2016]
- Gamespot, 2015. Valve and HTC Reveal Vive VR Headset. Tersedia di: <<http://www.gamespot.com/articles/valve-and-htc-reveal-vive-vr-headset/1100-6425606/>> [Diakses 10 Februari 2016]
- Helgason, David (November 2, 2012). Game developers, start your Unity 3D engines (interview). Tersedia di: <<http://venturebeat.com/2012/11/02/game-developers-start-your-unity-3d-engines-interview>> [Diakses 10 Februari 2016]
- Hunicke, Robin., LeBlanc., Marc, & Zubek., Robert. MDA: A Formal Approach to Game Design and Game Research. Game Developers Conference. San Jose, California, U.S.A. 2004.
- Jackson, Brian. 2015. What is Virtual Reality? [Definition and Examples]. <<http://www.marxentlabs.com/what-is-virtual-reality-definition-and-examples/>> [Diakses 10 Juli 2016]
- Kelion, Leo. 2015. HTC reveals virtual reality headset with Valve at MWC. <<http://www.bbc.com/news/technology-31664948>> [Diakses 10 Juli 2016]
- Kickstarter, 2012. Oculus Rift: Step Into the Game. Tersedia di: <<https://www.kickstarter.com/projects/1523379957/oculus-rift-step-into-the-game>> [Diakses 10 Februari 2016]
- Knies, Rob (February 21, 2011). "Academics, Enthusiasts to Get Kinect SDK". Tersedia di: <<https://paulhartigan.net/2011/02/22/academics-enthusiasts-to-get-kinect-sdk/>> [Diakses 10 Februari 2016]
- McKay, Elspeth, 2008. The Human-Dimensions of Human-Computer Interaction: Balancing the HCI
- Microsoft. 2011. About the Kinect for Windows SDK. <<http://research.microsoft.com/en-us/um/redmond/projects/kinectsdk/about.aspx>> [Diakses 10 Juli 2016]

- Oculus VR, 2015. The Oculus Rift, Oculus Touch, and VR Games at E3. Tersedia di: < <https://www.oculus.com/en-us/blog/the-oculus-rift-oculus-touch-and-vr-games-at-e3/> > [Diakses 10 Februari 2016]
- Orland, Kyle (February 21, 2011). "News - Microsoft Announces Windows Kinect SDK For Spring Release". Gamasutra. Tersedia di: <http://www.gamasutra.com/view/news/33136/Microsoft_Announces_Windows_Kinect_SDK_For_Spring_Release.php> [Diakses 10 Februari 2016]
- Parameswari, Citra. 2008. Implementasi Lingkungan Virtual Reality pada Aplikasi Bersepeda di UI dengan Memanfaatkan Kacamata Wireless 3 Dimensi E-Dimensional untuk PC.
- Pham, Alex (June 1, 2009). "E3: Microsoft shows off gesture control technology for Xbox 360". Tersedia di: <<http://latimesblogs.latimes.com/technology/2009/06/microsofte3.html>> [Diakses 10 Februari 2016]
- Salah, Albert Ali., Gevers Theo. 2011. Computer Analysis of Human Behavior. Springer Science & Business Media.
- Salen, Katie, dan Eric Zimmerman. 2003. *Rules of Play: Game Design Fundamentals*. Massachusetts : MIT Press.
- Samsung, 2016. Spesifikasi Samsung Gear VR. Tersedia di: < http://www.samsung.com/global/microsite/gearvr/gearvr_specs.html > [Diakses 10 Februari 2016]
- Sihite, Berta., Samopa., Febriliyan & Sani., Nisfu Asrul. Pembuatan Aplikasi 3D Viewer Mobile dengan Menggunakan Teknologi Virtual Reality (Studi Kasus: Perobekan Bendera Belanda di Hotel Majapahit). Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia. 2013.
- The Verge, 2015. Valve's VR headset is called the Vive and it's made by HTC. Tersedia di: < <http://www.theverge.com/2015/3/1/8127445/htc-vive-valve-vr-headset> > [Diakses 10 Februari 2016]
- VRS group. 2016. Gear VR Virtual Reality Headset. <<http://www.vrs.org.uk/virtual-reality-gear/head-mounted-displays/gear-vr.html>> [Diakses 10 Juli 2016]

LAMPIRAN A TUTORIAL PENGGUNAAN GAME BRIDGE OF BALANCE (BOB) DENGAN MENGGUNAKAN KINECT DAN OCULUS RIFT

Untuk menjalankan game Bridge of Balance, ada beberapa hal yang perlu disiapkan, yaitu:

1. Komputer dengan spesifikasi minimum pada tabel 7.1:

Tabel 7.6.1 Tabel Spesifikasi Minimum

Perangkat Keras	
Operating System	Windows 8 64-bit
Processor	Intel(R) Core(TM) i5-4590
Memory	8192MB RAM
Harddisk	4GB Free
Graphic Card	NVIDIA GEFORCE GTX 970
Accessory	3x USB 3.0 Slot

2. Satu buah Oculus Rift dengan versi minimum DK2.
3. Satu buah Kinect V2 beserta Kinect for Windows Adapter untuk menghubungkan Kinect dengan komputer.

Setelah semua hal di atas telah dipersiapkan dan dipasang, selanjutnya buka game Bridge of Balance dan pastikan kamera Kinect berada di tempat yang lapang dan menghadap ke pemain tanpa ada halangan. Setelah itu pakai Oculus Rift dan pastikan gambar terlihat tajam dan tidak kabur. Berikut adalah petunjuk setelah berada di main menu.

1. Pada main menu terdapat 3 gambar yang menunjukkan masing-masing level yang ada beserta deskripsinya.
2. Untuk memilih level, pandang level yang diinginkan sehingga reticle berada di atas gambar tersebut, kemudian klik tombol mouse.
3. Setelah level termuat, gerakkan tubuh anda sesuai dengan petunjuk yang ada pada layar hingga layar kalibrasi menghilang dan avatar player bergerak sesuai gerakan tubuh anda.

Setelah avatar bergerak, anda dapat mulai memainkan game Bridge of Balance.