

**PENGEMBANGAN PERMAINAN EDUKASI PENGENALAN  
CHORD GITAR BERBASIS *VIRTUAL REALITY* DENGAN  
MENGUNAKAN *MYO ARMBAND***

**SKRIPSI**

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:

Aditya Yudha Agung N

NIM: 145150201111106



PROGRAM STUDI TEKNIK INFORMATIKA  
JURUSAN TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BRAWIJAYA  
MALANG  
2018

# PENGESAHAN

PENGEMBANGAN PERMAINAN EDUKASI PENGENALAN *CHORD* GITAR BERBASIS  
*VIRTUAL REALITY* DENGAN MENGGUNAKAN *MYO ARMBAND*

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:  
Aditya Yudha Agung N  
NIM: 145150201111106

Skripsi ini telah diuji dan dinyatakan lulus pada  
27 Desember 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II

Tri Afirianto, S.T, M.T

Issa Arwani, S.Kom, M.Sc

NIK: 201309 851213 1 001

NIP: 19830922 201212 1 003

Mengetahui  
Ketua Jurusan Teknik Infomatika

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

NIP: 19710518 200312 1 001



## PENGESAHAN

PENGEMBANGAN PERMAINAN EDUKASI PENGENALAN CHORD GITAR BERBASIS  
VIRTUAL REALITY DENGAN MENGGUNAKAN MYO ARMBAND

SKRIPSI

Diajukan untuk memenuhi sebagian persyaratan  
memperoleh gelar Sarjana Komputer

Disusun oleh:

Aditya Yudha Agung N  
NIM: 145150201111106

Skripsi ini telah diuji dan dinyatakan lulus pada  
27 Desember 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Dosen Pembimbing II



Tri Afirianto, S.T, M.T

Issa Arwani, S.Kom, M.Sc

NIK: 201309 851213 1 001

NIP: 19830922 201212 1 003

Mengetahui

Ketua Jurusan Teknik Informatika



W. 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 di dalam 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 Desember 2018



Aditya Yudha Agung Nugraha

NIM: 145150201111106

## KATA PENGANTAR

Segala puji syukur untuk Tuhan Yang Maha Esa karena atas rahmat dan berkat-Nya penulis dapat menyelesaikan skripsi yang berjudul “Pengembangan Permainan Edukasi Pengenalan *Chord* Gitar Berbasis Myo dengan Menggunakan *Virtual Reality*”. Skripsi ini disusun untuk memenuhi sebagian persyaratan memperoleh gelas Sarjana Komputer di Fakultas Ilmu Komputer Universitas Brawijaya Malang. Melalui kesempatan ini, penulis ingin menyampaikan rasa hormat dan terima kasih penulis yang sebesar-besarnya kepada semua pihak yang telah memberikan bantuan baik lahir maupun batin selama penulisan skripsi ini. Penulis ingin menyampaikan rasa hormat dan terima kasih kepada:

1. Allah yang Maha Esa karena atas kehendak dan nikmat-Nya laporan skripsi ini telah selesai dengan tepat waktu.
2. Wayan Firdaus Mahmudy, S.Si, M.T., Ph.D, selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya.
3. Tri Astoto Kurniawan, S.T, M.T, selaku Ketua Jurusan Teknik Informatika Universitas Brawijaya.
4. Agus Wahyu Widodo, S.T, M.Cs, selaku Ketua Program Studi Teknik Informatika Universitas Brawijaya.
5. Tri Afirianto, S.T, M.T, selaku Dosen Pembimbing I yang telah membimbing dan mendampingi penulis dengan sangat baik dan teliti selama pembuatan skripsi ini.
6. Issa Arwani, S.Kom, M.Sc, selaku Dosen Pembimbing II yang telah membimbing penulis dengan sangat baik dan teliti selama pembuatan skripsi ini.
7. Seluruh Dosen FILKOM UB yang telah membagikan ilmunya kepada penulis selama masa perkuliahan.
8. Kedua Orang Tua penulis, yang selalu mendoakan dan mendukung penulis selama pengerjaan skripsi ini.
9. Teman-teman penulis yang telah membantu penulis baik melalui dukungan, nasihat, serta saran dalam proses pengerjaan skripsi ini.
10. Seluruh kerabat yang tidak dapat disebutkan satu persatu, penulis mengucapkan terima kasih atas semangat yang telah diberikan.

Malang, 27 Desember 2018

Penulis

Yudha.dev96@gmail.com

## ABSTRAK

Di dalam mempelajari gitar, perlu diketahuinya *chord-chord* dasar yang dapat dipelajari melalui buku ataupun internet. Namun, penggunaan media cetak kurang memberikan pengalaman nyata terhadap peserta didiknya. Oleh karena itu diperlukannya sebuah media pembelajaran interaktif yang dapat memberikan pengalaman nyata kepada peserta didiknya. Dengan menggunakan *Virtual Reality* dan *Myo Armband* penelitian ini menggabungkan kedua teknologi tersebut ke dalam sebuah permainan pembelajaran interaktif agar dapat memberikan pengalaman nyata kepada peserta didiknya. Sistem dibangun dengan menggunakan *MDA Framework*, dengan menjabarkan mekanik, dinamik dan estetika yang ada pada sistem permainan. Sistem diuji dengan menggunakan *System Usability Scale (SUS)* dan *task completion rate* untuk menguji *User Experience (UX)* dari sistem tersebut, *black box testing* untuk memvalidasi fungsi di dalam sistem serta *pre-test – post-test testing* untuk menguji pengaruh permainan terhadap peningkatan pemahaman materi. Hasil penelitian menunjukkan nilai dari pengujian SUS sebesar 67,5%, yang mana angka tersebut belum memenuhi standar SUS sebesar 68%. Hasil pengujian *black box testing* mendapatkan hasil sebesar 100% valid yang mana menunjukkan bahwa fungsional sistem permainan telah berjalan dengan benar. Hasil pengujian *task completion* memberikan hasil rata-rata sebesar 91,43% dengan standar *task completion* sebesar 78% yang mana menunjukkan bahwa sistem ini telah baik dari segi tingkat keefektifannya. Dari pengujian *pre-test* dan *post-test* didapatkan hasil rata-rata untuk *pre-test* sebesar 70 dan *post-test* sebesar 90. Dari kedua nilai tersebut didapatkan selisih sebesar 20 yang menunjukkan bahwa adanya peningkatan pemahaman pemain dengan kata lain permainan ini telah berhasil sebagai media pembelajaran interaktif.

**Kata kunci:** *Virtual Reality*, *Chord* Gitar, *Myo Armband*, Permainan Edukasi, *MDA Framework*

## ABSTRACT

*In learning the guitar, it is necessary to know the basic chords that can be learned through books or the internet. However, the use of print media does not provide real experience to students. Therefore we need an interactive learning media that can provide real experiences to students. By using virtual reality and Myo Armband this research combines the two technologies into interactive learning games so they can provide real experiences to their students. The system built using the MDA Framework, describing the mechanics, dynamics and aesthetics that exist in the game system. The system is tested using the System Usability Scale (SUS) and Task Completion Rate to test User Experience (UX) of the system, Black Box testing to validate functions in the system and pre-test - post-test testing to test the influence of the game on increasing understanding of material. The results showed that the value of the SUS test was 67.5%, which number did not meet the SUS standard of 68%. The black box testing test results get a 100% valid result which indicates that the functional system of the game has been running correctly. The results of task completion testing give an average yield of 91.43% with a standard task completion of 78% which indicates that this system is good in terms of its effectiveness. The pre-test and post-test tests get an average result for the pre-test of 70 and a post-test of 90. From the two values, there are 20 equalizations which indicate that there is an increase in player understanding in other words the game has succeeded as an interactive learning media.*

**Keywords:** *Virtual Reality, Guitar Chord, Myo Armband, Educational Game, MDA Framework*

## DAFTAR ISI

PENGESAHAN .....	ii
PERNYATAAN ORISINALITAS .....	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT .....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	xi
DAFTAR TABEL.....	xiv
BAB 1 PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Tujuan .....	3
1.4 Manfaat.....	3
1.5 Batasan Masalah .....	3
1.6 Sistematika Pembahasan .....	4
BAB 2 LANDASAN KEPUSTAKAAN .....	5
2.1 <i>Game</i> .....	5
2.1.1 Pengertian <i>Game</i> .....	5
2.1.2 <i>Genre Game</i> .....	5
2.1.3 <i>Game</i> Edukasi ( <i>Edutainment</i> ).....	6
2.1.4 <i>ESRB Rating</i> .....	7
2.2 Gitar .....	8
2.2.1 <i>Chord</i> .....	8
2.2.2 <i>Chord Dasar (Major Chord)</i> .....	8
2.3 <i>Virtual Reality</i> .....	11
2.3.1 Pengertian <i>Virtual Reality</i> .....	11
2.3.2 Kegunaan <i>Virtual Reality</i> .....	12
2.3.3 <i>Head Mounted Display (HMD)</i> .....	12
2.3.4 <i>Google Cardboard</i> .....	13
2.4 Pengembangan <i>Game</i> .....	13

2.4.1	<i>MDA Framework</i> .....	13
2.4.2	<i>Usability Testing</i> .....	13
2.4.3	<i>Pre-test &amp; Post-test Testing</i> .....	16
2.4.4	<i>Black Box Testing</i> .....	16
2.5	Perangkat Pengembangan .....	17
2.5.1	Unity3D .....	17
2.5.2	Bahasa Pemrograman C# .....	17
2.5.3	<i>Myo Armband</i> .....	17
2.5.4	TrinusVR .....	18
BAB 3	METODOLOGI .....	19
3.1	Studi Literatur .....	20
3.2	Perancangan <i>Game</i> .....	20
3.3	Pengimplementasian <i>Game</i> .....	20
3.4	Pengujian <i>Game</i> .....	20
3.4.1	<i>Usability Test</i> .....	21
3.4.2	<i>Pre-test dan Post-test</i> .....	21
3.4.3	<i>Black Box Testing</i> .....	21
BAB 4	PERANCANGAN.....	22
4.1	Mekanik .....	22
4.2	Dinamika .....	37
4.3	Estetika.....	38
BAB 5	IMPLEMENTASI .....	39
5.1	Spesifikasi Sistem .....	39
5.1.1	Spesifikasi Perangkat Lunak .....	39
5.1.2	Spesifikasi Perangkat Keras.....	39
5.2	Implementasi MDA .....	40
5.2.1	Mekanik.....	40
5.2.2	Dinamik .....	55
5.2.3	Estetis .....	57
BAB 6	HASIL DAN PEMBAHASAN .....	59
6.1	Pengujian <i>Black Box</i> .....	59
6.1.1	Pengujian Kontrol Pergerakan Pemain .....	59



6.1.2 Pengujian Pemilihan Menu .....	60
6.1.3 Pengujian Berinteraksi dengan Objek .....	60
6.1.4 Pengujian <i>Achievement</i> .....	61
6.1.5 Pengujian <i>Quiz</i> .....	62
6.2 Pengujian <i>Pre-test</i> dan <i>Post-test</i> .....	62
6.3 Pengujian <i>Task Completion</i> .....	65
6.4 Pengujian <i>System Usability Scale</i> .....	67
BAB 7 PENUTUP .....	70
7.1 Kesimpulan.....	70
7.2 Saran .....	70
DAFTAR PUSTAKA.....	72



## DAFTAR GAMBAR

Gambar 2.1 Nada-nada pada <i>fretboard</i> gitar .....	8
Gambar 2.2 Skala <i>octave</i> dan pembentuk <i>chord C major</i> .....	9
Gambar 2.3 Kunci gitar <i>chord C major</i> .....	9
Gambar 2.4 Kunci gitar <i>chord D major</i> .....	9
Gambar 2.5 Kunci gitar <i>chord E major</i> .....	10
Gambar 2.6 Kunci gitar <i>chord F major</i> .....	10
Gambar 2.7 Kunci gitar <i>chord G major</i> .....	10
Gambar 2.8 Kunci gitar <i>chord A major</i> .....	10
Gambar 2.9 Kunci gitar <i>chord B major</i> .....	11
Gambar 2.10 Skala <i>Likert SUS</i> .....	15
Gambar 2.11 Macam-macam <i>gesture</i> pada <i>Myo Armband</i> .....	18
Gambar 3.1 Metodologi Penelitian .....	19
Gambar 4.1 Model karakter guru .....	23
Gambar 4.2 Logo <i>game Guitar Learn VR</i> .....	23
Gambar 4.3 Rancangan tampilan <i>Title Menu</i> .....	24
Gambar 4.4 Rancangan tampilan <i>Pause</i> .....	24
Gambar 4.5 Rancangan tampilan <i>Dialog</i> .....	25
Gambar 4.6 Rancangan tampilan <i>Achievement</i> .....	25
Gambar 4.7 Rancangan tampilan <i>Game Over</i> .....	26
Gambar 4.8 Rancangan tampilan <i>Ending Game</i> .....	26
Gambar 4.9 Rancangan tampilan <i>Post Test Score</i> .....	27
Gambar 4.10 Rancangan tampilan <i>Quiz</i> .....	27
Gambar 4.11 Rancangan tampilan <i>Tutorial</i> .....	28
Gambar 4.12 Rancangan tampilan <i>notice</i> terdapat 3 jenis (A) tampilan <i>notice 1</i> , (B) tampilan <i>notice 2</i> dan (C) tampilan <i>notice 3</i> .....	28
Gambar 4.13 Relasi ruangan pada permainan .....	30
Gambar 4.14 Gambar <i>scene</i> pada permainan yang terdiri dari (1) ruang kamar tidur, (2)ruang ujian, (3)ruang kelas dan (4)ruang <i>lobby</i> .....	31
Gambar 5.1 Karakter Guru .....	40
Gambar 5.2 Logo permainan .....	41

Gambar 5.3 Tampilan <i>Title Menu</i> .....	41
Gambar 5.4 Tampilan <i>Pause</i> .....	42
Gambar 5.5 Tampilan <i>Achievement</i> .....	42
Gambar 5.6 Tampilan Dialog dengan guru .....	43
Gambar 5.7 Tampilan <i>Game Over</i> .....	43
Gambar 5.8 Tampilan <i>Score</i> .....	44
Gambar 5.9 Tampilan <i>Ending Game</i> .....	44
Gambar 5.10 Tampilan <i>Quiz</i> .....	45
Gambar 5.11 Tampilan <i>Tutorial</i> .....	45
Gambar 5.12 Tampilan Notice yang terdiri dari (1) <i>notice 1</i> , (2) <i>notice 2</i> , dan (3) <i>notice 3</i> .....	46
Gambar 5.13 Gambar dari <i>scene</i> ruangan yang terdiri dari (1) <i>scene kamar</i> , (2) <i>scene lobby</i> , (3) <i>scene</i> ruangan kelas dan (4) <i>scene</i> ruangan ujian .....	47
Gambar 5.14 Keluaran dari kontrol pergerakan pemain ketika mundur(kiri) dan maju(kanan) .....	48
Gambar 5.15 Hasil keluaran dari <i>pseudocode</i> membuka menu <i>pause</i> di dalam permainan.....	49
Gambar 5.16 Hasil keluaran dari <i>pseudocode</i> berpindah memilih <i>button</i> dan memilih menu .....	51
Gambar 5.17 Hasil keluaran dari <i>pseudococode</i> berinteraksi dengan objek .....	53
Gambar 5.18 Hasil keluaran dari <i>pseudocode achievement</i> .....	54
Gambar 5.19 Progression dari pemain .....	55
Gambar 5.20 Pemain belajar dengan cara <i>browse</i> .....	56
Gambar 5.21 Pemain belajar dengan melihat video tutorial .....	56
Gambar 5.22 Hasil dari produktivitas pemain .....	56
Gambar 5.23 nomor (1) merupakan posisi awal pemain, (2) merupakan keadaan ketika pemain bergerak mundur ke belakang, (3) keadaan ketika pemain bergerak maju kedepan, (4) keadaan ketika pemain menengok ke kiri, (5) keadaan ketika pemain menengok ke kanan, (6) keadaan pemain ketika melihat ke atas, (7) keadaan pemain ketika melihat ke bawah. ....	57
Gambar 5.24 Beberapa <i>challenge</i> yang ada di dalam permainan.....	58
Gambar 5.25 Elemen <i>discover</i> yang ada di dalam permainan.....	58
Gambar 6.1 Tampilan <i>UI pre-test</i> .....	63
Gambar 6.2 Hasil <i>pre-test</i> dan <i>post-test</i> .....	64



Gambar 6.3 Grafik *task completion* ..... 66  
Gambar 6.4 Grafik Hasil skor SUS ..... 68



## DAFTAR TABEL

Tabel 4.3 Daftar level yang ada di dalam permainan .....	29
Tabel 4.1 Aturan yang ada di dalam permainan .....	32
Tabel 4.4 Kontrol di dalam <i>game</i> .....	35
Tabel 4.2 Daftar <i>achievement</i> di dalam permainan.....	36
Tabel 5.1 Spesifikasi dari perangkat lunak.....	39
Tabel 5.2 Spesifikasi dari perangkat keras .....	39
Tabel 5.3 <i>Pseudocode</i> kontrol pergerakan pemain .....	48
Tabel 5.4 <i>Pseudocode</i> membuka menu <i>pause</i> .....	49
Tabel 5.5 <i>Pseudocode</i> berpindah memilih <i>button</i> dan memilih menu.....	50
Tabel 5.6 <i>Pseudocode</i> berinteraksi dengan objek .....	51
Tabel 5.7 <i>Pseudocode achievement</i> .....	53
Tabel 5.8 <i>Pseudocode</i> implementasi <i>quiz</i> .....	54
Tabel 6.1 Hasil pengujian kontrol pergerakan pemain .....	59
Tabel 6.2 Hasil pengujian pemilihan menu.....	60
Tabel 6.3 Hasil pengujian berinteraksi dengan objek permainan .....	60
Tabel 6.4 Hasil pengujian <i>achievement</i> .....	61
Tabel 6.5 Hasil pengujian <i>quiz</i> .....	62
Tabel 6.6 Hasil <i>pre-test</i> dan <i>post-test</i> dari masing-masing partisipan.....	63
Tabel 6.7 <i>Task</i> yang diberikan kepada pemain .....	65
Tabel 6.8 Hasil <i>task completion</i> untuk masing-masing partisipan.....	66
Tabel 6.9 Hasil kuesioner SUS .....	67

## BAB 1 PENDAHULUAN

### 1.1 Latar Belakang

Gitar merupakan alat musik yang tersusun dari 6 *string* yang terpisah oleh masing-masing *fret* pada *neck* atau lengan gitar. Cara memainkan gitar ialah dengan memetik senar dan menekan senar pada *fret* gitar. Di dalam gitar terdapat istilah *chord*, yang mana *chord* ialah sebuah susunan harmonis yang terdiri dari 2 atau lebih note yang berbunyi secara bersamaan (Benward dan Shaker, 2003). *Chord* pada gitar dapat dibentuk dengan cara menekan secara bersamaan beberapa senar pada masing-masing *fret* tertentu sehingga membentuk sebuah nada. Untuk dapat bermain gitar perlu diketahuinya masing-masing *chord* dasar. Susunan dari banyak *chord* yang disusun secara sekuensial dinamakan *chord progression* (Wiliam 2014). Sebuah *chord progression* digunakan untuk membangun sebuah nada pada sebuah kunci dasar. Atau dengan kata lain *chord progression* merupakan dasar dari sebuah musik. Oleh sebab itu, perlu diketahuinya *chord* dasar untuk dapat memainkan sebuah lagu atau musik. Macam-macam *chord* tersebut dapat dipelajari melalui buku pembelajaran gitar. Akan tetapi penggunaan media cetak kurang memberikan pengalaman nyata terhadap peserta didiknya (Arjaka, n.d). Untuk mendapatkan hasil pembelajaran yang maksimal hendaknya sistem pembelajaran bergerak dari pengalaman langsung, menuju ke representasi pengalaman lalu menuju ke representasi simbolik (Heinich, 1982). Artinya, pembelajaran yang baik ialah pembelajaran yang memberikan pengalaman nyata kepada peserta didiknya. Sehingga diperlukannya sebuah media yang dapat memberikan pembelajaran pengalaman langsung ke peserta didiknya. Menurut Wibawanto (2017), untuk menyampaikan materi ke peserta didik secara efektif dan efisien dapat dengan menggunakan sistem pembelajaran *multimedia* interaktif (*Interactive Multimedia Learning Systems*). Pada abad ke 21 ini, kemajuan penggunaan teknologi telah mendorong penggunaan pendekatan yang lebih menarik dan efektif dalam konteks pengajaran dan pembelajaran (LEOW, 2014). Oleh karenanya diperlukan sebuah media pembelajaran interaktif yang dapat membantu proses pembelajaran *chord* gitar. Teknologi yang akhir-akhir ini yang sering digunakan ialah *Virtual Reality*.

VR atau *Virtual Reality* merupakan sebuah teknologi yang membuat seseorang seakan berada di dalam sebuah lingkungan sintesis atau buatan serta dapat berinteraksi dengan lingkungan tersebut (Milgram, 1994). VR membutuhkan sebuah media untuk membantu untuk menampilkan lingkungan virtual tersebut kepada pengguna, salah satunya ialah dengan menggunakan HMD atau *Head mounted display*. HMD berbentuk seperti kacamata atau helm yang dikenakan di kepala, yang memiliki sebuah optik di depan mata untuk menampilkan *display*. Salah satu contoh HMD ialah *google cardboard*. Dengan menggunakan *google cardboard*, pengguna akan merasa seolah berada di dalam dunia virtual. Selain itu harga untuk *google cardboard* juga relatif murah sehingga dapat dijangkau untuk masyarakat umum. Salah satu penelitian yang bertema *Virtual Reality* ialah penggunaan *Virtual Reality* untuk peningkatan kemampuan

ingatan (Anopas & Wongsawat, 2014). Pada penelitian tersebut, *Virtual Reality* digunakan sebagai peningkat aktivitas *neuron* otak serta melatih kemampuan mengingat otak. Selain itu, terdapat juga sebuah teknologi yang hampir sama dengan VR yaitu AR. Berbeda dengan VR, AR atau *augmented reality* menempatkan objek-objek virtual ke dalam dunia nyata. Akan tetapi, dalam perencanaan sebuah permainan dengan lingkungan virtual, penggunaan VR akan lebih cocok untuk diterapkan. Karena VR sendiri membawa pengguna ke dalam lingkungan virtual tersebut. Selanjutnya agar pengguna merasa benar-benar berada di dalam dunia virtual tersebut, diperlukannya juga sebuah alat atau teknologi yang dapat memungkinkan pengguna untuk berinteraksi (Sebagai alat masukan) dengan dunia virtual tersebut. Myo merupakan alat yang dapat memungkinkan hal tersebut.

Myo adalah sebuah alat yang dapat mendeteksi sinyal dari otot lengan untuk mengenali pergerakan tangan seperti *gesture fist*, *wave in*, *wave out*, *fingerspread* dan *double tap* yang dikembangkan oleh Thalmic Labs. Myo dapat memungkinkan pengguna untuk mengontrol benda elektronik secara nirkabel dengan menggunakan pergerakannya. Teknologi lain yang hampir sejenis dengan Myo ialah Kinect. Kinect juga dapat mengenali pergerakan tangan, hanya saja metode Kinect dalam pendeteksi gerakannya sedikit berbeda. Keunggulan Myo terhadap Kinect ialah Myo lebih *responsive*, karena memiliki *latency* yang lebih rendah. Myo juga lebih *portable* sementara Kinect memerlukan sebuah ruangan tertentu dalam pemakaiannya. Salah satu penelitian yang membahas mengenai Myo ialah, pengenalan gerakan lengan manusia untuk permainan *hand cricket*. Pada penelitian ini, Myo digunakan sebagai alat pendeteksi otot lengan untuk pose atau *gesture* tangan permainan *hand cricket*. Dari data otot yang didapatkan tersebut, peneliti dari paper tersebut menggunakan *Support Vector Machine* untuk mengklasifikasikan pose tadi dengan tujuan untuk mendapatkan akurasi yang tinggi dalam pengenalan pola tangan yang dibuat oleh pemain dari *hand cricket* selama permainan berlangsung.

Dengan kemampuan VR dan Myo diatas, proses pembelajaran akan lebih interaktif dan menyenangkan jika diterapkan. Oleh karena itu penelitian ini akan merancang dan mengimplementasikan sebuah permainan pembelajaran *chord* gitar dengan menggunakan VR dan Myo. Diharapkan permainan ini dapat membantu pengguna atau peserta didik agar lebih mudah dan menyenangkan dalam mempelajari *chord* dasar gitar.

## 1.2 Rumusan Masalah

Berdasarkan dari latar belakang, maka rumusan masalah yang didapatkan ialah sebagai berikut:

1. Bagaimana cara mengimplementasikan permainan edukasi pengenalan *chord* gitar berbasis *Virtual Reality* dengan menggunakan Myo *Armband*?
2. Bagaimana tingkat efektifitas, tingkat kepuasan, fungsional permainan edukasi pengenalan *chord* gitar berbasis *Virtual Reality* dengan menggunakan

*Myo Armband* sebagai media pembelajaran interaktif?

3. Bagaimana peningkatan pemahaman materi pemain setelah memainkan permainan edukasi pengenalan *chord* gitar berbasis *Virtual Reality* dengan menggunakan *Myo Armband*?

### 1.3 Tujuan

Tujuan dilakukannya penelitian ini ialah:

1. Mengetahui bagaimana cara mengimplementasikan permainan edukasi pengenalan *chord* gitar berbasis *Virtual Reality* dengan menggunakan *Myo Armband*
2. Mengetahui tingkat efektifitas, tingkat kepuasan, fungsional permainan edukasi pengenalan *chord* gitar berbasis *Virtual Reality* dengan menggunakan *Myo Armband* sebagai media pembelajaran interaktif
3. Mengetahui peningkatan pemahaman materi pemain setelah memainkan permainan edukasi pengenalan *chord* gitar berbasis *Virtual Reality* dengan menggunakan *Myo Armband*

### 1.4 Manfaat

Adapun manfaat yang diharapkan dari penelitian ini, yaitu:

1. Bagi Penulis  
Penelitian ini diharapkan dapat menambah wawasan penulis dalam membuat permainan edukasi dan kemampuan untuk mengembangkan perangkat lunak yang menggunakan *Virtual Reality* dan sensor *Myo Armband*.
2. Bagi Pemain  
Penelitian ini diharapkan dapat membantu pemain atau pelajar yang ingin belajar *chord* dasar gitar agar lebih mudah dan menarik.
3. Bagi lembaga Universitas Brawijaya  
Penelitian ini diharapkan dapat bermanfaat bagi Universitas Brawijaya sebagai tumpuhan untuk penelitian yang lebih lanjut lagi.
4. Bagi Mahasiswa  
Penelitian ini diharapkan dapat membantu mahasiswa yang sedang melakukan penelitian yang sejenis dengan tujuan sebagai referensi pengembangan lebih lanjut atau dengan studi kasus yang berbeda.

### 1.5 Batasan Masalah

Batasan masalah dari penelitian ini adalah:

1. Terbatas hanya untuk *chord* dasar (*major*) gitar.
2. Permainan berjalan pada *platform* PC.

3. Menggunakan *smartphone* sebagai *media stream* dari layar PC ke layar *smartphone*.

## 1.6 Sistematika Pembahasan

Sistematika pembahasan dibagi menjadi beberapa bab sebagai berikut:

### Bab 1 : Pendahuluan

Bab ini berisi tentang latar belakang dari kenapa penelitian ini dilakukan, identifikasi masalah, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan penelitian, dan sistematika penelitian.

### Bab 2 : Landasan Kepustakaan

Bab ini berisi tentang teori-teori yang terkait dengan penelitian beserta referensi-referensinya.

### Bab 3 : Metode Penelitian

Bab ini berisi metode yang digunakan dalam penelitian, yang mencakup kajian pustaka, perancangan *game*, implementasi *game*, dan pengujian *game*.

### Bab 4 : Perancangan

Bab ini berisi tentang perancangan, yakni penggambaran lingkungan-lingkungan dari sistem .

### Bab 5 : Implementasi

Bab ini berisi tentang implementasi yang sebelumnya telah dirancang dari sistem.

### Bab 6 : Hasil dan Pembahasan

Bab ini berisi tentang pengujian serta pembahasan dari rancangan dan implementasi penelitian dari bab sebelumnya.

### Bab 6 : Penutup

Bab ini berisi tentang kesimpulan yang diperoleh dari penelitian serta saran untuk penelitian lebih lanjut

## BAB 2 LANDASAN KEPUSTAKAAN

### 2.1 Game

#### 2.1.1 Pengertian Game

Menurut Schereiber (2009) pada bukunya, terdapat banyak pendapat mengenai definisi sebuah *game* dari beberapa peneliti, seperti:

- a. Menurut David Parlett  
Berpandangan bahwa sebuah *game* memiliki “akhir dan cara”: sebuah tujuan, sebuah hasil dan sebuah rangkaian peraturan untuk dapat menuju ke akhir.
- b. Menurut Clark C. Abt  
Berpandangan bahwa sebuah *game* ialah sebuah aktivitas yang meliputi keputusan pemain, pencarian tujuan pada sebuah konteks yang dibatasi.
- c. Menurut Chris Crawford  
Berpandangan bahwa *game* memiliki 4 properti, yaitu:
  1. *Closed formal system*, yang berarti *game* memiliki aturan-aturan. Formal dalam kasus ini berarti aturan tersebut dapat di definisikan.
  2. *Interaction*, yang berarti *game* memiliki interaksi.
  3. *Conflict*, yang berarti *game* memiliki konflik.
  4. *Safety*, yang berarti *game* menawarkan keamanan (misal ketika tertembak di dalam *game* maka pemainnya tidak akan terluka).
- d. Menurut Greg Costikyan  
Berpandangan bahwa sebuah *game* merupakan bentuk dari sebuah seni yang mana partisipan, disebut dengan pemain dapat membuat keputusan dengan tujuan untuk mengatur sumber daya melalui mata uang *game* dalam rangka mengejar tujuan.
- e. Menurut Katie Salen & Eric Zimmerman  
Mereka berpandangan bahwa *game* adalah sebuah sistem yang mana seorang pemain akan menghadapi sebuah konflik buatan, yang tersusun dari aturan-aturan, yang menghasilkan keluaran yang dapat terhitung (yang dapat terhitung disini maksudnya ialah seperti halnya sebuah konsep “menang” dan “kalah”).

#### 2.1.2 Genre Game

*Genre game* mengidentifikasi mengenai jenis *gameplay* yang ada pada sebuah *game*. Menurut Hanna (n.d) *genre game* terbagi menjadi seperti *Adventure*, *Action*, *Action-Adventure*, *Platform*, *Fighting*, *First Person Shooter*, *Strategy*, *Turn-Based*, *Role Playing*, *Massively Multiplayer Online*, *Stealth*, *Simulation*, *Racing*, *Sports*, *Rhythm*, *Puzzle*, *Board*, dan *Educational*.

### 2.1.3 Game Edukasi (*Edutainment*)

*Game* edukasi merupakan bagian dari *edutainment*. *Edutainment* Sendiri merupakan gabungan dari kata *entertainment* dan *education* (Colace & et al, 2006). Tujuan utama dari *edutainment* ialah untuk membantu pembelajaran disertai dengan *entertainment*. *Edutainment* telah digunakan sebagai *formula* klasik untuk menghasilkan permainan edukasi komputer dengan berdasarkan pada teori-teori sejak tahun 1970-an. Robert Heyman merupakan orang yang pertama kali mengusulkan ide dari *edutainment*. Robert menamai film dari *game* edukasi sebagai "*Education by Entertaining*".

Menurut Aksakal (2015) pada jurnalnya, menjelaskan bahwa terdapat beberapa pendapat oleh beberapa peneliti mengenai definisi *edutainment*. Peneliti tersebut ialah Shulman & Bowel, Druin & Solomon, Colace & co, Buckingham & Scanlon, dan Okan.

- a. Menurut Shulman & Bowel  
*Edutainment* didefinisikan sebagai pendorong pembelajaran yang menyenangkan dengan cara interaksi dan komunikasi, menelusuri dengan membuat kesadaran untuk belajar, percobaan dan kegagalan.
- b. Menurut Druin & Solomon  
*Edutainment* ialah sebuah tempat yang mencampur banyak *item* (seperti suara, animasi, video, tulisan dan gambar) dan sebuah tempat yang mana peserta didik dapat bermain dan belajar.
- c. Menurut Colace & co  
B berpendapat bahwa *edutainment* merupakan sebuah jenis dari *entertainment* yang dirancang dengan tujuan untuk mendidik dengan menyisipkan macam-macam hiburan seperti perangkat lunak multimedia, situs internet, musik, film, video dan permainan komputer.
- d. Menurut Buckingham & Scanlon  
*Edutainment* dinamai sebagai sebuah jenis campuran yang didasarkan pada penayangan dan animasi yang dibuat dengan format seperti permainan, dan materi visual.
- e. Menurut Okan  
*Edutainment* bertujuan untuk memikat perhatian peserta didik dan meregenerasi perasaan mereka .

Dalam penyampaianannya, *edutainment* dapat disampaikan dengan cara seperti:

1. Berperan dan berinteraksi
2. Drama
3. Cerita (simulasi)
4. Pengajar dan suasana kelas
5. *Edutainment* di dalam lingkungan komputer
6. *Edutainment* dengan acara TV
7. *Edutainment* dengan robot

Pada penyampaian *edutainment* dalam lingkup komputer tergolong ke dalam bagian grup dari permainan komputer. *Edutainment* pada komputer didefinisikan sebagai campuran antara permainan, cerita dan *material visual*. Tujuannya ialah agar mendapatkan perhatian dari peserta didik dan memfokuskan perhatian peserta didik terhadap materi yang disampaikan.

Beberapa keuntungan dari penggunaan permainan komputer untuk pembelajaran:

1. Lebih murah dan fleksibel  
Pembelajaran dapat yang mana saja dan kapan pun tanpa harus menghadiri sebuah kelas tertentu.
2. Meningkatkan motivasi  
Peserta didik dapat termotivasi saat melakukan pembelajaran. Sebagai contoh penggunaan sistem penghargaan atau *reward system* kepada peserta didik.
3. Membantu dalam mengatasi sebuah skenario  
Di dalam *game*, sebuah skenario dapat divisualisasikan dengan lingkungan 3D. Hal tersebut akan memberikan gambaran mengenai sebuah kejadian (skenario) kepada peserta didik, dengan begitu peserta didik akan mengerti apa yang akan terjadi ketika kejadian tersebut berlangsung ataupun apa yang seharusnya dilakukan.
4. Membantu mencapai tujuan pembelajaran  
*Game* dapat membantu untuk mencapai tujuan pembelajaran.
5. Membantu dalam mengevaluasi peserta didik  
Pada *game*, peserta didik dapat dievaluasi dengan cara memberikan sebuah kondisi yang melibatkan pemilihan-pemilihan tindakan pada saat sebuah kejadian berlangsung. Dari hasil pemilihan yang dipilih oleh peserta didik tersebut dapat dijadikan sebagai evaluasi terhadap peserta didik.

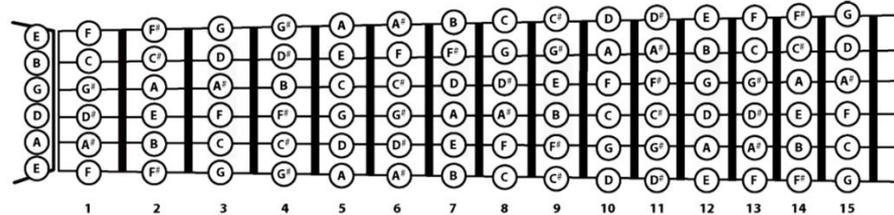
#### 2.1.4 ESRB Rating

*Entertainment Software Rating Board* (ESRB) merupakan sebuah *rating* yang berisi informasi mengenai video game dan aplikasi sehingga seorang konsumen, terutama para orang tua dapat memilih software tersebut apakah layak untuk digunakan oleh anaknya atau tidak. ESRB dibagi menjadi 3 bagian yaitu:

1. Kategori Rating, yang menjelaskan mengenai kelayakan umur. Rating tersebut terbagi menjadi everyone, everyone 10+, teen, mature 17+ dan adult only 18+.
2. Deskripsi Konten, menjelaskan mengenai konten yang ada dan yang menjadi sebab ditentukannya rating umur.
3. Elemen Interaktif, menjelaskan interaksi baik seperti fitur online dari produk, kemampuan penggunaan untuk berinteraksi satu sama lainnya, membagikan lokasi pengguna dengan pengguna lainnya.

## 2.2 Gitar

Gitar merupakan alat musik yang tersusun dari 6 *string* yang terpisah oleh masing-masing *fret* pada *neck* atau lengan gitar. Cara memainkan gitar ialah dengan memetik senar dan menekan senar pada *fret* gitar. Dengan menekan senar pada masing-masing *fret* tertentu maka gitar akan menghasilkan nada-nada tertentu juga.



**Gambar 2.1** Nada-nada pada *fretboard* gitar

Sumber: <http://www.simplifyingtheory.com/guitar-notes-piano/>

Gambar 2.1 menunjukkan masing-masing nada pada senar dengan *fret* tertentu. Pada senar 1 dengan *fret* 1 akan menghasilkan nada F, pada senar 1 dengan *fret* 2 akan menghasilkan nada F# dan seterusnya. Dapat diperhatikan juga bahwa jika *fret* sudah mencapai fret ke 12, maka nada akan mulai kembali seperti pada awal. Akan tetapi walau nada tersebut sama, nada E pada *fret* 0 dengan nada E pada *fret* 12 memiliki *pitch* yang berbeda.

*Pitch* sendiri ialah tinggi rendahnya sebuah nada dalam Hz. Nada E pada *fret* 12 memiliki *pitch* yang lebih tinggi daripada nada E pada *fret* 0. Pada dasarnya, semakin bertambahnya atau bergesernya *fret* ke arah mendalam atau ke badan gitar maka semakin tinggi juga *pitch* yang dihasilkan. Juga semakin ke bawah senar gitar maka semakin tinggi juga *pitch*-nya.

### 2.2.1 Chord

*Chord* merupakan susunan harmonis yang terdiri dari 2 atau lebih *note* musik yang berbunyi secara bersamaan (Benward & Shaker, 2003). *Chord* pada gitar dapat dibentuk dengan cara menekan secara bersamaan beberapa senar pada masing-masing *fret* tertentu sehingga membentuk sebuah nada. Terdapat beberapa jenis *chord* gitar seperti *major chord*, *minor chord*, *seventh major*, *sixth chord*, *suspend chords*, *diminished chords*, *slash chord*, dan *triad chord*.

### 2.2.2 Chord Dasar (Major Chord)



<b>C</b>	<b>D</b>	<b>E</b>	<b>F</b>	<b>G</b>	<b>A</b>	<b>B</b>	<b>C</b>
<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>

Octave

**C major =**                    **1     3     5**  
    **C     E     G**

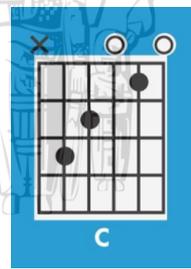
**Gambar 2.2 Skala octave dan pembentuk chord C major**

Sumber: <https://www.fender.com/articles/play/what-is-a-major-chord>

Gambar 2.2 menunjukkan bahwa C Major dapat dibentuk dengan menekan 3 senar dengan nada *root* ditambah selisi 1 tingkat nada antara 3 nada yang disusun. Nada *root* ialah nada pembentuk atau dasar. Jadi semisal ketika ingin membentuk *chord C major* maka akan dibutuhkannya not C-E-G. Contoh lainnya, jika kita ingin membentuk *chord D major* maka kita akan membutuhkan nada D-F-A dan seterusnya.

Chord-chord major yang terdapat pada gitar:

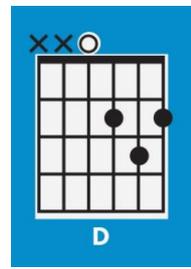
1. Chord C Major



**Gambar 2.3 Kunci gitar chord C major**

Untuk membentuk *chord C Major* nada yang akan dibutuhkan ialah C-E-G.

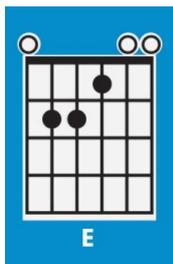
2. Chord D Major



**Gambar 2.4 Kunci gitar chord D major**

Untuk membentuk *chord D Major* nada yang akan dibutuhkan ialah D-F-A.

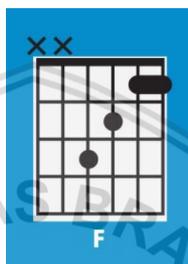
3. Chord E Major



**Gambar 2.5 Kunci gitar *chord E major***

Untuk membentuk *chord E Major* nada yang akan dibutuhkan ialah E-G-B.

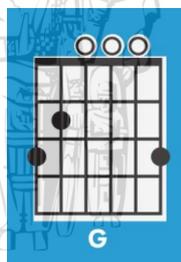
4. *Chord F Major*



**Gambar 2.6 Kunci gitar *chord F major***

Untuk membentuk *chord F Major* nada yang akan dibutuhkan ialah F-A-C.

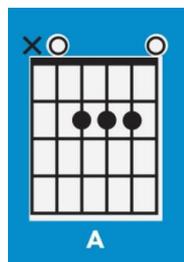
5. *Chord G Major*



**Gambar 2.7 Kunci gitar *chord G major***

Untuk membentuk *chord G Major* nada yang akan dibutuhkan ialah G-B-D.

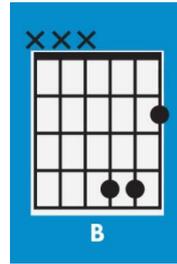
6. *Chord A Major*



**Gambar 2.8 Kunci gitar *chord A major***

Untuk membentuk *chord A Major* nada yang akan dibutuhkan ialah A-C-E.

## 7. Chord B Major



Gambar 2.9 Kunci gitar *chord B major*

Untuk membentuk *chord B Major* nada yang akan dibutuhkan ialah B-D-F.

## 2.3 Virtual Reality

### 2.3.1 Pengertian Virtual Reality

*Virtual reality* merupakan sebuah medium dengan potensi kemampuan yang besar. Kemampuan tersebut ialah kemampuan untuk memindahkan ke tempat lain (secara maya), untuk merasakan pengalaman yang mendalam dan untuk memberikan rasa jika benar-benar berada di tempat tersebut (Parisi, 2015). Dengan kata lain VR memungkinkan pengguna untuk merasakan pengalaman nyata yang mendalam akan sebuah dunia buatan.

VR meliputi koleksi dari banyak teknologi, seperti: 3D *display*, perangkat keras *motion tracking*, *input device*, *framework* perangkat lunak dan peralatan pengembangan. Contoh dari *device* VR ialah: Oculus Rift, Samsung Gear VR, dan Google *Cardboard*. Masing-masing alat tersebut memiliki level pengalaman VR yang berbeda tergantung pada harganya.

*Virtual reality* memiliki satu tujuan, yaitu meyakinkan penggunanya bahwa ia berada di tempat lain. Hal tersebut dilakukan dengan cara menipu otak manusia pada *cortex visual* dan bagian dari otak yang bekerja untuk mendeteksi *motion*. Teknologi yang berperan untuk menciptakan ilusi tersebut ialah:

1. *Stereoscopic Display*. Dikenal dengan 3D *display* atau *head mounted displays* (HMD). Jenis *display* ini menggunakan sebuah kombinasi dari banyak gambar, *realistic optical distortion*, dan lensa khusus untuk menghasilkan sebuah gambar *stereo* yang dapat membuat mata kita menafsirkan bahwa gambar tersebut memiliki kedalaman tiga dimensi.
2. *Motion Tracking Hardware*. Seperti *Gyroscopes* dan *accelerometers* digunakan pada *Virtual Reality* sebagai *hardware* untuk mendeteksi kapan tubuh kita bergerak dan kepala kita berputar, agar aplikasi dapat meng-*update* jendela *view*-nya.
3. *Input Device*. *Virtual reality* membutuhkan tipe masukan yang baru, melebihi *keyboard* dan *mouse*, termasuk *controller game* dan sensor gerak tubuh yang dapat mengenali gerakan.

4. *Desktop and Mobile Platforms*. Meliputi perangkat keras komputer, sistem operasi, perangkat lunak untuk *interface* dan alat, *framework*, dan *engine* yang dapat menjalankan aplikasi dan perangkat lunak untuk membangunnya.

### 2.3.2 Kegunaan Virtual Reality

Parisi (2015) menyebutkan bahwa kegunaan dari *Virtual Reality* meliputi:

1. *Video Games*. Dengan adanya *Virtual Reality*, pengalaman bermain pemain akan menjadi lebih mendalam. Pemain akan merasakan sensasi baru dalam bermain *game*.
2. *Virtual Worlds*. Membuat sebuah dunia virtual dengan tujuan sebagai media sosial dengan berbasis VR. Media sosial dan VR merupakan kombinasi yang kuat, beberapa perusahaan seperti High Fidelity dan AltSpace VR telah mengembangkan media sosial berbasis *Virtual Reality*.
3. *Education*. Untuk beberapa tahun terakhir visualisasi 3D menjadi alat yang bagus sebagai sarana pembelajaran interaktif. *Immersion* dari VR dapat membuat pembelajaran menjadi lebih efektif dan mudah dipahami.
4. *Productivity*. VR digunakan oleh para peneliti dan perusahaan untuk membentuk sebuah tempat kerja secara 360 untuk menyimpan informasi pribadi, kontak, proyek perusahaan dan lain-lain.
5. *Tourism*. Panorama 360 dalam VR akan sangat menarik. Mereka menawarkan pengalaman berada di suatu tempat wisata tanpa harus berada ditempat tersebut secara nyata.
6. *Architecture and Real Estate*. VR digunakan sebagai visualisasi bangunan dan kompleks yang sedang dalam tahap pembangunan. Hal tersebut dapat digunakan sebagai gambar model bangunan jadi saat pemasaran.
7. *Live Event*. Penggunaan video VR akan sangat menjanjikan untuk konser musik, berita dan acara lainnya. Beberapa musisi telah menggunakan VR untuk menyiarkan siaran langsung mereka.
8. *Web Browsing*. Mozilla telah melakukan eksperimen dengan menambahkan VR pada browser mereka. Pada eksperimen tersebut mozilla bereksperimen bagaimana caranya menjelajahi informasi pada web melalui *Virtual Reality*.
9. *Enterprise Application*. Terdapat banyak potensi dari VR untuk *enterprise*, termasuk simulasi dan pelatihan militer, diagnosa dan latihan medis, dan desain *engineering*.

### 2.3.3 Head Mounted Display (HMD)

HMD atau *Head Mounted Display* merupakan alat *display* VR yang dipasang di kepala dan memiliki *optic display* yang digunakan untuk menampilkan informasi atau gambar ke mata penggunanya. Kegunaan HMD meliputi banyak bidang

seperti *entertainment*, militer, medis dan edukasi. Google cardboard merupakan salah satu jenis dari HMD.

### 2.3.4 Google Cardboard

Google *Cardboard* VR ialah sebuah device VR yang sangat simple, murah dan mampu mengadaptasikan *smartphone* menjadi sebuah *device* VR tanpa harus membutuhkan perangkat keras baru (Parisi, 2015). Pada Mei tahun 2014, Google memperkenalkan device ini saat konferensi I/O. *Cardboard* VR tersusun dari sebuah *box* dan 2 buah lensa, yang mana kira-kira memakan biaya sekitar 2 USD. Untuk menggunakan *Cardboard* VR cukup simpel, yaitu dengan menjalankan aplikasi yang mendukung *cardboard* VR pada *smartphone* dan menempatkan *smartphone* pada *box cardboard*.

## 2.4 Pengembangan Game

### 2.4.1 MDA Framework

*MDA framework* (*Mechanics*, *Dynamics* dan *Aesthetics*) merupakan sebuah *framework* yang digunakan untuk mendesign *game* dan dikembangkan pada konferensi para pengembang *game* di San Jose tahun 2001-2004. *MDA* merupakan pendekatan formal untuk memahami permainan serta menjadi sebuah jembatan penghubung antara desain permainan dengan pengembangan, kritik permainan dan riset permainan. LeBlanc (2004) mendefinisikan *MDA* sebagai berikut:

- a. *Mechanic*, yang merupakan sinonim dari peraturan di dalam permainan. *Mechanic* mengatur bagaimana permainan berjalan. Seperti bagaimana aksi dari pemain? Apa akibat dari aksi tersebut? Bagaimana permainan berakhir? Semua itu diatur di dalam *mechanic*.
- b. *Dynamic*, menggambarkan jalannya permainan ketika aturan-aturan pada *mechanic* diterapkan.
- c. *Aesthetic*, merujuk kepada pengalaman pemain dari permainan tersebut.

### 2.4.2 Usability Testing

*Usability testing* merupakan sebuah *test non* fungsional yang digunakan untuk menghitung bagaimana kemudahan sistem dapat digunakan oleh *end user*-nya. Hal tersebut sulit untuk di evaluasi dan diukur, tetapi dapat dievaluasi berdasarkan *parameter* berikut:

1. Tingkat kemampuan yang diperlukan untuk menggunakan perangkat lunak.
2. Waktu yang diperlukan untuk terbiasa menggunakan perangkat lunak.
3. Peningkatan produktivitas dari pengguna jika ada.
4. Tugas yang diberikan kepada pengguna saat menggunakan perangkat lunak.

Pengukuran usability dapat diukur dengan menggunakan cara berikut:

### 1. Mengukur Efektivitas

Keefektifan dapat diukur dengan menghitung rata-rata keberhasilan. Rata-rata keberhasilan dihitung dengan menggunakan nilai biner 1 jika *tester* dapat menyelesaikan *task* dan bernilai 0 jika *tester* tidak dapat menyelesaikan *task* yang diberikan. Rumus dari Efektivitas ialah sebagai berikut:

$$Effectiveness = \frac{\text{banyaknya tugas yang berhasil dilakukan}}{\text{jumlah semua tugas yang dilakukan}} \times 100\%$$

Dengan rata-rata dari tingkat keberhasilan yang bagus sebesar 78% (Sauro, 2011). Rata-rata tersebut berdasarkan benchmark dari

### 2. Mengukur Efisiensi

Efisiensi diukur dengan menghitung waktu dari tugas/*task* yang diperlukan pengguna untuk menyelesaikan sebuah tugas tersebut. Efisiensi dapat diukur dengan 2 cara yaitu:

#### a. Time Based Efficiency

$$Time Based Efficiency = \frac{\sum_{j=1}^R \sum_{i=1}^N \frac{n_{ij}}{t_{ij}}}{NR}$$

Yang mana:

N = jumlah total dari *task*

R = jumlah pengguna

$n_{ij}$  = hasil dari tugas *i* oleh pengguna *j*, yang mana jika dapat menyelesaikan tugas maka bernilai 1 jika tidak maka 0

$t_{ij}$  = waktu yang dihabiskan oleh pengguna *j* untuk menyelesaikan tugas *i*, jika tugas tidak selesai maka waktu dihitung sampai pengguna berhenti atau menyerah.

#### b. Overall Relative Efficiency

$$Overall Relative Efficiency = \frac{\sum_{j=1}^R \sum_{i=1}^N n_{ij} t_{ij}}{\sum_{j=1}^R \sum_{i=1}^N t_{ij}} \times 100\%$$

Yang mana:

N = jumlah total dari *task*

R = jumlah pengguna

$n_{ij}$  = hasil dari tugas *i* oleh pengguna *j*, yang mana jika dapat menyelesaikan tugas maka bernilai 1 jika tidak maka 0

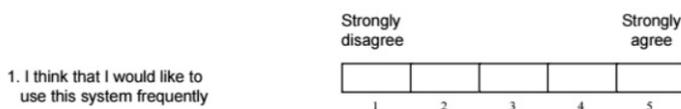
$T_{ij}$  = waktu yang dihabiskan oleh pengguna  $j$  untuk menyelesaikan tugas  $i$ , jika tugas tidak selesai maka waktu dihitung sampai pengguna berhenti atau menyerah.

### 3. Mengukur Tingkat Kepuasan Pengguna

Untuk mengukur tingkat kepuasan dari pengguna, dapat menggunakan *System Usability Scale (SUS)*. SUS merupakan skala *Likert* yang memiliki 10 pertanyaan seputar penggunaan sistem. Masing-masing pertanyaan tersebut memiliki rentang nilai dari 1 sampai 5 yang mengatakan seberapa setuju seorang *tester* atau pengguna terhadap *statement* dari pertanyaan tersebut. Nilai 1 berarti sangat tidak setuju sedangkan nilai 5 berarti sangat setuju. Terdapat 10 template *statement* SUS secara umum yaitu:

- a) Saya berfikir saya akan sering memakai sistem ini.
- b) Saya berfikir bahwa sistem ini kompleks.
- c) Saya berfikir bahwa sistem ini mudah digunakan.
- d) Saya berfikir saya memerlukan bantuan saat memakai sistem ini.
- e) Saya menemukan fungsi-fungsi di dalam sistem ini terintegrasi dengan baik.
- f) Saya berfikir bahwa terlalu banyak ketidak konsistenan di dalam aplikasi ini.
- g) Saya berfikir bahwa akan banyak orang dengan mudah mempelajari sistem ini.
- h) Saya menemukan sistem ini tidak praktis.
- i) Saya sangat percaya diri ketika memakai sistem ini.
- j) Saya perlu belajar banyak hal ketika sebelum memakai sistem ini.

Dari 10 *statement* tersebut, pengguna dimintai untuk memberi nilai pada masing-masing *statement* seperti gambar berikut:



**Gambar 2.10 Skala Likert SUS**

Untuk menghitung nilai SUS, dapat menggunakan rumus berikut:

$$\begin{aligned}
 \text{SUS Score} &= ((st_1 - 1) + (5 - st_2) + (st_3 - 1) + (5 - st_4) + (st_5 - 1) \\
 &\quad + (5 - st_6) + (st_7 - 1) + (5 - st_8) + (st_9 - 1) + (5 - st_{10})) \\
 &\quad * 2.5
 \end{aligned}$$



Yang mana:

$st_n$  = nilai *statement* ke n

Nilai SUS dari masing-masing pengguna/tester kemudian dihitung rata-ratanya. Hasil rata-rata tersebut kemudian dibandingkan dengan standar nilai SUS yaitu sebesar 68 (Thomas, 2015). Partisipan minimal yang dibutuhkan untuk mengukur usability dari sistem ialah sebanyak 5 orang (Nielsen, 2009).

### 2.4.3 Pre-test & Post-test Testing

*Pre-test* dan *post-test* sering digunakan pada penelitian *behavioral*, intinya digunakan untuk membandingkan grup dan atau menghitung perubahan terhadap perlakuan eksperimen yang telah dilakukan. Terdapat rumus untuk menghitung *pre-test* dan *post-test* yaitu:

$$D = Y_2 - Y_1$$

Yang mana:

D = *gain score*

Y1 = *pretest score*

Y2 = *posttest score*

### 2.4.4 Black Box Testing

*Blackbox testing* merupakan sebuah metodologi untuk mengetes masukan yang tersedia di dalam sebuah aplikasi dan kemungkinan hasil keluaran dari masing-masing masukan. *Black box testing* juga disebut sebagai *blind test* (Lozancic,2017), karena *black box* hanya memperhatikan masukan dan keluaran dari fungsi yang di uji apakah menghasilkan keluaran yang sudah benar atau tidak sesuai dengan ekspektasi tanpa memperhatikan alur data, algoritma, dan *state* dari aplikasi. Kebanyakan *black box testing* menggunakan *graphical user interface* (GUI) sebagai media koordinasi dengan aplikasi, sehingga untuk menguji aplikasi tersebut dapat dengan memasukan masukan pada aplikasi lalu mengamati hasil keluarannya melalui tampilan GUI aplikasi.

Khan (2012) berpendapat bahwa *black box testing* memiliki karakteristik seperti berikut:

1. Dilakukan oleh *end user*, *tester* dan *developer* (*user* yang bersedia untuk menjadi partisipan tes).
2. *Internal behaviour* dari aplikasi diabaikan.
3. Lebih mudah dan tidak memakan waktu.
4. Tidak cocok untuk pengetesan algoritma.

## 2.5 Perangkat Pengembangan

### 2.5.1 Unity3D

Unity3D merupakan sebuah *game engine* yang dapat digunakan untuk membuat *game* baik 2D dan 3D. Unity3D dapat dijalankan baik di Windows, Mac OS dan Linux. Serta hasil *game*-nya juga dapat di-*build* ke Android, Web, Windows, Linux dan Mac OS. Unity3D memiliki *store asset* yang dapat digunakan di dalam *game engine*-nya. *Asset-asset* tersebut ada yang berbayar dan ada juga yang tidak berbayar. *Asset store* tersebut dapat membantu *developer* untuk mencari kebutuhan baik *model*, *script* dan *audio*. Hal tersebut tentunya akan mempercepat proses pembuatan *game*.

Unity3D menggunakan beberapa *graphic API*, seperti Direct3D pada windows dan Xbox one, OpenGL pada Linux dan macOS dan Windows, OpenGL ES pada Android dan iOS. Unity3D juga mendukung API *graphic low level* seperti Metal pada iOS dan macOS dan Vulkan pada Android, Linux dan Windows serta Direct3D 12 pada windows dan Xbox One.

Di dalam *scripting*-nya, Unity3D menggunakan beberapa Bahasa pemrograman. Bahasa tersebut seperti C#, Javascript dan Boo. Akan tetapi, Javascript dan Boo sudah tidak di dukung lagi pada unity3D versi 2017.

### 2.5.2 Bahasa Pemrograman C#

C# merupakan pemrograman berbasis objek dengan didukung oleh Microsoft .NET *Framework*. C# dapat digunakan pada Windows (dengan menggunakan .NET *Framework*), Linux, Mac OS dan OS dengan basis UNIX lainnya (selama masih menggunakan Mono *Framework*). C# dapat digunakan untuk membangun aplikasi seperti *game* (desktop dan konsol Xbox), aplikasi *desktop*, dan *web*. Selain itu terdapat beberapa alasan mengapa menggunakan C# daripada Bahasa pemrograman lainnya, seperti:

1. Termasuk kedalam Bahasa pemrograman .NET, sehingga C# dapat berintegrasi dengan bahasa pemrograman lainnya yang tergabung ke dalam .NET *Framework* juga.
2. C# memiliki LINQ (*Language Integrated Query*), dimana LINQ tersebut ialah berupa sintak *query* yang dapat digunakan untuk setiap kumpulan data.
3. WPF atau *Windows Presentation Foundation* yang digunakan untuk membuat tampilan aplikasi yang menarik.
4. Tersedianya juga IDE gratis oleh Microsoft untuk membangun aplikasi menggunakan Bahasa pemrograman C# yaitu Microsoft *Visual Studio*.

### 2.5.3 Myo Armband

Myo *Armband* adalah sebuah alat yang dapat mendeteksi sinyal dari otot lengan guna mengenali pergerakan tangan yang dikembangkan oleh *Thalmic Labs*. Myo dapat memungkinkan pengguna untuk mengontrol benda elektronik secara nirkabel dengan menggunakan pergerakan tangan. Penggunaan Myo dapat

berupa seperti, pengatur media pemutar musik dan video (mengatur suara, *stop*, *play*, *pause* dan lain-lain), sebagai remote pengendali *drone* dan sebagai *controller* untuk *game*.

Di dalam Myo sendiri terdapat *gesture-gesture* yang digunakan untuk mewakili sebuah masukan kontrol. *Gesture-gesture* tersebut seperti *fist*, *wave left*, *wave right*, *finger spread*, dan *double tap*.



**Gambar 2.11** Macam-macam *gesture* pada *Myo Armband*

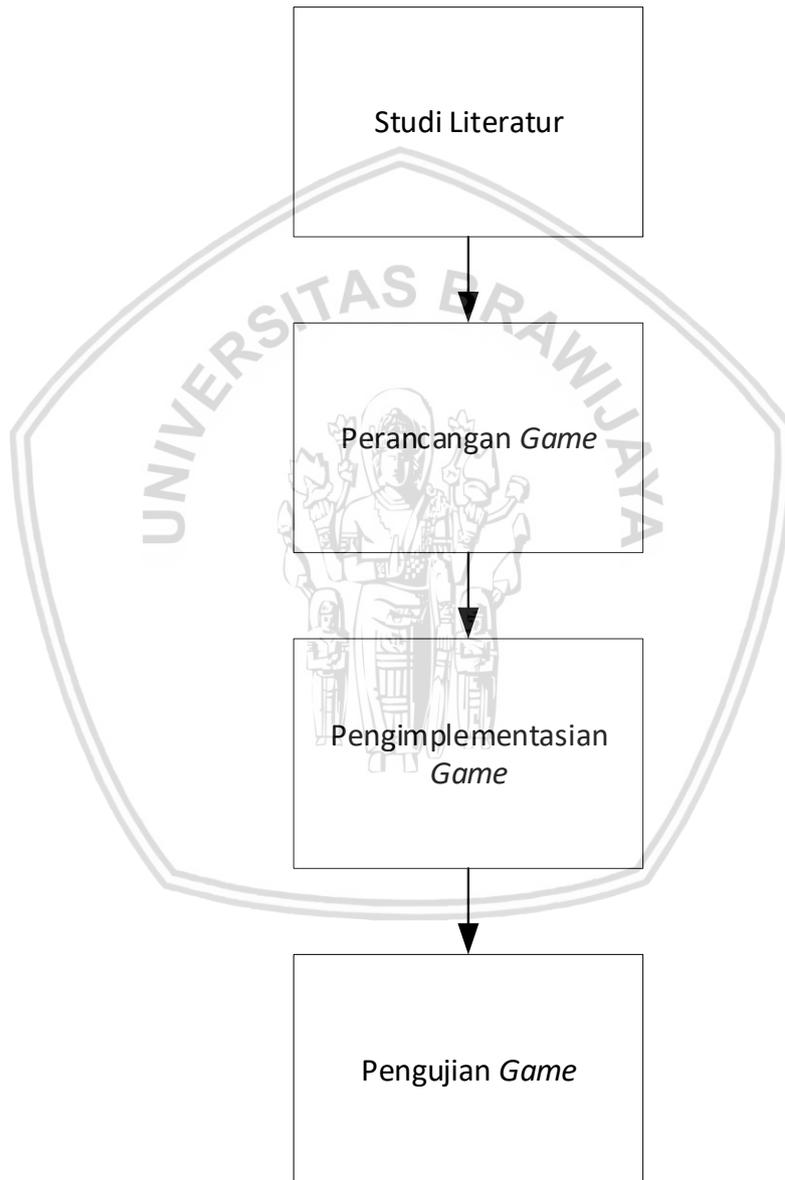
Gambar 2.11 menunjukkan bahwa terdapat 5 macam *gesture*. Pertama yaitu *gesture fist* yang berbentuk tangan menggenggam, lalu *wave left* yang berbentuk tangan berbelok ke kiri, lalu *wave right* yang berbentuk tangan berbelok ke kanan, lalu *spread* yang berbentuk tangan menyebar dan yang terakhir ialah *double tap*. *Gesture-gesture* tersebut dapat digunakan sebagai sebuah masukan untuk melakukan sebuah kontrol, sebagaimana pengaturan dari masing-masing *gesture*.

#### 2.5.4 TrinusVR

TrinusVR merupakan sebuah SDK pada *store unity3D*. TrinusVR dapat memungkinkan *developer* untuk men-stream *display window* dari permainan dan ditampilkan ke *smartphone* dalam bentuk tampilan *stereoscopic*. TrinusVR juga dapat mendeteksi *sensor gyroscope* pada *smartphone*, sehingga dapat menghasilkan *Virtual Reality* yang mana ketika pengguna menengokkan kepalanya maka kamera dari permainan juga dapat bergerak.

### BAB 3 METODOLOGI

Bab metodologi akan menjelaskan mengenai tahapan penelitian yang akan dilakukan dalam pengembangan permainan pengenalan *chord* gitar berbasis Myo dan VR. Metode penelitian akan membahas mengenai alur pelaksanaan penelitian terhadap pengembangan sistem yang akan dibuat sehingga proses penelitian menjadi terarah. Alur pelaksanaan penelitian akan terbagi menjadi beberapa tahap seperti yang ditunjukkan pada Gambar 3.1.



**Gambar 3.1 Metodologi Penelitian**

Gambar 3.1 menunjukkan bahwa alur penelitian dimulai dari studi literatur. Kemudian penelitian akan berlanjut ke tahap selanjut yaitu perancangan *game*/permainan. Setelah itu tahap selanjutnya ialah pengimplementasian *game*. dan yang terakhir ialah pengujian *game*.



### 3.1 Studi Literatur

Studi literatur merupakan pembelajaran mengenai materi-materi yang mendukung penelitian ini. Berikut merupakan dasar teori yang digunakan dalam penelitian ini:

1. *Game*
2. Gitar
3. Pengembangan *Game*
  - a. *MDA Framework*
  - b. *Usability Testing*
  - c. *Pre-test & Post-test*
  - d. *Black box Testing*
4. *Virtual Reality*
5. Perangkat Pengembangan
  - a. Unity3D
  - b. Bahasa Pemrograman C#
  - c. *Myo Armband*
  - d. TrinusVR
  - e. VR Box atau Google Cardboard

### 3.2 Perancangan *Game*

Perancangan *game* dilakukan setelah semua dasar teori telah dipelajari pada tahap studi literatur. Perancangan *game* bertujuan untuk merancang semua aspek-aspek yang berhubungan dengan *game* seperti, cara bermain, aturan permainan, *goal* permainan serta aspek pendukung seperti *visual*, *audio* dan *controller*. Ide dari *game* didapatkan dari *brainstorming*, yaitu dengan memikirkan ide-ide *game* sebanyak mungkin, lalu memilih dan menuliskan ide-ide tersebut. Setelah itu, ide-ide tersebut diterapkan kedalam *MDA Framework* untuk menentukan *mechanic*, *dynamic* dan *aesthetic* dari *game*.

### 3.3 Pengimplementasian *Game*

Pada tahap ini, *game* mulai diimplementasikan dengan menggunakan Unity3D sebagai *game engine*-nya. Permainan ini akan menggunakan Bahasa pemrograman C# sebagai bahasa *scripting*. Permainan akan berjalan pada *platform* PC, dengan menggunakan Google Cardboard sebagai media *stream display* layar PC ke layar *smartphone*.

### 3.4 Pengujian *Game*

Pada tahap pengujian permainan edukasi pengenalan *chord* gitar ini, akan dilakukan pengujian yang melibatkan beberapa pengujian untuk mengecek kelayakan/kesesuaian permainan ini apakah sudah sesuai atau tidak. Pengujian tersebut ialah *Usability Test* dan *Pre-test & Post-test*.

### 3.4.1 Usability Test

Pengujian *usability* digunakan untuk mengetahui yang mana letak kesulitan sistem oleh seorang pengguna (*User Experience*) dalam menggunakan sistem tersebut. Para penguji nantinya akan diberikan beberapa *task* atau tugas yang harus diselesaikan di dalam *game* tersebut. Dari presentase *task* yang dilakukan oleh pemain maka akan diketahui tingkat keefektifan dari permainan. Lalu penggunaan kuisisioner SUS akan dilakukan untuk mengecek tingkat kepuasan pengguna setelah menggunakan sistem nantinya.

### 3.4.2 Pre-test dan Post-test

Pengujian ini bertujuan untuk mengetahui seberapa besar pengaruh permainan edukasi pengenalan *chord* gitar terhadap penguji. Pengujian ini akan mengecek seberapa besar materi yang dipahami oleh penguji sebelum dan sesudah bermain permainan ini. Pada penelitian ini, *pre-test* digunakan untuk mengetahui pemahaman penguji mengenai pemahaman *chord* dasar gitar yang akan disampaikan di dalam permainan, sedangkan *post-test* digunakan untuk mengetahui pemahaman penguji mengenai *chord* dasar gitar setelah bermain permainan ini. Dari kedua hasil *pre-test* dan *post-test* dapat diketahui peningkatan pemahaman dari penguji. Dan dapat disimpulkan apakah permainan ini dapat meningkatkan pemahaman mengenai *chord* dasar gitar atau tidak.

### 3.4.3 Black Box Testing

*Black Box testing* akan digunakan untuk mengecek apakah fungsi-fungsi yang ada di dalam permainan sudah berjalan sesuai dengan benar atau tidak (fungsional). Pada penelitian ini, jenis *black box testing* yang akan digunakan ialah *requirement testing*. *Requirement testing* digunakan untuk mengecek fungsi utama dari sistem apakah sudah mengeluarkan keluaran/hasil yang valid atau tidak valid.

## BAB 4 PERANCANGAN

Perancangan sistem dari aplikasi ini menggunakan pendekatan MDA *framework*. Kelebihan dari MDA *framework* sendiri ialah untuk menjembatani antara mekanik, dinamis, dan estetis yang saling berhubungan dalam perancangan sebuah permainan. Sebelum menentukan mekanik, dinamik dan estetis dari permainan, peneliti melakukan *brainstorming* terlebih dahulu. Dari hasil *brainstorming* tersebut didapatkan hasil seperti berikut:

### 4.1 Mekanik

Pada Mekanik, dijelaskan mengenai aturan, komponen-komponen yang ada di dalam *game*. Komponen-komponen tersebut terdiri atas:

a. Gambaran umum dari permainan

Di dalam permainan akan terdapat sistem hari yang mana hari dimulai dari hari Senin sampai hari Minggu. Pemain hanya akan diberi waktu 1 minggu (Senin-Minggu) untuk mengikuti les pembelajaran. Permainan merupakan sebuah simulasi dari skenario yang mana pemain merupakan seorang yang ingin mengikuti les pembelajaran gitar dengan menghadiri ke sebuah tempat les pembelajaran. Di tempat tersebut (ruang kelas) pemain dapat mempelajari chord dasar pada gitar dan juga mengikuti tes (post test dan tes akhir) untuk mengukur kemampuan pemahamannya. Untuk dapat menyelesaikan permainan ini pemain harus belajar dan ikut tes akhir dalam waktu 1 minggu tersebut.

b. Konsep Art

Sebuah permainan terdiri dari aset-aset yang akan ada di dunia permainan. Bagian ini akan menjelaskan aset-aset apa saja yang akan ada nantinya.

i. Karakter

Di dalam *game* ini hanya terdapat 1 *NPC* yaitu guru. Guru bertugas sebagai *NPC* yang menyampaikan materi ke pelajar atau pemain serta bertugas sebagai pengawas ujian. Model dari karakter guru, menggunakan model gratis yang dapat diakses melalui *website* Mixamo dengan alamat:

<https://www.mixamo.com/#/?page=1&query=malc&type=Character>

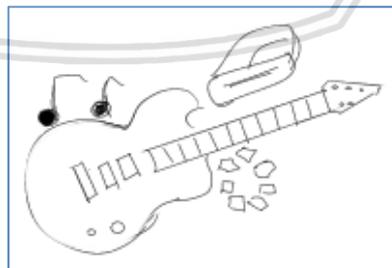


**Gambar 4.1 Model karakter guru**

Gambar 4.1 merupakan gambaran model 3D karakter Guru. Penggunaan model karakter tersebut dipilih dengan memiliki kecocokan sebagai karakter seorang guru seni serta model tersebut memiliki animasi pergerakan yang telah disediakan oleh *Mixamo* sehingga tidak perlu membuat animasi pergerakan lebih lanjut untuk model tersebut. Hal tersebut dapat mempercepat waktu pengimplementasian permainan.

ii. Logo

Logo pada *game* akan didesain menyerupai konsep *game* sendiri. Desain dari logo *game* dapat dilihat pada Gambar 4.2.



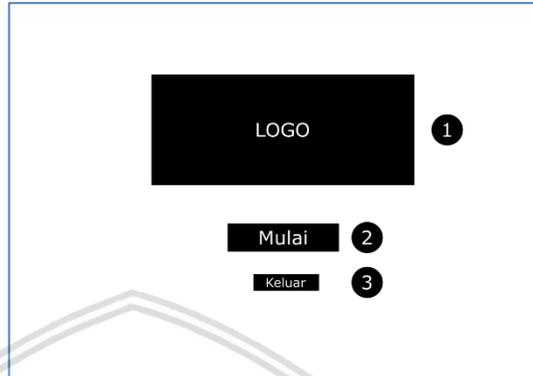
**Gambar 4.2 Logo game Gitar Learn VR**

Gambar 4.2 menunjukkan bahwa akan terdapat gambar gitar pada logo yang memiliki arti bahwa permainan ini akan mempelajari alat musik gitar. Lalu penggunaan simbol *Head Mounted Display* atau *HMD* memiliki arti bahwa *game* ini merupakan *game VR*. Serta simbol *Myo Armband* yang memiliki arti

bahwa perangkat keras Myo Armband akan digunakan di dalam memainkan *game* ini.

iii. *User Interface*

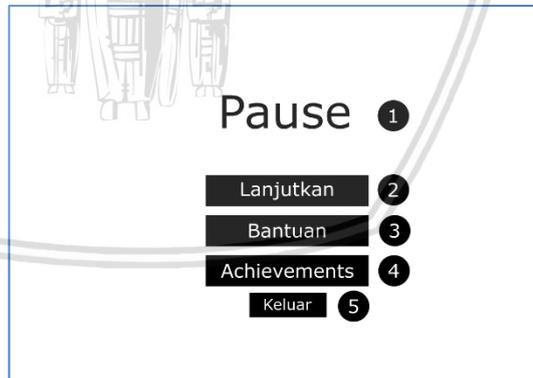
a) Tampilan *Title Menu*



**Gambar 4.3 Rancangan tampilan *Title Menu***

Gambar 4.3 merupakan tampilan *Title Menu* akan muncul pertama kali setelah permainan berjalan, pada *Title Menu* akan terdapat (1)logo, (2)tombol Mulai, dan (3)tombol Keluar. (1)*Logo* akan ditampilkan di tengah layar. Selanjutnya ketika (2)tombol Mulai ditekan, maka permainan akan berjalan ke *scene* selanjutnya. Sebaliknya, jika (3)tombol Keluar ditekan maka *game* akan keluar.

b) Tampilan *Pause*



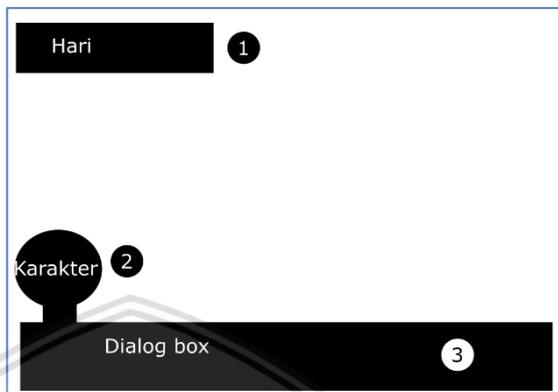
**Gambar 4.4 Rancangan tampilan *Pause***

Gambar 4.4 merupakan rancangan tampilan *Pause* yang akan muncul ketika pemain menjeda permainan. Di dalam menu *pause* terdapat elemen (1)label *Pause*, (2)tombol *Lanjutkan*, (3)tombol *Bantuan*, (4)tombol *Achievements*, dan (5)tombol *Keluar*. Yang mana tombol *Lanjutkan* digunakan untuk melanjutkan permainan, lalu tombol *Bantuan* berfungsi untuk menampilkan tampilan Tutorial, lalu tombol *Achievements*



untuk membuka daftar *achievement* yang ada, lalu tombol Keluar untuk berhenti dari *game*.

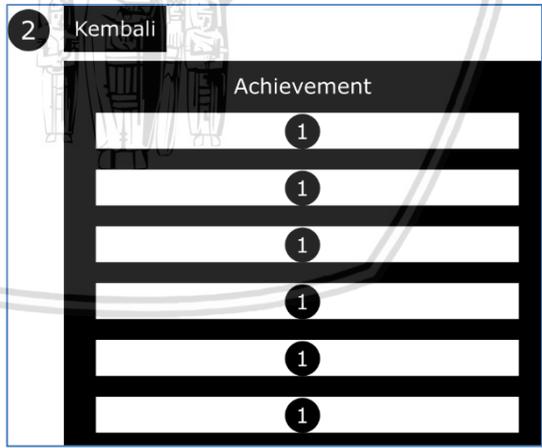
c) Tampilan Dialog



Gambar 4.5 Rancangan tampilan Dialog

Gambar 4.5 merupakan tampilan Dialog yang muncul ketika terjadi percakapan antara pemain dengan NPC. Elemen yang terdapat pada tampilan Dialog ialah (1)hari, (2)avatar karakter, dan(3)dialog *box*.

d) Tampilan *Achievement*



Gambar 4.6 Rancangan tampilan *Achievement*

Gambar 4.6 merupakan tampilan *Achievement* yang muncul ketika pemain menekan tombol *achievement* pada *menu pause*. Pada tampilan ini akan terdapat (1)daftar *Achievement* yang tersedia yang secara *optional* dapat pemain kejar atau capai. Serta (2)tombol Kembali yang berfungsi untuk kembali ke *menu pause*.



e) Tampilan *Game Over*



**Gambar 4.7 Rancangan tampilan *Game Over***

Gambar 4.7 merupakan tampilan *game over* yang keluar ketika pemain kalah di dalam *game*. Kekalahan tersebut dapat meliputi karena pemain tidak memenuhi syarat mengikuti tes akhir pada hari Minggu. Di dalam tampilan tersebut terdapat tombol seperti (2)Main Lagi yang berguna untuk memulai ulang permainan dari awal, serta (3)tombol Keluar untuk berhenti bermain. Serta label (3)GAME OVER yang mengidentifikasi permainan berakhir.

f) Tampilan *Ending Game*



**Gambar 4.8 Rancangan tampilan *Ending Game***

Gambar 4.8 merupakan tampilan dari *Ending Game* yang muncul ketika pemain telah menyelesaikan *game*. Pada tampilan tersebut akan menampilkan (1)label, (2)score total yang didapatkan pemain, serta (4)Achievement apa saja yang telah didapatkan pemain. Terdapat 2 tombol pada tampilan tersebut, yaitu (4)tombol keluar yang berfungsi untuk berhenti bermain *game* dan (5)tombol lanjut untuk melanjutkan



permainan jika pemain masih ingin bermain untuk mengejar semua *achievement* yang ada dengan *state*-nya sekarang.

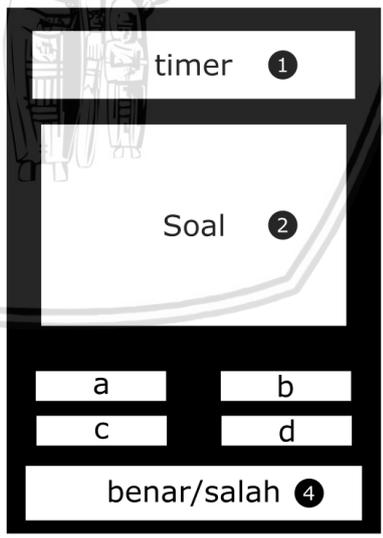
g) Tampilan *Score*



**Gambar 4.9 Rancangan tampilan *Post Test Score***

Gambar 4.9 merupakan tampilan *Post-Test Score* yang mana akan muncul jika pemain telah selesai melakukan sebuah *test*. Tampilan tersebut akan menampilkan (1)*label score*, (2)*score* dari *post test* beserta (3)*Rating* dari *score* yang didapat. Terdapat juga (4)tombol *Lanjut* yang berfungsi untuk melanjutkan permainan.

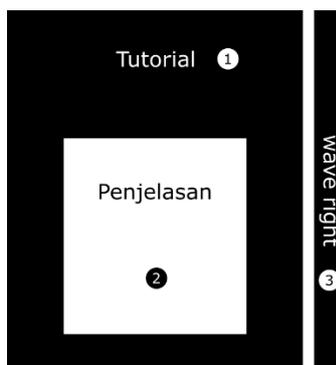
h) Tampilan *Quiz*



**Gambar 4.10 Rancangan tampilan *Quiz***

Gambar 4.10 merupakan rancangan dari tampilan *Quiz*, yang mana pada tampilan tersebut akan terdapat (1)*Timer* dari soal, (2) Soal/pertanyaan, (3) pilihan jawaban sebanyak 4 pilihan, dan (4) sebuah label yang memberikan informasi benar/salah jawaban yang dipilih oleh pemain.

i) Tampilan Tutorial



**Gambar 4.11 Rancangan tampilan Tutorial**

Gambar 4.11 merupakan rancangan dari tampilan Tutorial. Tampilan Tutorial sendiri berfungsi sebagai bantuan kepada pemain mengenai kontrol dari permainan. Pada tampilan tersebut akan terdapat (1) Label *Tutorial*, (2) Penjelasan dari *tutorial*, dan (3) *label wave right* yang mengidentifikasi.

j) Tampilan Notice



**Gambar 4.12 Rancangan tampilan *notice* terdapat 3 jenis (A) tampilan *notice* 1, (B) tampilan *notice* 2 dan (C) tampilan *notice* 3**

Gambar 4.12 menampilkan 3 jenis dari tampilan *notice*. Tampilan tersebut terdiri atas:

- a) Tampilan *notice* 1, tampilan jenis ini digunakan jika sistem ingin menampilkan pesan yang kesannya tidak terlalu *urgent* atau penting. Pada tampilan *notice* ini akan menampilkan kalimat *notice* kepada pemain di dalam kotak *notice* pada tampilan tersebut. Contoh penggunaan jenis tampilan *notice* 1 ialah ketika pemain mendapatkan sebuah *achievement*.

- b) Tampilan *notice 2*, tampilan jenis ini digunakan jika sistem ingin menampilkan pesan yang bersifat perlu adanya konfirmasi pilihan dari pemain. Pada jenis tampilan ini akan terdapat tombol (1)batal dan (2)lanjut, tombol (1)batal berfungsi untuk membatalkan aksi yang akan terjadi selanjutnya dan melanjutkan permainan, sedangkan tombol (2)lanjut akan melanjutkan ke aksi selanjutnya serta melanjutkan permainan. Ketika pesan ini ditampilkan oleh sistem, maka permainan akan ditunda untuk sementara sampai pemain memilih salah satu tombol konfirmasi. Contoh penggunaan tampilan *notice 2* ialah ketika pemain ingin memulai pembelajaran.
- c) Tampilan *notice 3*, tampilan jenis ini digunakan jika sistem ingin memberitahukan informasi bahwa aksi yang dilakukan pemain tidak dapat dilakukan atau sebuah *event* sudah selesai terjadi. Di dalam tampilan ini akan terdapat (1)label pesan konfirmasi yang berfungsi untuk menampilkan pesan dari sistem kepada pemain, kemudian (2)tombol Lanjut yang berfungsi untuk melanjutkan permainan. Ketika pesan ini ditampilkan oleh sistem, maka permainan akan ditunda untuk sementara sampai pemain memilih tombol konfirmasi. Contoh penggunaan tampilan *notice 3* ialah ketika pemain ingin memasuki kelas tetapi dia sudah mengikuti kelas untuk hari tersebut.

#### c. *Level*

Pada saat *game* berjalan, akan terdapat pembagian waktu yang terbagi menjadi hari Senin sampai dengan hari Minggu. Hari-hari tersebut merupakan hari di dalam game. Pada masing-masing hari tersebut, pemain dapat menghadiri kelas pembelajaran hanya sekali dalam sehari. Serta pembelajaran hanya akan berlangsung di dalam ruang kelas.

*Level* dari permainan ini terdapat 2 yaitu jenis tes yang akan dilalui oleh pemain. Tes tersebut seperti *post test* dan tes akhir. Penjabaran tiap-tiap level dapat dilihat pada Tabel 4.1.

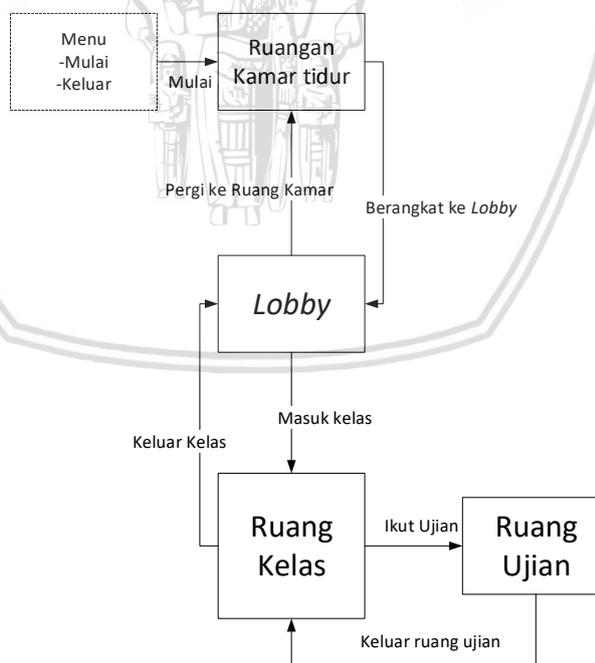
**Tabel 4.1 Daftar level yang ada di dalam permainan**

No	Nama <i>Level</i>	Syarat	Keterangan
1	<i>Post test</i>	-Mengikuti pembelajaran	<i>Post test</i> dapat diikuti oleh pemain ketika selesai mengikuti pembelajaran. Pada <i>post test</i> , tidak terdapat batasan waktu untuk menjawab soal-soal.

2	Tes akhir	<p>-Telah mengikuti 3 kali pembelajaran -dan berada di ruang ujian -berinteraksi dengan NPC guru di ruang ujian.</p>	<p><i>Post test</i> dapat diikuti oleh pemain ketika telah memenuhi syarat untuk mengikuti tes akhir. Pada tes akhir, terdapat batasan waktu untuk menjawab soal-soal.</p>
---	-----------	--	--

*Post test* akan dilakukan oleh pemain sesaat setelah menerima materi pembelajaran pada masing-masing hari kecuali hari Minggu. Dan tes akhir akan dilakukan pada hari Minggu waktu dalam *game*. Dapat dilihat juga pada Tabel 4.1, Perbedaan tingkat kesulitan dari masing-masing tes ialah adanya batasan waktu untuk menjawab soal-soal pada tes akhir, sementara pada *post tes* tidak ada batasan waktu.

Di dalam *game* sendiri terdapat 4 tempat yang dapat dikunjungi oleh pemain. Ruangan tersebut seperti: ruang kamar tidur, *lobby*, ruang kelas, dan ruang ujian.



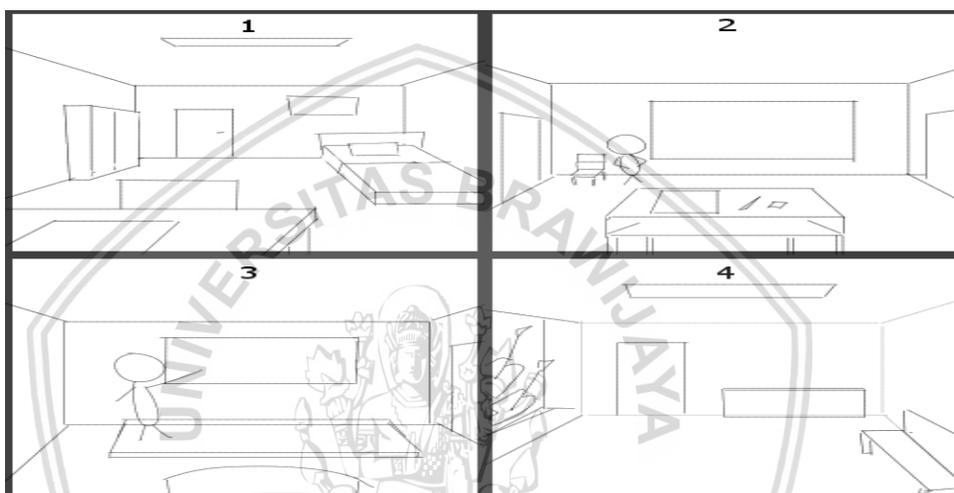
**Gambar 4.13 Relasi ruangan pada permainan**

Gambar 4.13 menjelaskan bahwa pemain bermulai dari ruang kamar tidurnya, pemain kemudian berangkat menuju ke *lobby* untuk menuju ke ruang kelas. Untuk keluar dari ruang kelas menuju *lobby*, pemain harus



mengikuti pembelajaran dan *post test* terlebih dahulu(jika sedang dalam pembelajaran). Serta untuk memasuki ruang ujian, pemain harus melalui pintu yang menuju ke ruang ujian serta sudah memenuhi syarat untuk mengikuti ujian. Pada saat berada di *lobby* dan telah mengikuti kelas, pemain hanya dapat menuju ke kamarnya saja tidak bisa masuk kembali ke dalam kelas.

Di dalam permainan akan terdapat 4 macam *scene* atau ruangan. Ruangan-ruangan tersebut ialah ruang kamar tidur, ruang ujian, ruang kelas dan ruang *lobby*.



**Gambar 4.14 Gambar *scene* pada permainan yang terdiri dari (1) ruang kamar tidur, (2)ruang ujian, (3)ruang kelas dan (4)ruang *lobby***

Adapun benda-benda yang akan terdapat di dalam masing-masing ruangan seperti yang dapat dilihat pada Gambar 4.14 seperti:

- a) Ruang kamar tidur  
Tempat tidur, meja, kursi, lemari, lampu dan pintu.
- b) Ruang ujian  
Meja ujian, papan, kursi pengawas, dan pintu.
- c) Ruang kelas  
Papan, kursi dan pintu.
- d) Ruang *lobby*  
Kursi, meja resepsionis, pintu, dan pajangan gitar.

## d. Aturan

Pada permainan akan terdapat aturan-aturan yang mengatur bagaimana pemain bermain permainan ini dan bagaimana permainan dapat berlanjut ke depan.

**Tabel 4.2 Aturan yang ada di dalam permainan**

No	Aturan	Syarat	Keterangan
1	Berpindah tempat	-Pemain harus melakukan interaksi(melakukan <i>gesture fist</i> pada Myo) di depan pintu -Objek pintu berada dalam jangkauan <i>raycast</i> yaitu sebesar 2 <i>float</i>	Untuk berpindah tempat ke tempat lain, pemain harus membentuk <i>gesture fist</i> dan berdiri di depan pintu tersebut.
2	Bergerak atau berjalan	-Pemain membentuk <i>gesture wave right</i> pada <i>myo</i> untuk maju -Pemain membentuk <i>gesture wave left</i> pada <i>myo</i> untuk mundur	Untuk bergerak baik maju ataupun mundur, pemain harus membentuk <i>gesture wave right</i> atau <i>left</i>
3	Melihat lingkungan sekitar	-Pemain menolehkan kepalanya(sudah memakai <i>cardboard</i> ) secara 360	Pemain dapat melihat keadaan atau lingkungan sekitar dengan cara menolehkan kepalanya untuk mengubah rotasi kamera di dalam <i>game</i> .
4	Berpindah hari ke hari selanjutnya(Tidur)	-Pemain membentuk <i>gesture fist</i> di depan kasur di kamar tidurnya. -kasur terkena tembakan <i>raycast</i> yang mana sejauh 2 <i>float</i> .	Pemain dapat tidur di kasurnya yang akan menyebabkan <i>game</i> berpindah ke hari selanjutnya.

5	Mengikuti pembelajaran	<ul style="list-style-type: none"> <li>-Pemain memasuki ruang kelas pada hari itu dan masih belum mengikuti pembelajaran pada hari itu.</li> <li>-Pemain berinteraksi dengan NPC guru untuk mengkonfirmasi pembelajaran.</li> </ul>	Pemain dapat memasuki ruang kelas jika masih belum mengikuti kelas pada hari itu dan berinteraksi dengan NPC guru jika ingin memulai mengikuti pembelajaran.
6	Mengikuti <i>post tes</i>	<ul style="list-style-type: none"> <li>-Pemain telah mengikuti pembelajaran</li> <li>-berinteraksi dengan NPC guru untuk konfirmasi mulai <i>post test</i></li> </ul>	Pemain harus mengikuti <i>post-tes</i> sebelum keluar dari ruang kelas
7	Mengikuti tes akhir	-Pemain telah mengikuti pembelajaran sebanyak 3 kali	Pada akhir dari <i>game</i> , pemain harus mengikuti tes akhir untuk mendapatkan nilai akhir dan <i>ending game</i> .
8	Pemain mendapatkan <i>achievement</i>	-Pemain telah melalui/memenuhi kondisi tertentu dari <i>achievement</i>	Terdapat beberapa <i>achievement</i> yang dapat diperoleh oleh pemain dengan cara memenuhi syarat dari <i>achievement-achievement</i> tersebut.

Aturan dari permainan ini ialah pemain bergerak (membentuk *gesture Myo*) mengelilingi ruangan untuk berpindah ke ruangan berikutnya dengan tujuan untuk mengikuti kelas pembelajaran serta mengikuti ujian akhir yang ada di dalam permainan. Untuk berinteraksi dengan pintu ataupun dengan objek lain di dalam permainan, pemain diharuskan untuk berada di depan pintu atau objek lain agar objek tersebut dapat terdeteksi oleh *raycast* permainan. Di dalam *game* juga terdapat beberapa kata-kata motivasi yang tersebar di dalam ruangan *game* untuk ditemukan oleh pemain. Pada tiap hari, pemain hanya dapat mengikuti pembelajaran sebanyak 1 kali saja. Dan pada akhir pembelajaran, pemain harus mengikuti sebuah *post tes* untuk

mengetahui bagaimana tingkat kepahamannya terhadap materi yang telah disampaikan. Setelah mengikuti *post tes*, pemain kembali ke *lobby* dan pemain hanya dapat kembali ke ruang kamarnya. Pemain tidak dapat kembali lagi ke ruang kelas pada hari itu. Oleh karenanya, pemain harus kembali ke ruangan kamarnya untuk berpindah hari ke hari selanjutnya. Aktivitas akan berulang-ulang terus sampai hari Minggu. Pada hari Minggu tersebut pemain akan mengikuti tes akhir yang memiliki syarat, pemain sudah mengikuti pembelajaran sebanyak 3 kali pertemuan. Jika tidak permainan akan berakhir. Dari penjabaran diatas didapatkan kondisi menang dan kalah dari permainan yaitu:

a. Kondisi menang:

Pemain telah mengikuti tes akhir dan mendapatkan nilai akhir.

b. Kondisi kalah:

Pemain tidak memenuhi kondisi untuk mengikuti tes akhir atau tidak mengikuti tes akhir.

Terdapat beberapa pertimbangan saat menetapkan peraturan-peraturan yang ada pada permainan seperti:

1. Terdapat fungsi tidur pada permainan, tujuannya ialah agar permainan ini dapat berlanjut ke hari berikutnya karena merupakan kebutuhan agar permainan dapat berlanjut ke depan. Alasan lainnya ialah untuk meningkatkan tingkat *immersion* dari permainan agar pemain dapat merasa bahwa ia benar-benar berada di dunia permainan tersebut dan terus termotivasi untuk bermain dan belajar.
2. Adanya aturan pengadaan *post-test* setelah pembelajaran, tujuannya ialah agar pemain dapat bersiap dan mengukur tingkat kemampuannya setelah mengikuti pembelajaran serta bersiap mengikuti tes akhir. dengan kosep *trial and error* tersebut pemain diharapkan dapat belajar dari kesalahan soal yang ia jawab.
3. Terdapat sistem *achievement*, tujuannya agar pemain dapat memiliki sebuah target atau tujuan in-game yang ingin ia capai. Dengan begitu pemain dapat merasakan tantangan untuk mengejar *achievement-achievement* yang ada.

f. Kontrol *Game*

Kontrol dari *game* ini ialah menggunakan Myo sebagai alat masukan pergerakan karakter dan memilih menu, serta *smartphone Android* disertai dengan *VR box* sebagai *device Virtual Reality*. Adapun daftar kontrol-kontrol yang ada di dalam *game* seperti yang dapat dilihat pada Tabel 4.3.

Tabel 4.3 Kontrol di dalam *game*

No	Kontrol	Masukan	Keterangan
1	Menggerakkan pemain	-Pemain membentuk <i>gesture wave right</i> pada Myo untuk maju  -Pemain membentuk <i>gesture wave left</i> pada Myo untuk mundur	Untuk bergerak baik maju ataupun mundur, pemain harus membentuk <i>gesture wave right</i> atau <i>left</i>
2	Membuka menu pause	-Pemain membentuk <i>gesture double tap</i> pada Myo untuk membuka menu pause	Pemain dapat membuka menu pause dengan membentuk <i>gesture double tap</i> .
3	Melakukan aksi atau berinteraksi dengan sebuah objek	-Pemain membentuk <i>gesture fist</i> pada Myo untuk berinteraksi  -Objek yang akan diinteraksi berada di dalam jangkauan <i>raycast</i> dari pemain, panjang <i>raycast</i> tersebut sebesar 2 <i>float</i>	Pemain dapat berinteraksi dengan <i>object</i> dengan cara membentuk <i>gesture fist</i> .
4	Memilih menu pilihan	-Pemain membentuk <i>gesture fist</i> pada Myo untuk memilih	Pemain dapat memilih menu pilihan dengan cara membentuk <i>gesture fist</i> .
5	Berpindah memilih menu	-Pemain membentuk <i>gesture wave right</i> pada Myo untuk memilih menu selanjutnya	Pemain dapat berpindah memilih menu dengan menggunakan <i>gesture wave right</i> dan <i>left</i> .

		-Pemain membentuk <i>gesture wave left</i> pada Myo untuk memilih menu berikutnya	
--	--	---	--

Jenis kontrol di dalam *game* ini terdapat dua yaitu *kontrol* dengan menggunakan Myo *Armband* dan kontrol dengan menggunakan *cardboard*. kontrol dengan Myo banyak digunakan untuk kontrol pergerakan karakter dan pemilihan menu. Sedangkan kontrol dengan menggunakan *cardboard* digunakan untuk merotasikan kamera yang ada di dalam *game*.

Pemain juga dapat memperoleh beberapa *achievement-achievement* di dalam permainan seperti yang dapat dilihat pada Tabel 4.4.

**Tabel 4.4 Daftar *achievement* di dalam permainan**

No	Nama <i>Achievement</i>	Syarat	Keterangan
1	<i>Fresh Learner</i>	-Mengikuti pembelajaran pertama kali	<i>Achievement</i> ini didapat oleh pemain ketika pertama kali selesai mengikuti pembelajaran
2	<i>First Challenge</i>	-Mengikuti <i>post tes</i> pertama kali	<i>Achievement</i> ini didapat oleh pemain ketika pertama kali selesai mengikuti <i>post tes</i>
3	<i>Ready for Test</i>	-Mengikuti 3 kali pembelajaran	<i>Achievement</i> ini didapat oleh pemain ketika telah mengikuti pembelajaran sebanyak 3 kali
4	<i>Hard learner</i>	-Mengikuti 5 kali pembelajaran	<i>Achievement</i> ini didapat oleh pemain ketika pemain telah mengikuti total 5 kali pembelajaran
5	<i>Good Score</i>	-Memiliki <i>score tes</i> akhir diatas 70 <i>point</i>	<i>Achievement</i> ini didapat oleh pemain

			ketika dia berhasil mendapatkan <i>score</i> tes akhir diatas 70 <i>point</i> .
6	<i>Excellent Score</i>	-Memiliki <i>score</i> tes akhir diatas 90 <i>point</i>	<i>Achievement</i> ini didapat oleh pemain ketika dia berhasil mendapatkan <i>score</i> tes akhir diatas 90 <i>point</i> .
7	<i>Perfect Score</i>	-Memiliki <i>score</i> tes akhir sebesar 100 <i>point</i>	<i>Achievement</i> ini didapat oleh pemain ketika dia berhasil mendapatkan <i>score</i> tes akhir sebesar 100 <i>point</i> .

e. *Challenge*

Terdapat beberapa *challenge* yang ada pada *game* ini seperti:

- a. Waktu untuk mempelajari materi yang terbatas. Karena pemain hanya diberi waktu dari Senin sampai Sabtu untuk menguasai materi sebelum mengikuti tes akhir yang dilaksanakan pada hari Minggu.
- b. *Post-test* setiap kali pemain selesai menerima materi. Pada setiap akhir pembelajaran, pemain nantinya harus melaksanakan *post-test* terkait materi yang sudah diterima.
- c. Terdapat batas waktu untuk mengerjakan tes akhir.

## 4.2 Dinamika

*Game* dinamik merupakan perilaku mekanik pada saat permainan sedang berjalan ketika pemain memberikan masukan di dalam permainan. Beberapa dinamis yang akan ada di dalam permainan seperti *progression*, *creativity* dan *productivity*.

### 1. *Progression*

Pada awal permainan pemain akan memulai dari 0. Pemain tidak akan memiliki *achievement* sama sekali pada awal permainan. Sering dengan pemain mengikuti kegiatan belajar di kelas, pemain akan mendapatkan *achievement* sehingga memungkinkan pada akhir permainan pemain dapat mendapatkan semua *achievement* yang ada.

## 2. *Creativity*

Pemain akan dituntut untuk belajar secara kreatif (bagaimana cara pemain akan mempelajari materi). Di dalam permainan, pemain dapat mengikuti kelas pembelajaran. Setelah sesi pembelajaran tersebut akan terdapat sebuah mode kreatif untuk mempelajari materi dengan dua cara, yaitu menampilkan kembali *chord* pembelajaran atau melihat video tutorial yang ada di dalam permainan.

## 3. *Productivity*

Produktifitas dapat dilihat dari *achievement-achievement* yang telah didapatkan oleh pemain. Hal tersebut menandakan bagaimana pemain telah menggunakan waktunya untuk melakukan atau mengejar *achievement* yang ada di dalam permainan. Selain itu produktifitas dapat dilihat juga dari hasil *post-test* atau kemampuan pemain dalam menyerap materi.

### 4.3 Estetika

Estetika merupakan *feedback* perasaan yang dirasakan oleh pemain ketika bermain *game* ini. Terdapat beberapa poin dari estetika yang ada pada *game* ini yaitu:

#### 1. *Sensation*

Pemain akan merasakan sensasi petualangan yang didapatkan dari penjelajahan lingkungan di dalam *game*, memasuki ruangan-ruangan virtual yang dapat dilihat secara 360 derajat.

#### 2. *Challenge*

Pemain akan merasakan tantangan, keseruhan dan rasa akan kebanggaan yang didapatkan dari penyelesaian sebuah *achievement* di dalam *game*, menyelesaikan rintangan seperti *post tes* dan tes akhir di dalam *game*.

#### 3. *4Discover*

Pemain akan merasakan kesenangan dan motivasi yang didapatkan dari penemuan kata-kata motivasi di dalam *game*.

## BAB 5 IMPLEMENTASI

Pada dasarnya, implementasi merupakan pembangunan sebuah sistem dengan acuan perancangan yang telah dilakukan sebelumnya. Pada tahap sebelumnya, permainan ini dirancang dengan menggunakan metode MDA. Perancangan tersebut akan diimplementasikan dan kemudian akan dibahas pada bab implementasi.

### 5.1 Spesifikasi Sistem

Spesifikasi sistem digunakan untuk mengetahui *environment* pengembangan permainan yang dilakukan di dalam penelitian ini. Spesifikasi tersebut terbagi menjadi 2 macam yaitu spesifikasi untuk perangkat lunak dan spesifikasi untuk perangkat keras.

#### 5.1.1 Spesifikasi Perangkat Lunak

Spesifikasi perangkat lunak merupakan paparan dari perangkat lunak yang digunakan di dalam membangun permainan ini. Macam-macam perangkat lunak yang digunakan untuk membangun permainan ini dapat dilihat pada Tabel 5.1.

**Tabel 5.1 Spesifikasi dari perangkat lunak**

Nama Komponen	Spesifikasi
Sistem operasi PC	Windows 10 Pro Build 16299
Sistem operasi <i>smartphone</i>	Android 7.1.2 N2G47H
Alat	<ol style="list-style-type: none"> <li>1. Unity3D 5.6.5f1, digunakan sebagai <i>game engine</i> dari permainan.</li> <li>2. Microsoft Visual Studio 2017, digunakan sebagai IDE menulis kode pemograman.</li> </ol>

#### 5.1.2 Spesifikasi Perangkat Keras

Spesifikasi perangkat keras merupakan paparan dari perangkat keras yang digunakan selama mengembangkan permainan ini. Macam-macam dari perangkat keras yang digunakan dapat dilihat pada Tabel 5.2.

**Tabel 5.2 Spesifikasi dari perangkat keras**

Nama Komponen	Spesifikasi
Laptop	Intel I7 6700HQ 8 CPU 2.6 Ghz 8GB RAM, Nvidia GTX960m 2GB VRAM
Myo	Myo <i>Armband</i> Thalmic Lab
<i>Smartphone</i>	Redmi 4A 2GB RAM

VR Box	Mini VR box for 3.5" – 6"
--------	---------------------------

## 5.2 Implementasi MDA

Pada bab sebelumnya, MDA dirancang untuk menentukan mekanik, dinamik, dan estetis dari permainan ini. Pada bab implementasi ini akan menjelaskan hasil implementasi dari MDA yang sudah dirancang sebelumnya. Seperti implementasi dari *scene*, karakter, tampilan UI, serta implementasi dari fungsi-fungsi kontrol dan juga estetis dari permainan.

### 5.2.1 Mekanik

#### 1. Karakter

Pada implementasi, model guru kemudian diberi animasi agar terlihat lebih hidup. Lalu guru hanya akan ditempatkan di dalam kelas serta ruang ujian sesuai dengan bab perancangan. Model guru hanya diam di tempat sampai pemain berinteraksi dengannya.



Gambar 5.1 Karakter Guru

Gambar 5.1 merupakan model karakter dari guru di dalam permainan. Model tersebut memiliki sebuah *agent control* yang dapat memungkinkan pemain agar dapat berinteraksi dengan karakter guru tersebut. *Agent control* tersebut berupa *sphere* kecil di kiri atas karakter. ketika *raycast* dari pemain mengenai *collider* dari karakter guru tersebut maka *sphere* tersebut akan menyala, mengidentifikasi bahwa pemain dapat berinteraksi dengan karakter guru tersebut dengan cara membentuk *gesture fist* pada tangan.

#### 2. Logo

Logo pada permainan ini diimplementasikan sesuai dengan konsep pada bab perancangan sebelumnya. Hasil dari implementasi logo dapat dilihat pada Gambar 5.2

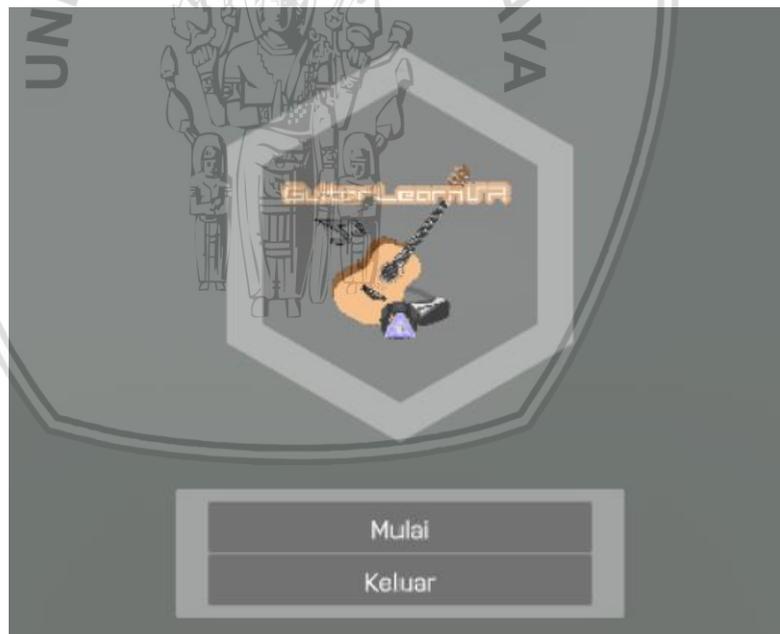


**Gambar 5.2 Logo permainan**

Gambar 5.2 merupakan logo permainan yang mana didesain semirip mungkin dengan konsep pada perancangan awal. Yang mana logo tersebut memiliki gambar gitar, HMD, serta Myo Armband. Pada implementasinya, logo tersebut mendapatkan tambahan tulisan “Guitar Learn VR” yang mana tulisan tersebut merupakan nama dari permainan ini.

### 3. User Interface

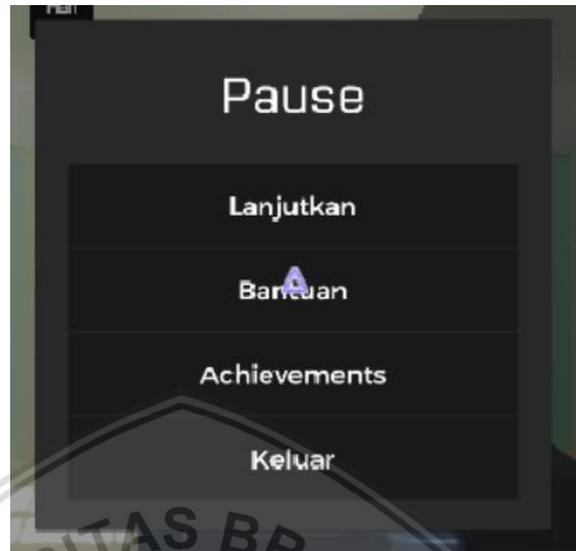
#### i. Tampilan *Title Menu*



**Gambar 5.3 Tampilan *Title Menu***

Gambar 5.3 merupakan tampilan jadi dari *Title Menu* permainan. Tampilan tersebut sudah sesuai dengan rancangan tampilan pada perancangan. Yang mana terdapat elemen-elemen seperti logo dari permainan, tombol Mulai yang berfungsi untuk memulai permainan serta tombol Keluar yang berfungsi untuk keluar/berhenti dari permainan.

ii. Tampilan *Pause*



**Gambar 5.4 Tampilan *Pause***

Gambar 5.4 merupakan gambar dari tampilan *Pause* ketika permainan dijeda. Tampilan tersebut sudah sesuai dengan konsep perancangan dari tampilan *Pause* sebelumnya. Yang mana terdapat elemen-elemen seperti label tulisan “*Pause*”, tombol Lanjutkan untuk melanjutkan permainan, tombol Bantuan untuk melihat tampilan tutorial, tombol Achievements untuk melihat daftar *achievement*, serta tombol Keluar untuk keluar dari permainan.

iii. Tampilan *Achievement*



**Gambar 5.5 Tampilan *Achievement***

Gambar 5.5 merupakan tampilan *Achievement*. Pada tampilan tersebut terdapat tombol Kembali yang digunakan untuk kembali ke tampilan pause, serta pada tampilan tersebut juga menampilkan daftar dari *Achievement* yang tersedia. Jika pemain



telah mendapatkan sebuah *achievement* maka warna *panel achievement*-nya ikut berubah menjadi hijau.

iv. Tampilan Dialog



**Gambar 5.6 Tampilan Dialog dengan guru**

Gambar 5.6 menunjukkan tampilan dialog dengan NPC guru. Tampilan tersebut sudah memenuhi elemen-elemen yang harus ada pada perancangan sebelumnya. Elemen-elemen tersebut ialah hari, gambar *avatar* dari karakter yang sedang berbicara serta kotak dialog beserta kalimatnya.

v. Tampilan *Game Over*

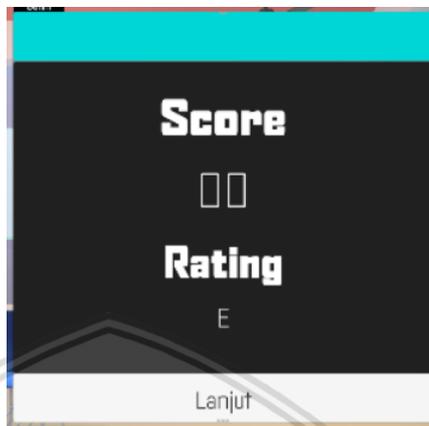


**Gambar 5.7 Tampilan *Game Over***

Gambar 5.7 merupakan tampilan dari permainan ketika *Game Over* atau berakhir. Tampilan tersebut hanya akan tampil jika pemain tidak berhasil memenuhi 3 kali pembelajaran sampai batas hari Minggu. Pada tampilan tersebut terdapat elemen-elemen

seperti (1)label bertuliskan “Game Over”, (2)tombol main lagi untuk mengulang permainan dari awal, serta (3)tombol keluar untuk keluar dari permainan.

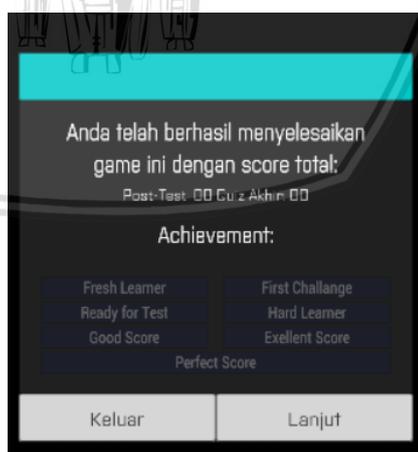
vi. Tampilan *Score*



**Gambar 5.8 Tampilan *Score***

Gambar 5.8 merupakan tampilan dari hasil *Score* setelah mengikuti *quiz*. Tampilan tersebut telah sesuai dengan perancangan yang mana terdapat elemen-elemen seperti (1)label tulisan *Score*, (2)nilai dari *score*, (3)label tulisan *Rating*, (4)*rating* dari *score* dan (5)tombol *Lanjut* untuk melanjutkan permainan.

vii. Tampilan *Ending Game*



**Gambar 5.9 Tampilan *Ending Game***

Gambar 5.9 merupakan tampilan dari *Ending Game*. Tampilan tersebut telah sesuai dengan perancangan awal sebelumnya yang mana tampilan tersebut menampilkan informasi seperti (1)label “Anda telah berhasil menyelesaikan *game* ini dengan *score* total:”, (2)nilai *score* total, (3)*achievement-*

achievement yang didapat oleh pemain, (4)tombol Keluar untuk keluar permainan, dan (5)tombol Lanjut untuk kembali ke menu awal permainan.

viii. Tampilan *Quiz*



Gambar 5.10 Tampilan *Quiz*

Gambar 5.10 merupakan tampilan ketika pemain sedang mengikuti tes. Pada tampilan tersebut terdapat (1)Timer waktu yang mana merupakan *timer* per soal, (2)Soal yang harus dijawab oleh pemain, (3)pilihan jawaban serta (4)benar atau salah jawaban yang telah dipilih pemain.

ix. Tampilan *Tutorial*



Gambar 5.11 Tampilan *Tutorial*

Gambar 5.11 menunjukkan implementasi dari tampilan *Tutorial*. Pada tampilan tersebut terdapat beberapa elemen UI seperti yang telah dirancang pada bab perancangan. Elemen-

elemen tersebut seperti (1)label tulisan “Tutorial”, (2)Penjelasan tutorial, (3)serta label “*Wave Right*” untuk berpindah ke tampilan Tutorial selanjutnya(pemain harus membentuk *gesture wave right* terlebih dahulu).

x. Tampilan *Notice*

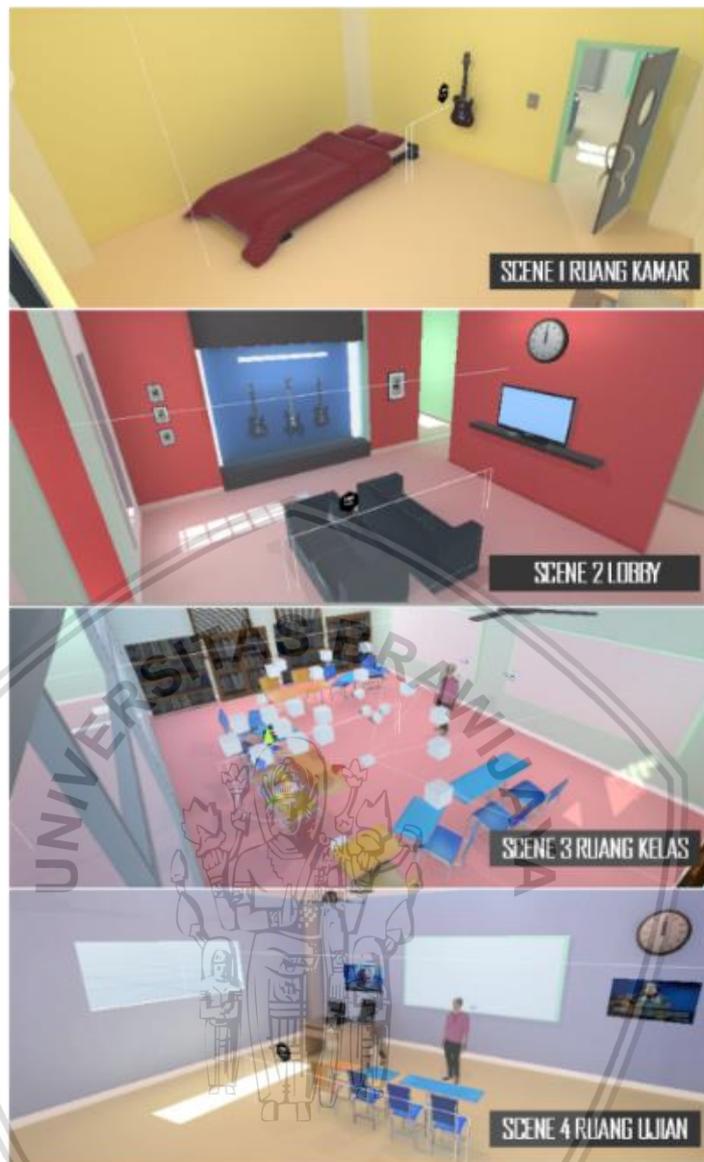


Gambar 5.12 Tampilan *Notice* yang terdiri dari (1)*notice 1*, (2)*notice 2*, dan (3)*notice 3*

Gambar 5.12 menunjukkan implementasi dari tampilan *notice*. Pada masing-masing jenis *notice* memiliki kegunaan tersendiri, seperti *notice 1* yang digunakan untuk menampilkan pesan “Mendapatkan *achievement* baru”, *notice 2* yang digunakan untuk mengkonfirmasi aksi pemain, dan *notice 3* yang digunakan untuk memberi peringatan/*warning* kepada pemain.

4. *Scene*

*Scene* dari permainan diimplementasikan sesuai dengan perancangan yang telah dilakukan sebelumnya. Terdapat 4 *scene* utama di dalam permainan. *Scene* tersebut ialah kamar pemain, lobby dari tempat belajar, kelas, dan ruang ujian. Implementasi dari *scene* dapat dilihat pada Gambar 5.13



**Gambar 5.13** Gambar dari *scene* ruangan yang terdiri dari (1) *scene* kamar, (2) *scene lobby*, (3) *scene* ruangan kelas dan (4) *scene* ruangan ujian

Gambar 5.13 merupakan pratinjau gambar lingkungan untuk keempat *scene*. Sub gambar (1) merupakan *scene* kamar, *scene* tersebut dibangun sesuai dengan sketsa yang dilakukan sebelumnya pada bab perancangan yang mana *scene* kamar tersebut memiliki meja, kamar tidur, ac dan lemari. Sub gambar (2) merupakan *scene lobby* permainan. perancangan *scene lobby* tersebut ialah terdapat kursi, lampu, dan pajangan gitar. Guna dari *scene* ini ialah sebagai penghubung antara kamar pemain dengan ruang kelas. Sub gambar (3) merupakan *scene* kelas, pada *scene* ini pembelajaran akan berlangsung. Pada *scene* tersebut akan terdapat meja, papan dan guru yang menerangkan materi. Sub gambar (4) merupakan *scene* ruang ujian, pada *scene* ini ujian akhir akan dilaksanakan.

Pada ruangan ini sendiri akan terdapat meja, serta papan yang mengidentifikasi bahwa ruangan tersebut merupakan ruangan ujian.

5. Implementasi Kontrol dan Aturan

Kontrol dari pemain berasal dari Myo *Armband*. Masukan dari Myo *Armband* akan digunakan untuk menggerak karakter, berpindah menu, memilih menu serta berinteraksi dengan object lingkungan. masukan dari Myo *Armband* tersebut berupa pose tangan yang dibentuk oleh pemain. Pose tersebut akan diproses dengan menggunakan script untuk mengatur kondisi jika pose tersebut dibentuk maka apa yang akan dilakukan oleh sistem.

Tabel 5.3 merupakan tabel *pseudocode* dari kontrol pergerakan pemain. *Pseudocode* tersebut menerangkan jika pose dari tangan pemain bernilai *wave in* maka yang akan dilakukan oleh sistem ialah menggerakkan karakter ke depan. Sementara jika pose tangan bernilai *wave out* maka karakter akan bergerak ke belakang.

**Tabel 5.3 Pseudocode kontrol pergerakan pemain**

Kontrol pergerakan pemain	
1	Di dalam <i>method update</i> monobehaviour unity
2	IF pose sekarang bernilai pose wave in.
3	GERAKAN karakter ke depan
4	ELSEIF pose sekarang bernilai wave out.
5	GERAKAN karakter ke belakang
6	ENDIF



**Gambar 5.14 Keluaran dari kontrol pergerakan pemain ketika mundur(kiri) dan maju(kanan)**

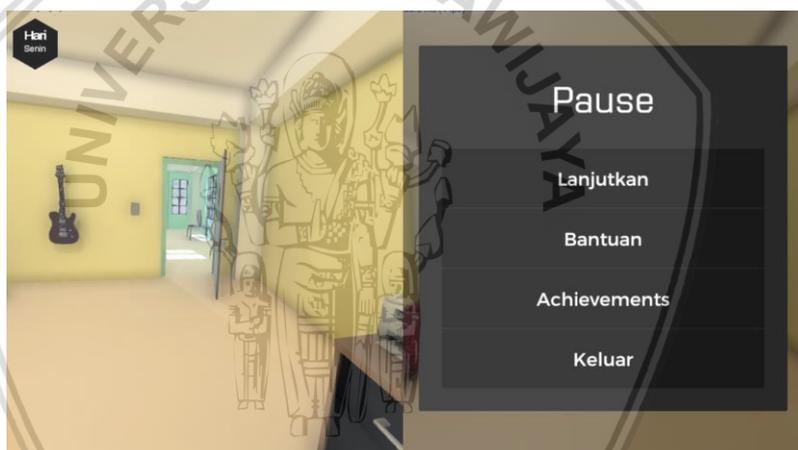
Gambar 5.14 merupakan gambaran Keluaran dari kontrol pergerakan pemain. Ketika pemain membentuk *gesture wave in* maka pemain akan bergerak ke depan(kanan) dan ketika pemain membentuk *gesture wave out* maka pemain akan bergerak ke belakang(kiri).

Untuk membuka tampilan *pause* di dalam permainan, pemain dapat membukanya dengan membentuk *gesture fingerspread*. Setelah pemain membentuk *gesture* tersebut, maka tampilan *pause* akan muncul dan permainan akan dijeda. Permainan akan kembali berjalan jika pemain membentuk *gesture fingerspread* sekali lagi. Pseudocode untuk menjeda permainan dapat dilihat pada Tabel 5.4.



**Tabel 5.4 Pseudocode membuka menu *pause***

	Kontrol membuka <i>menu pause</i>
1	Di dalam method update monobehaviour unity
2	IF pose terakhir dari myo tidak sama dengan pose myo sekarang
3	Set Pose terakhir dengan nilai pose sekarang
4	IF pose sekarang sama dengan fingerspread
5	IF game tidak di-pause
6	Pause game
7	Tampilkan menu pause
8	ELSEIF permainan di-pause
9	lanjutkan game
10	Sembunyikan menu pause
11	ENDIF
12	ENDIF
13	ENDIF
14	ENDIF
15	ENDIF



**Gambar 5.15 Hasil keluaran dari *pseudocode* membuka menu *pause* di dalam permainan**

Ketika pemain membentuk *gesture fingerspread* maka permainan akan dijeda dan sistem akan menampilkan menu *pause* kepada pemain seperti yang tertampilkan pada Gambar 5.15.

Di dalam permainan, untuk berpindah memilih *button* pemain harus membentuk pose *wave in* dan *wave out* serta pose *fist* untuk mengeksekusi perintah untuk *button* yang terpilih. Pose *wave in* digunakan untuk memilih menu UI sebelumnya sedangkan *wave out* digunakan untuk memilih menu UI selanjutnya. Ketika menggunakan Myo Armband sebagai masukan memilih menu, perlu diberi batasan agar ketika membentuk salah satu pose tersebut maka program tidak akan mengubah pilihan menu secara terus-menerus setiap kali perulangan *frame* dari permainan. Penggunaan *variable last\_pose* dapat membantu untuk menyelesaikan masalah ini. Jika *last\_pose* tidak sama dengan pose sekarang maka masuk

percabangan if untuk jenis pose untuk mengupdate nilai last\_pose dan melakukan perubahan memilih *button* UI. Jika sama maka jangan lakukan apa-apa. Dengan begini, program akan mengganti pilihan menu setiap kali pemain membentuk pose tersebut setelah pose lainnya dibentuk. *Pseudocode* untuk fungsi tersebut dapat dilihat pada Tabel 5.5.

**Tabel 5.5 *Pseudocode* berpindah memilih *button* dan memilih menu**

	Kontrol berpindah memilih <i>button</i> UI dan memilih menu
1	Membuat array dari <i>button</i>
2	Membuat variable <i>selected</i>
3	Pada method <i>onEnable</i>
4	Set nilai <i>selected</i> menjadi 0
5	Tutup method <i>onEnable</i>
6	Di dalam method <i>Update</i>
7	IF pose terakhir dari <i>myo</i> tidak sama dengan pose <i>myo</i> sekarang
8	SET Pose terakhir sama dengan pose sekarang
9	sekarang
10	IF pose sekarang sama dengan pose <i>wave in</i>
11	IF <i>selected</i> bernilai 0
12	SET <i>Selected</i> sama dengan Panjang array <i>button</i> dikurangi 1
13	dengan Panjang array <i>button</i> dikurangi 1
14	Else
15	KURANGI nilai <i>selected</i> sebanyak 1
16	ENDIF
17	ELSEIF pose sekarang sama dengan pose <i>waveout</i>
18	IF <i>selected</i> bernilai sebesar Panjang array dikurangi 1
19	SET <i>Selected</i> bernilai 0
20	0
21	ELSE
22	TAMBAH nilai <i>selected</i> sebanyak 1
23	selected sebanyak 1
24	ENDIF
25	ELSEIF pose sekarang sama dengan pose <i>fist</i>
26	INVOKE array dari <i>button</i> dengan index <i>selected</i>
27	dengan index <i>selected</i>
28	ENDIF
29	ENDIF
30	ELSEIF pose sekarang sama dengan pose <i>fist</i>
31	INVOKE array dari <i>button</i> dengan index <i>selected</i>
32	dengan index <i>selected</i>
33	ENDIF
34	ENDIF



**Gambar 5.16** Hasil keluaran dari *pseudocode* berpindah memilih *button* dan memilih menu

Gambar 5.16 merupakan gambaran kondisi yang terjadi ketika pemain berpindah memilih *button* UI dengan menggunakan pose *wave in* dan *wave out*. *Button* akan dipilih sesuai dengan masukan dari pemain. *Button* dengan warna biru mengidentifikasi bahwa *button* tersebut sedang dipilih oleh pemain.

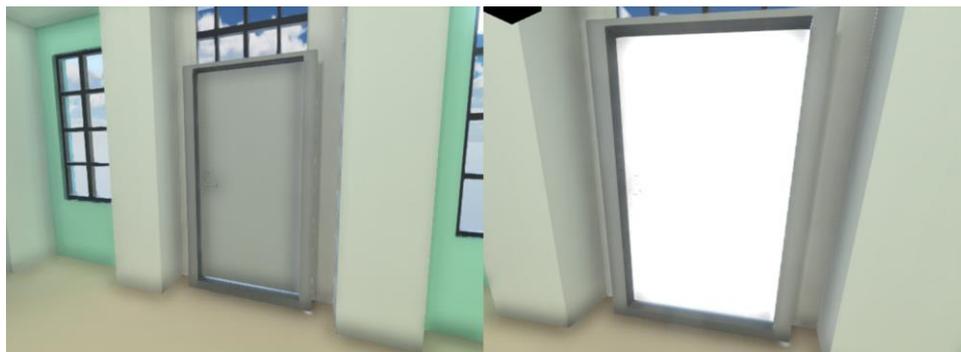
Di dalam permainan sendiri pemain dapat berinteraksi dengan beberapa objek di dalam permainan. Objek tersebut dapat diinteraksi ketika kita sudah dekat dan melihat ke arahnya. Untuk mengimplementasikan fungsi tersebut, penggunaan *raycast* pada *unity* akan sangat membantu. *Raycast* akan ditembakkan dari tengah kamera pemain sepanjang 2 *float*. Ketika *raycast* tersebut berbenturan dengan objek yang dapat diinteraksikan maka objek tersebut akan memancarkan cahaya atau *emission*. Dan ketika itu pemain membentuk pose *fist* maka *event* dari objek tersebut akan dijalankan. Ketika *raycast* sudah tidak membentur dengan objek tersebut maka objek tersebut akan berubah menjadi semula (tidak memancarkan sinar). Alur algoritma dapat dilihat pada Tabel 5.6.

**Tabel 5.6** *Pseudocode* berinteraksi dengan objek

	Berinteraksi dengan object
1	Di dalam method update monobehaviour unity
2	MEMBUAT <i>raycast</i> di tengah-tengah kamera pemain dengan jarak tembak kedepan sebesar 2 float
3	IF <i>raycast</i> mengenai object
4	IF object tersebut dapat diinteraksikan atau merupakan sebuah tombol 3D
5	SET object tersebut bersinar
	IF Pemain membentuk pose fist
6	IF scene sekarang berada di kamar



7	IF object tersebut pintu lobby
8	UBAH scene ke lobby
9	IF object tersebut kasur
10	MELAKUKAN tidur
11	IF scene sekarang berada di lobby
12	IF object tersebut pint uke kelas
13	
14	UBAH scene ke kelas
15	IF object tersebut pintu menuju ke kamar
16	UBAH scene ke kamar
17	IF scene sekarang berada di kelas
18	IF object tersebut pintu ke ruang ujian
19	UBAH scene ke ujian
20	IF object tersebut pintu menuju lobby
21	UBAH scene ke lobby
22	IF object tersebut NPC guru dan belum mengikuti kelas untuk hari tersebut
23	
24	MULAI kelas pembelajaran
25	IF object tersebut NPC guru dan sudah mengikuti kelas untuk hari tersebut
26	
27	MULAI post test setelah pembelajaran
28	IF object tersebut tombol play video ke 2 di papan
29	MAINKAN video 2 di papan
30	
31	IF object tersebut tombol play video ke 1 di papan
32	MAINKAN video 1 di papan
33	
34	IF scene sekarang berada di ruang ujian
35	IF object tersebut pintu menuju kelas
36	
37	UBAH scene ke kelas
38	IF object tersebut NPC guru
39	MULAI test akhir
40	ELSEIF raycast sudah tidak mengenai object tersebut lagi
	MENGEMBALIKAN object tersebut menjadi tidak bersinar



**Gambar 5.17** Hasil keluaran dari *pseudocode* berinteraksi dengan objek

Gambar 5.17 menunjukkan bahwa objek yang sebelumnya belum terkena *raycast* terlihat normal(kiri), sedangkan objek yang sudah terkena *raycast* akan terlihat memancarkan cahaya putih(kanan) yang mana mengidentifikasi bahwa objek tersebut sedang dipilih.

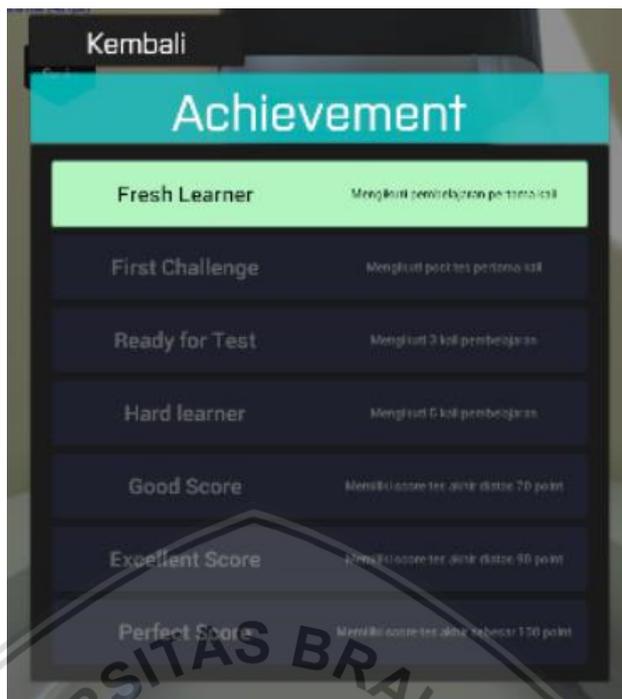
6. Implementasi sistem *Achievement*

Di dalam permainan terdapat sistem *reward achievement*. Sistem ini akan memberikan semacam penghargaan kepada pemain ketika telah menyelesaikan sebuah kondisi tertentu. Untuk mengimplementasikannya, pertama perlu dipersiapkan variabel-variabel untuk masing-masing *achievement*-nya. setelah itu, program akan mengecek secara rutin kondisi dari *achievement-achievement* tersebut untuk memperbarui tampilan UI. Penjelasan algoritma dapat dilihat pada Tabel 5.7.

**Tabel 5.7 Pseudocode achievement**

	Implementasi achievement
1	IF telah mengikuti kelas untuk pertama
2	kalinya
3	TAMPIL, mendapatkan achievement
4	IF telah mengikuti posttest untuk pertama
	kalinya
5	TAMPIL, mendapatkan achievement
6	IF telah mengikuti kelas sebanyak 3 kali
	TAMPIL, mendapatkan achievement
7	IF telah mengikuti kelas sebanyak 5 kali
8	TAMPIL, mendapatkan achievement
9	IF memiliki skor akhir lebih dari sama dengan
	70
10	TAMPIL, mendapatkan achievement
11	IF memiliki nilai test akhir lebih dari 80
12	TAMPIL, mendapatkan achievement
13	IF memiliki nilai test akhir sebesar 100
14	TAMPIL, mendapatkan achievement
15	ENDIF





**Gambar 5.18** Hasil keluaran dari *pseudocode achievement*

Gambar 5.18 menunjukkan perubahan dari panel *Achievement* yang mana ketika pemain telah menyelesaikan salah satu *achievement* maka warna panel *achievement* tersebut akan berubah menjadi berwarna hijau.

7. Implementasi Level

Di dalam permainan, level permainan ada 2 yaitu *post test* dengan tes akhir. Perbedaan dari *post-test* dan tes akhir sendiri ialah adanya batas waktu untuk menjawab soal. Untuk mengimplementasikannya, dapat dengan menggunakan *enumerator* pada *unity3D*. *Enumerator* dapat memungkinkan kita untuk menghitung *countdown* yang digunakan untuk menjawab masing-masing soal. Untuk fungsi *quiz*, alur kerjanya ialah program akan menjalankan fungsi *prepare* terlebih dahulu yang tujuannya ialah untuk mempersiapkan pemain untuk menjawab soal. Di dalam *prepare* tersebut kemudian akan memanggil fungsi *showsoal*. *Showsoal* digunakan untuk menampilkan soal ke UI permainan. Setelah itu *quiz* akan berjalan sampai pemain menjawab semua soal yang ada. Penjelasan algoritma lebih lanjut dapat dilihat pada Tabel 5.8.

**Tabel 5.8** *Pseudocode* implementasi *quiz*

	Implementasi quiz
1	Di dalam method prepare
2	Memulai sebuah coroutine untuk membuat timer untuk pemain
3	END prepare
4	Enumerator areYouReady



```

5      Membuat string readyBois
6      Memuat variable indexString
7      While true
8          IF Countdown soal telah mencapai 0
9              MUNCULKAN soal
10         Else
11             Mengurangi nilai countdown
12             sebanyak 1
13             Tunggu 1 detik lalu loop dari awal
14         ENDIF
15     END areyouready
16     Di dalam method showSoal()
17         IF jumlah soal yang belum dijawab telah habis
18             TAMPILKAN nilai dari test
19         ELSE
20             TAMPILKAN soal pada UI
21         IF jika pemain sekarang berada di ruang kelas
22             Tidak menghitung countdown per soal
23         Else
24             Mulai menghitung countdown per soal
25         ENDIF
26     END showsoal()
    
```

### 5.2.2 Dinamik

Dari dinamik yang telah dirancang pada awal permainan didapatkan hasil:

#### 1. Progression

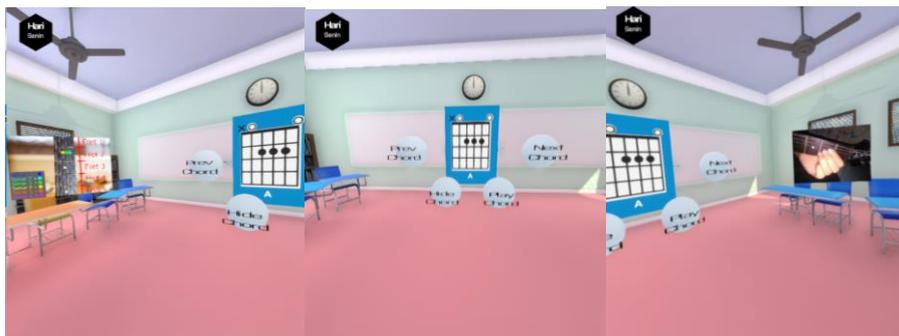


Gambar 5.19 Progression dari pemain

Pada awal permainan pemain masih belum menyelesaikan apapun, namun seiring permainan berlangsung pemain akan mendapatkan *achievement* seperti pada Gambar 5.19. Pendapatan *achievement* tersebut merupakan progression dari pemain.

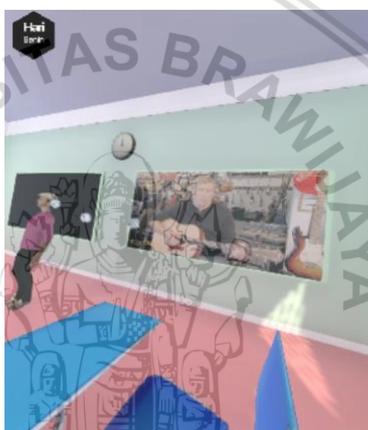


## 2. Creativity



**Gambar 5.20** Pemain belajar dengan cara *browse*

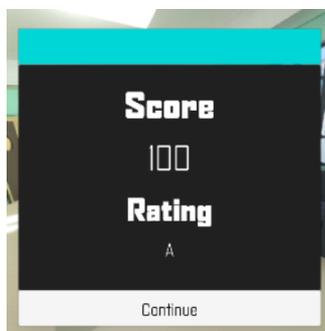
Sesuai kreatifitas dari pemain, pemain dapat mempelajari materi dengan beberapa cara (setelah melakukan pembelajaran). Gambar 5.20 merupakan pembelajaran materi dengan cara *browsing*.



**Gambar 5.21** Pemain belajar dengan melihat video tutorial

Terdapat juga mode pembelajaran dengan cara melihat video tutorial yang dapat pemain liat melalui papan seperti pada Gambar 5.21. Video tersebut dapat diakses dengan menggunakan tombol 3D di dekat papan.

## 3. Productivity



**Gambar 5.22** Hasil dari produktivitas pemain

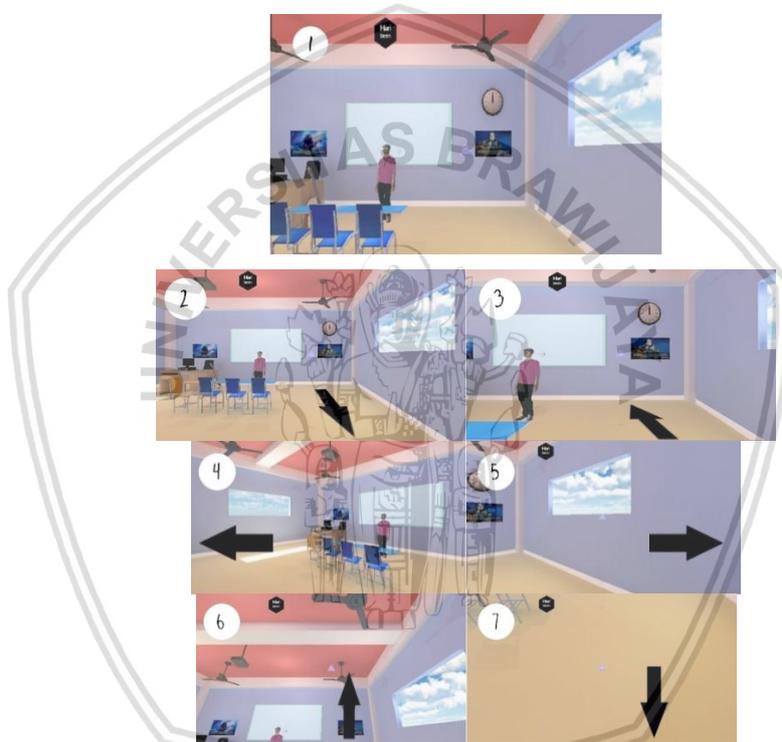
Produktifitas dari pemain dapat dilihat juga dari hasil test yang ia dapatkan, semakin tinggi nilai tes tersebut maka menandakan bahwa materi yang

telah dipelajari oleh pemain semakin tinggi menandakan bahwa pemain menggunakan waktunya untuk belajar.

### 5.2.3 Estetis

#### 1. Sensation

Sensasi permainan ini ialah pemain dapat menjelajah antar ruangan di dalam permainan secara virtual. Dengan menggunakan *HMD(Smartphone)* sebagai media untuk melihat lingkungan secara virtual dan *Myo Armband* sebagai masukan untuk kontrol pergerakan karakternya, interaksi dengan objek dan pemilihan menu. Dengan kemampuan VR tersebut pemain dapat merasakan sensasi nyata seperti dia berada di dalam lingkungan tersebut.



Gambar 5.23 nomor (1) merupakan posisi awal pemain, (2) merupakan keadaan ketika pemain bergerak mundur ke belakang, (3) keadaan ketika pemain bergerak maju kedepan, (4) keadaan ketika pemain menengok ke kiri, (5) keadaan ketika pemain menengok ke kanan, (6) keadaan pemain ketika melihat ke atas, (7) keadaan pemain ketika melihat ke bawah.

Gambar 5.23 merupakan gambaran permainan ketika pemain memainkan permainan ini. Pemain dapat menengokkan kepalanya secara 360 derajat dan dapat bergerak maju dan mundur mengelilingi ruangan.

#### 2. Challenge

Pemain merasakan tantangan, keseruan dan rasa akan kebanggaan yang didapatkan dari penyelesaian sebuah *achievement* di dalam permainan,

dan menyelesaikan rintangan seperti *post test* dan tes akhir di dalam permainan.

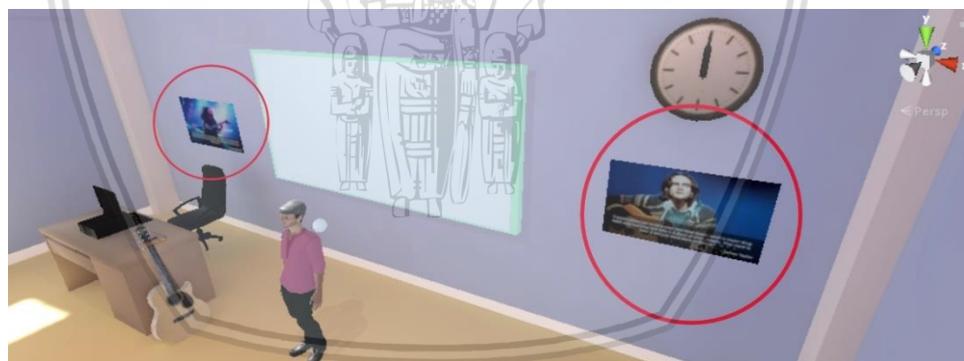


**Gambar 5.24 Beberapa *challenge* yang ada di dalam permainan.**

Gambar 5.24 menunjukkan elemen *challenge* yang ada di dalam permainan. Elemen-elemen tersebut seperti post test, test akhir dan *achievement*.

### 3. *Discover*

Pemain merasakan kesenangan dan motivasi yang didapatkan dari penemuan kata-kata motivasi di dalam permainan. Kata-kata tersebut tersebar di dinding-dinding ruangan di dalam permainan dalam bentuk poster *in-game*.



**Gambar 5.25 Elemen *discover* yang ada di dalam permainan**

Gambar 5.25 menunjukkan bahwa poster yang dilingkari merah merupakan contoh dari elemen *discover*. Poster tersebut terletak pada dinding ruangan yang mana dapat ditemukan oleh pemain jika berkeliling di dalam ruangan. Jika diperhatikan lebih lanjut, pada poster tersebut terdapat kata-kata motivasi dari musisi-musisi gitar terkenal yang mana dapat menyemangati pemain untuk belajar gitar.

## BAB 6 HASIL DAN PEMBAHASAN

### 6.1 Pengujian *Black Box*

Pengujian *black box* digunakan untuk mengecek fungsionalitas perangkat lunak. Pengujian *black box* berfokus pada pengecekan masukan dan keluaran dari fungsi atau fitur yang ada di dalam perangkat lunak. masukan dan keluaran dari perangkat lunak akan divalidasi untuk mendeteksi ketidaksesuaian fungsi-fungsi yang terdapat di dalam perangkat lunak tersebut.

#### 6.1.1 Pengujian Kontrol Pergerakan Pemain

Tabel 6.1 Hasil pengujian kontrol pergerakan pemain

No	Test case	Expected Result	Result	Status
1	Menggerakkan pemain ke depan dengan menggunakan pose <i>wave in Myo Armband</i>	Pemain akan maju ke depan	Pemain maju ke depan	Valid
2	Menggerakkan pemain ke belakang dengan menggunakan pose <i>wave out Myo Armband</i>	Pemain akan bergerak ke belakang	Pemain mundur ke belakang	Valid
3	Melihat ke kiri dengan cara menolehkan kepala pemain ke kiri	Kamera akan berotasi ke kiri	Kamera berotasi ke kiri	Valid
4	Melihat ke kiri dengan cara menolehkan kepala pemain ke kanan	Kamera akan berotasi ke kanan	Kamera berotasi ke kanan	Valid
5	Melihat ke atas dan ke bawah dengan cara menganggukan kepala	Kamera akan bergerak ke atas dan ke bawah	Kamera bergerak ke atas dan ke bawah	Valid

Tabel 6.1 merupakan hasil dari pengujian *black box* untuk setiap kasus uji pada fungsi pergerakan pemain yang ada di dalam permainan. Dari hasil pengujian pada tabel tersebut, didapatkan data valid untuk setiap kasus uji. Sehingga untuk fungsi tersebut dapat dikatakan telah berjalan dengan benar.

### 6.1.2 Pengujian Pemilihan Menu

Tabel 6.2 Hasil pengujian pemilihan menu

No	Test case	Expected Result	Result	Status
1	Membentuk pose tangan <i>wave in</i> untuk memilih <i>button</i> menu sebelumnya	<i>Button</i> menu sebelumnya akan terpilih	<i>Button</i> menu sebelumnya terpilih	Valid
2	Membentuk pose tangan <i>wave out</i> untuk memilih <i>button</i> menu selanjutnya	<i>Button</i> menu selanjutnya akan terpilih	<i>Button</i> menu selanjutnya terpilih	Valid
3	Membentuk pose tangan <i>fist</i> untuk memilih <i>button</i> yang aktif sekarang	<i>Button</i> akan terpilih dan dieksekusi	<i>Button</i> terpilih dan dieksekusi	Valid

Tabel 6.2 merupakan hasil dari pengujian *black box* untuk setiap kasus uji pada fungsi pemilihan *UI* yang ada di dalam permainan. Dari hasil pengujian pada tabel tersebut, didapatkan data valid untuk setiap kasus uji. *Test case* pertama mendapatkan hasil valid, *test case* ke 2 mendapatkan hasil valid, dan *test case* ketiga mendapatkan hasil valid Sehingga untuk fungsi tersebut dapat dikatakan telah berjalan dengan benar.

### 6.1.3 Pengujian Berinteraksi dengan Objek

Tabel 6.3 Hasil pengujian berinteraksi dengan objek permainan

No	Test case	Expected Result	Result	Status
1	Pemain melihat ke arah objek yang dapat diinteraksi dengan jarak diluar <i>raycast</i>	Objek tidak akan bersinar	Objek tidak bersinar	Valid

2	Pemain melihat ke arah objek yang dapat diinteraksi dengan jarak jangkauan <i>raycast</i>	Objek akan bersinar	Objek bersinar	Valid
3	Pemain membentuk pose <i>fist</i> terhadap objek yang telah bersinar	Objek akan mengeksekusi event yang telah ditanamkan di objek tersebut	Objek mengeksekusi event yang telah ditanamkan di objek tersebut	Valid

Tabel 6.3 Hasil pengujian berinteraksi dengan objek permainan merupakan hasil dari pengujian *black box* untuk setiap kasus uji pada fungsi interaksi dengan objek yang ada di dalam permainan. Dari hasil pengujian pada tabel tersebut, didapatkan data valid untuk setiap kasus uji. Sehingga untuk fungsi tersebut dapat dikatakan telah berjalan dengan benar.

#### 6.1.4 Pengujian *Achievement*

Tabel 6.4 Hasil pengujian *achievement*

No	<i>Test case</i>	<i>Expected Result</i>	<i>Result</i>	Status
1	Pemain memperoleh <i>achievement</i>	Akan muncul <i>notice</i> mendapatkan <i>achievement</i>	Muncul <i>notice</i> mendapatkan	Valid
2	Pemain memperoleh <i>achievement</i>	Warna panel dari <i>achievement</i> tersebut akan berubah menjadi hijau	Warna panel dari <i>achievement</i> tersebut berubah menjadi hijau	Valid

Tabel 6.4 merupakan hasil dari pengujian *blackbox* untuk setiap kasus uji pada fungsi *achievement* yang ada di dalam permainan. Dari hasil pengujian pada tabel tersebut, didapatkan data valid untuk setiap kasus uji. *Test case* pertama mendapatkan hasil valid dan *test case* kedua mendapatkan hasil valid Sehingga untuk fungsi *achievement* di dalam permainan telah berjalan dengan benar.

### 6.1.5 Pengujian Quiz

Tabel 6.5 Hasil pengujian quiz

No	Test case	Expected Result	Result	Status
1	Pemain memilih jawaban benar	Akan muncul indikator benar	muncul indikator benar	Valid
2	Pemain memilih jawaban salah	Akan muncul indkator salah	muncul indikator benar	Valid
3	Pemain tidak memilih jawaban sampai waktu habis	Akan muncul indikator salah	muncul indikator salah	Valid
4	Pemain mengikuti <i>post test</i>	Tidak akan terdapat Batasan waktu mengerjakan soal	Tidak terdapat Batasan waktu mengerjakan soal	Valid
5	Pemain mengikuti <i>test akhir</i>	Akan terdapat Batasan waktu mengerjakan soal	Terdapat Batasan waktu mengerjakan soal	Valid

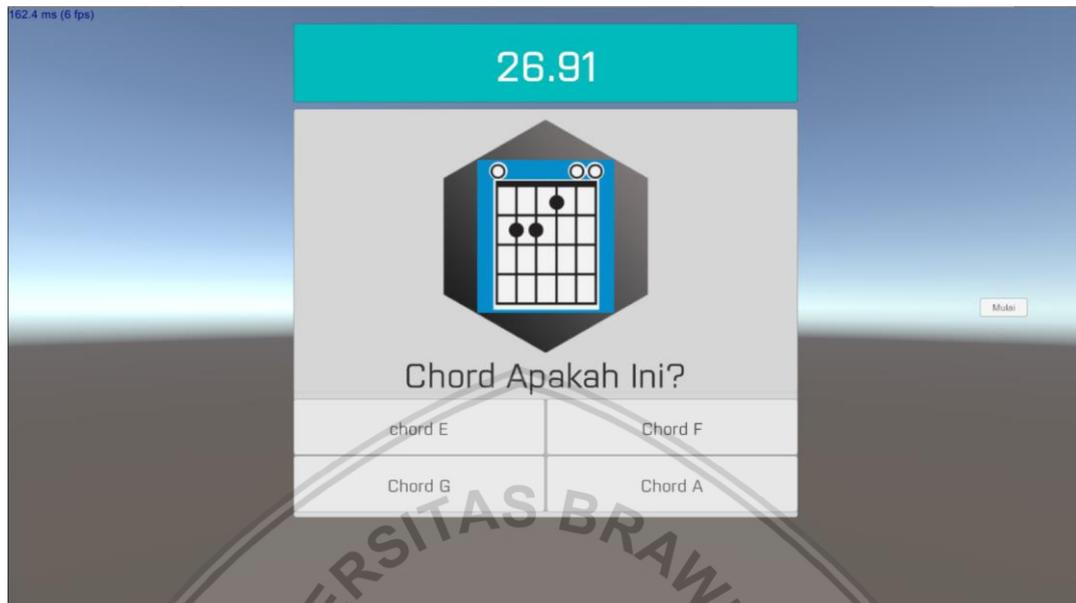
Tabel 6.5 merupakan hasil dari pengujian *blackbox* untuk setiap kasus uji pada *quiz* yang ada di dalam permainan. Dari hasil pengujian pada tabel tersebut, didapatkan data valid untuk setiap kasus uji. Sehingga untuk fungsi *quiz* di dalam permainan telah berjalan dengan benar.

Dari pengujian *black box* yang didapatkan, semua fungsional dari permainan teruji menghasilkan keluaran yang valid semua. Dengan demikian sistem dapat berjalan dengan harapan tanpa kesalahan.

### 6.2 Pengujian *Pre-test* dan *Post-test*

Pengujian ini akan melibatkan 5 partisipan dalam pengujiannya. Partisipan akan mengikuti beberapa tahap dalam pengujian ini. Tahap pertama, partisipan akan melakukan *pre-test* terlebih dahulu sebelum memainkan permainan edukasi ini. Pada tahap *pre-test* ini, tes akan dilakukan pada *scene pretest* yang sudah dibuat di dalam permainan. Pada *scene* tersebut terdapat soal sebanyak 10 soal

dengan *timer* masing-masing soal sebesar 30 detik. Kondisi tersebut disamakan dengan kondisi soal saat ujian akhir di dalam permainan. Persamaan kondisi tersebut bertujuan agar data dapat terkumpul secara akurat.



**Gambar 6.1 Tampilan UI pre-test**

Gambar 6.1 menunjukkan tampilan saat tes *pre-test*. Pada tampilan tersebut akan terdapat gambar soal, pertanyaan serta 4 tombol pilihan untuk memilih jawaban. Sebelum soal ditampilkan, pemain harus memilih tombol mulai untuk memulai mengikuti *pre-test*.

Insight Assesment(2018) menuliskan bahwa untuk sebuah pengujian dengan partisipan yang kurang dari 25 partisipan, dapat menghitung nilai dari perubahan dengan menggunakan rumus untuk mencari gain yaitu:

$$Gain = y_2 - y_1$$

Yang mana:

Y2 = merupakan nilai post-test

Y1 = merupakan nilai pre-test

Dari rumus tersebut kemudian didapatkan hasil seperti pada tabel Tabel 6.6

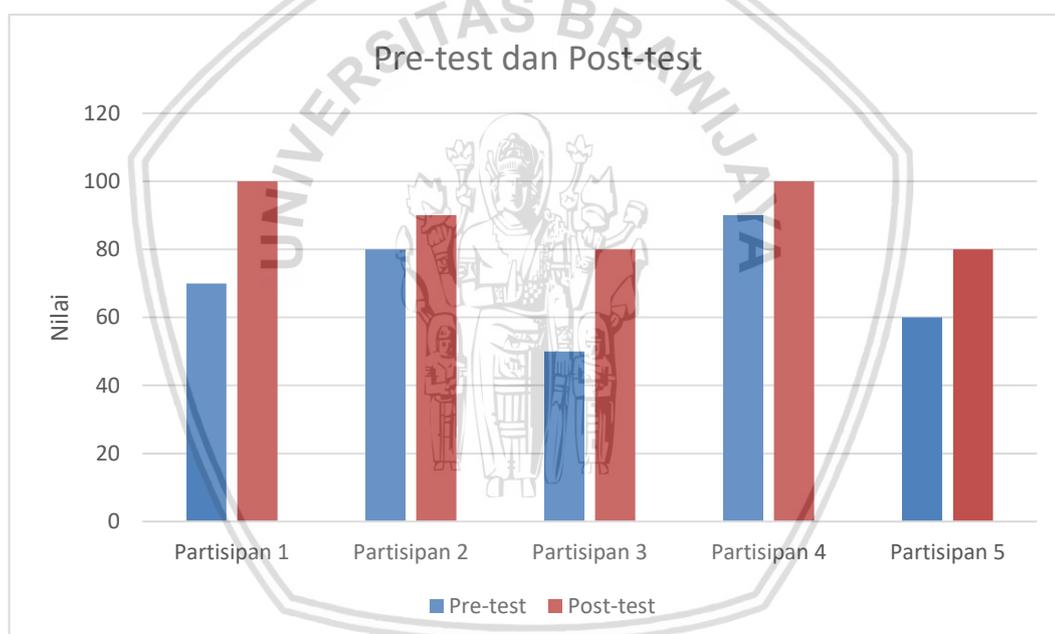
**Tabel 6.6 Hasil *pre-test* dan *post-test* dari masing-masing partisipan**

No	Partisipan	<i>Pre-test</i>	<i>Post-test</i>	Gain
1	Partisipan 1	70	100	30
2	Partisipan 2	80	90	10
3	Partisipan 3	50	80	30
4	Partisipan 4	90	100	10



5	Partisipan 5	60	80	20
Rata-rata		70	90	20

Tabel 6.6 merupakan hasil dari pengujian *pre-test* dan *post-test* untuk masing-masing partisipan. Hasil dari *pre-test* dan *post-test* dari masing-masing partisipan dapat dihitung selisihnya untuk mengetahui perubahan baik peningkatan ataupun penurunan. Berdasarkan tabel tersebut, partisipan 1(p1) mengalami perubahan sebesar 30, partisipan 2(p2) mengalami perubahan sebesar 10, partisipan 3(p3) mengalami perubahan sebesar 30, partisipan 4 mengalami perubahan sebesar 10, dan partisipan 5(p5) mengalami perubahan sebesar 20. Perubahan terbesar yang terjadi ialah sebesar 30 oleh partisipan 1 dan partisipan 3. Sedangkan perubahan terkecil yang terjadi ialah sebesar 10 yang dialami oleh partisipan 2 dan 4.



**Gambar 6.2 Hasil *pre-test* dan *post-test***

Gambar 6.2 merupakan grafik dari hasil *pre-test* dan *post-test*. Grafik yang berwarna biru merupakan representasi nilai dari *pre-test* sedangkan grafik yang berwarna merah merupakan representasi nilai dari *post-test*. Dapat dilihat juga secara keseluruhan, bahwa semua nilai dari *post-test* bernilai lebih besar daripada nilai *pre-test*.

Secara garis besar, setiap partisipan yang memainkan permainan ini mengalami peningkatan pada hasil *post-test* nya jika dibandingkan dengan nilai *pre-test* nya. Hal tersebut juga menandakan bahwa permainan ini telah berfungsi sebagai media pembelajaran.



### 6.3 Pengujian *Task Completion*

Pengujian *task completion* digunakan agar dapat mengetahui tingkat keefisienan dari perangkat lunak. Pemain diberikan tugas sebanyak 7 tugas yang harus diselesaikan selama proses pengujian. Dengan memperhatikan total keberhasilan pemain untuk menyelesaikan semua tugas, pengujian ini dilakukan. Adapun jenis *task* yang akan diberikan kepada pemain seperti yang ditampilkan pada Tabel 6.7

**Tabel 6.7 *Task* yang diberikan kepada pemain**

No	<i>Task</i> yang akan pemain selesaikan
1	Berpindah ke ruangan <i>lobby</i> dan kembali lagi ke ruangan kamar
2	Membuka menu <i>pause</i>
3	Membuka <i>achievement</i> dan melihat ada berapa banyak <i>achievement</i> yang ada
4	Tidur dan mengkonfirmasi bahwa hari telah berganti
5	Pergi ke kelas dan mengikuti pembelajaran
6	Mengikuti post test dan melihat hasilnya
7	Mengikuti test akhir

Latar belakang penentuan *task* tersebut ialah untuk menguji seberapa bisa partisipan nantinya melakukan aksi standar yang ada di dalam permainan. Dengan begitu dapat diketahui seberapa sulit permainan ini bagi pemain. Sebagai contoh seperti *task* 1 “berpindah ke ruangan *lobby* dan kembali ke ruangan kamar”, *task* tersebut ditentukan dengan alasan untuk melihat seberapa pemahaman partisipan nantinya terhadap pemetaan antar ruangan serta bagaimana cara bergerak dan berinteraksi dengan objek. *Task* 2 ditentukan dengan alasan untuk melihat bagaimana partisipan dapat memanipulasi jalannya permainan. *Task* 3 ditentukan untuk melihat kesulitan partisipan dalam mencari informasi dan juga bergerak memilih UI. *Task* 4 ditentukan untuk mengetahui seberapa paham partisipan dalam mengerti aturan di dalam permainan agar dapat melanjutkan permainan ke depan. *Task* 5 ditentukan untuk mengetahui pemahaman partisipan akan bagaimana cara mengikuti pembelajaran. *Task* 6 ditentukan untuk mengetahui apakah partisipan mengerti cara untuk mengikuti post-test atau tidak. *Task* 7 ditentukan apakah pemain dapat mengerti mengenai syarat untuk mengikuti tes akhir atau tidak.

Dari 7 *task* pada Tabel 6.7 pemain akan ditugaskan untuk menyelesaikannya, dengan memperhatikan masing-masing *task* apakah pemain dapat menyelesaikannya atau tidak. Dari banyaknya *task* yang dapat pemain selesaikan dapat dicari nilai dari *Task Completion Rate* dengan menggunakan rumus:

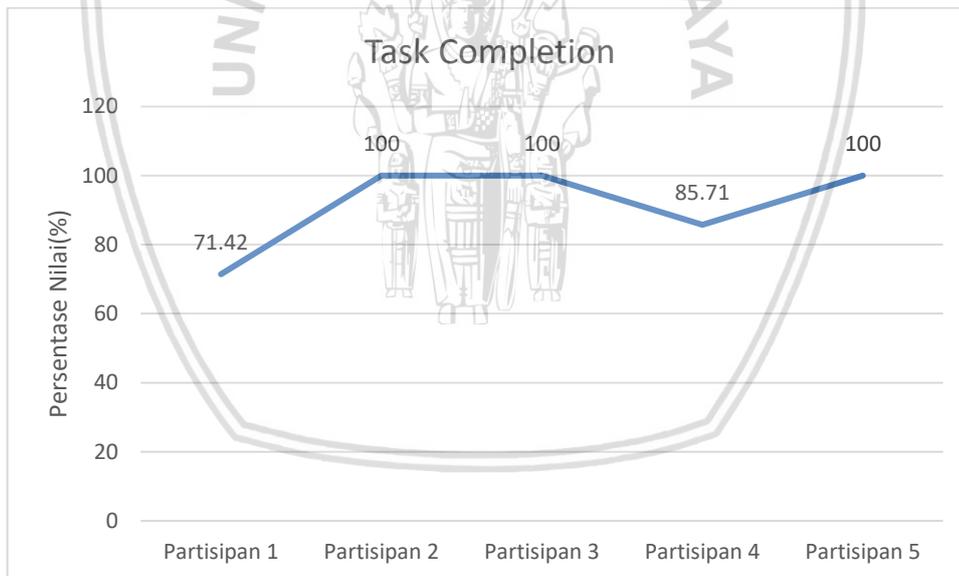
$$Task\ Completion\ Rate = \frac{\text{banyaknya task yang berhasil dilakukan}}{\text{total dari semua task}} * 100\%$$

Dari rumus diatas didapatkan hasil pada Tabel 6.8

**Tabel 6.8 Hasil *task completion* untuk masing-masing partisipan**

No	Partisipan	Jumlah <i>Task</i> Terselesaikan(dari 7)	Persen(%)
1	Partisipan 1	5	71.42857143
2	Partisipan 2	7	100
3	Partisipan 3	7	100
4	Partisipan 4	6	85.71428571
5	Partisipan 5	7	100
	Rata-rata	6.4	91.42857143

Tabel 6.8 menunjukkan bahwa penyelesaian tugas terendah (5 dari 7 *task*) terjadi pada partisipan pertama(p1) yaitu sebesar 71,43%. Kemudian disusul oleh partisipan ke-4 dengan nilai 85,7% (6 dari 7 *task*), sedangkan partisipan 2,3 dan 5 sama-sama memiliki *task completion rate* maksimal yaitu 100%. Dari masing-masing hasil *task completion rate* dari partisipan kemudian didapatkan rata-rata sebesar 91,43%.



**Gambar 6.3 Grafik *task completion***

Gambar 6.3 merupakan grafik dari *task completion rate*. Dapat dilihat bahwa nilai terendah dari *task completion* sebesar 71% yaitu oleh partisipan 1 dan nilai tertinggi ialah 100% oleh partisipan 2, 3, dan 5. Dapat dilihat juga, 3 dari 5 partisipan dapat menyelesaikan semua *task* yang diberikan kepadanya.

Terdapat beberapa alasan dari perolehan data yang telah didapatkan seperti:

- a) P1 tidak dapat menyelesaikan 2 tugas yang diberikan, 2 tugas tersebut ialah tugas untuk tidur pada Kasur yang terletak pada ruangan kamar dan

mengikuti ujian akhir yang berada pada ruangan ujian. Kegagalan pada tugas pertama yaitu tidur ialah dikarenakan pemain tidak dapat mendeteksi *collider* dari objek Kasur. Sehingga pemain tidak dapat berinteraksi(tidur) dengan objek tersebut. Sedangkan kegagalan pada tugas kedua ialah dikarenakan pemain tidak dapat atau kebingungan saat mencari ruangan ujian. Pemain mengira ujian akhir akan dilakukan di dalam kelas, sedangkan ujian akhir seharusnya dilaksanakan pada ruangan ujian. Pada akhirnya pemain mengikuti pembelajaran, bukan mengikuti ujian akhir. Kegagalan dalam mengikuti ujian akhir juga terjadi pada partisipan ke 4(p4).

- b) Hal yang sama juga terjadi pada p4, yang mana dia mengikuti pembelajaran bukan mengikuti ujian akhir. Kedua partisipan juga memberikan pendapat agar di dalam permainan lebih baik ditambahkan petunjuk yang lebih mengenai lokasi dan apa yang harus dia lakukan di dalam permainan.

Dari hasil presentase yang didapatkan pada pengujian ini kemudian dapat dicari nilai rata-ratanya, lalu didapatkan nilai rata-ratanya yaitu sebesar 91,4%. Nilai tersebut telah melebihi nilai standar dari *task completion rate* yaitu sebesar 78% (Shauro,2011). Dengan kata lain *task completion* untuk permainan ini telah baik memenuhi standar.

### 6.4 Pengujian System Usability Scale

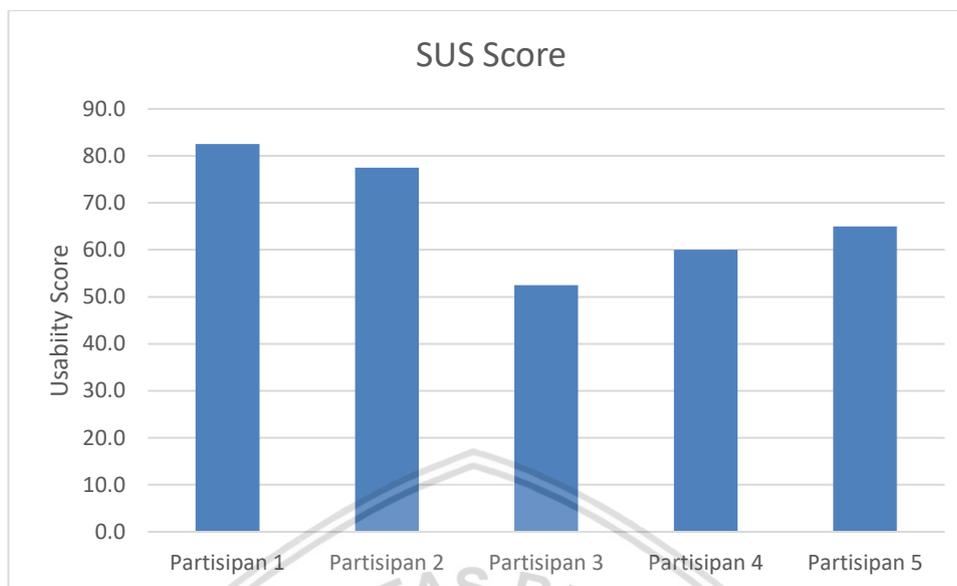
*System Usability Scale* (SUS) digunakan untuk menghitung tingkat kepuasan dari pengguna. SUS terdiri dari 10 pertanyaan dengan masing-masing pertanyaan yang memiliki skala dari 1 sampai 5 yang mana nilai sebesar 3 merupakan nilai normal atau netral. Hasil dari pengujian yang telah dilakukan dapat dilihat pada Tabel 6.9

Tabel 6.9 Hasil kuesioner SUS

No	Partisipan	Statement										SUS Score
		I	II	III	IV	V	VI	VII	VIII	IX	X	
1	P1	4	3	5	3	5	2	5	1	5	2	82.5
2	P2	5	3	2	3	5	1	4	1	5	2	77.5
3	P3	4	3	2	5	5	1	4	3	2	4	52.5
4	P4	3	3	4	4	4	2	4	3	4	3	60
5	P5	3	2	5	3	5	3	4	3	3	3	65
	<b>Rata-rata</b>	<b>3.8</b>	<b>2.8</b>	<b>3.6</b>	<b>3.6</b>	<b>4.8</b>	<b>1.8</b>	<b>4.2</b>	<b>2.2</b>	<b>3.8</b>	<b>2.8</b>	<b>67.5</b>

Tabel 6.9 Menunjukkan hasil dari kuesioner yang telah dilakukan kepada 5 partisipan. Pada tabel tersebut dapat dilihat hasil skor SUS tertinggi sebesar 82,5 yaitu oleh partisipan 1 (p1). Sedangkan skor SUS terendah sebesar 52,5 oleh partisipan 3 (p3). Dari total skor SUS yang didapatkan dapat kemudian dihitung rata-ratanya.





**Gambar 6.4 Grafik Hasil skor SUS**

Gambar 6.4 merupakan grafik hasil skor SUS yang telah dihitung. Partisipan 1 memiliki skor sebesar 82,5, partisipan 2 sebesar 77,5, partisipan 3 sebesar 52,5, partisipan 4 sebesar 60 dan partisipan 5 sebesar 65,0. Serta dapat dilihat bahwa partisipan 1 memiliki skor tertinggi sedangkan skor terendah dimiliki oleh partisipan 3. Berdasarkan dari Tabel 6.9, rata-rata yang didapatkan dari nilai SUS ialah sebesar 67,5. Nilai tersebut tergolong di bawah rata-rata skor SUS yang mana standar untuk nilai SUS yang normal ialah sebesar 68 (Thomas, 2015).

Hal yang menyebabkan rendahnya nilai SUS pada partisipan 3 ialah dikarenakan SUS *score* yang diisi olehnya memiliki nilai yang kecil pada *statement* 3 yaitu “apakah aplikasi ini mudah untuk digunakan?”. Pada *statement* 3 sendiri jika nilai semakin kecil maka akan mengakibatkan nilai SUS dari aplikasi akan mengecil. Pada kasus ini, partisipan 3 mengisi *statement* tersebut dengan nilai 2 atau “tidak setuju”. Pada saat pengujiannya sendiri, partisipan 3 mengalami kendala pada saat ingin melakukan masukan (membentuk *gesture*) dengan alat *Myo Armband*. Masukan *Myo Armband* pada saat pengujian berlangsung menjadi kurang responsive. Ketika partisipan 3 membentuk sebuah *gesture* hal yang terjadi ialah masukan tersebut terkadang terdeteksi dan terkadang tidak terdeteksi. Sehingga pada saat pengujiannya, partisipan 3 meminta bantuan untuk mengoperasikan permainan ini. Hal tersebut juga mengakibatkan partisipan 3 mengisi *statement* 4 yaitu “Saya berfikir jika saya membutuhkan bantuan ketika menggunakan aplikasi ini” dengan nilai 5 (Sangat setuju) yang mana semakin besar nilai *statement* tersebut maka akan mengakibatkan nilai SUS menjadi kecil. Hal tersebut kemungkinan juga menjadi satu alasan kenapa partisipan 3 memberikan nilai sebesar 4 pada *statement* ke 10 yaitu “Saya perlu mempelajari banyak hal sebelum dapat menggunakan aplikasi ini”. Tabel 6.9 menunjukkan bahwa partisipan 3 memberikan nilai sebesar 2 pada *statement* 3, nilai 5 pada *statement*

ke 4 serta nilai 4 pada *statement* 10. Oleh karenanya, nilai skor SUS dari partisipan 3 menjadi kecil.

Berdasarkan hasil angket yang diisi oleh partisipan 4 pada Tabel 6.9, hal yang membuat nilai SUS dari partisipan 4 bernilai rendah ialah karena partisipan 4 merasa bahwa ia masih perlu bantuan penguji saat bermain permainan ini (*statement* 4 bernilai 4). Hal tersebut disebabkan oleh alasan yang sama dengan partisipan 3, yaitu *Myo Armband* kurang responsif saat dipakai. Masukan dari *gesture* dari partisipan terkadang terdeteksi dan terkadang tidak. Oleh karena itu, partisipan meminta bantuan kepada penguji saat memainkan permainan ini.

Berdasarkan Tabel 6.9 dari partisipan 5, hal yang membuat nilai SUS dari partisipan 5 kurang dari 68 adalah dikarenakan sebagian besar nilai angket dari partisipan 5 ialah 3. Hal tersebut kemungkinan dapat disebabkan karena partisipasi masih awam ketika mengikuti pengujian ini.



## BAB 7 PENUTUP

### 7.1 Kesimpulan

Berdasarkan hasil dari penelitian yang telah dilakukan dari proses perancangan sampai dengan pengujian, dapat diambil beberapa kesimpulan antara lain:

1. Pengimplementasian permainan edukasi pengenalan *chord* gitar berbasis *Myo* dan *Virtual Reality* dapat diwujudkan dengan menggunakan Unity3D sebagai *game engine*, dan model perancangan MDA *Framework* sebagai metode pendekatan perancangan permainan. Pengimplementasian VR pada permainan dapat diwujudkan dengan menggunakan *library* trinusVR yang ada pada Unity3D *store*. Serta untuk mengimplementasikan *Myo* sebagai masukan permainan dapat menggunakan *library* yang telah disediakan oleh pengembang *Myo*.
2. Berdasarkan hasil dari pengujian *black box* dan *task completion* dapat ditarik kesimpulan bahwa permainan ini telah berjalan dengan benar dan memiliki tingkat keefektifan yang baik. Dari hasil pengujian *blackbox*, didapatkan hasil valid 100 persen. Dari hasil tersebut menandakan bahwa semua fungsional sistem permainan ini telah berjalan dengan benar. Dari pengujian *task completion*, didapatkan hasil sebesar 91,43% dengan standar *task completion* yang baik sebesar 78%. Sehingga dapat dikatakan bahwa tingkat keefektifan dari sistem ini sudah baik. Namun, pada pengujian SUS didapatkan hasil rata-rata sebesar 67,5 dengan standar SUS yang baik sebesar 68. Dari hasil SUS tersebut menandakan bahwa permainan ini belum mencapai standar tingkat kepuasan pemain sehingga perlu adanya peningkatan lebih lanjut.
3. Dari hasil pengujian *pre-test* dan *post-test*, telah didapatkan hasil nilai rata-rata *pre-test* sebesar 70 dan *post-test* sebesar 90. Hasil tersebut menggambarkan bahwa terdapat peningkatan nilai pemain sebelum dan sesudah memainkan permainan ini yang mana menandakan bahwa permainan ini telah sesuai dengan harapan sebagai media pembelajaran interaktif.

### 7.2 Saran

Dari penelitian yang telah dilakukan, didapatkan saran agar permainan dapat dikembangkan menjadi lebih baik:

1. Penambahan mode pembelajaran dan jenis pendekatan pembelajaran akan lebih baik jika ditambahkan agar menambah varietas jenis pembelajaran agar pemain tidak cepat merasa bosan karena mempelajari hal sama berulang-ulang.
2. Respon dari alat *Myo* perlu ditingkatkan karena terkadang *gesture* tangan pemain tidak terdeteksi dengan benar.

3. Perlu ditambahkan UI atau petunjuk dalam permainan mengenai lokasi ataupun *progress* dari permainan untuk menghindari kebingungan dari pemain.
4. Ukuran UI perlu dioptimisasikan agar terlihat jelas terlihat oleh pemain. Akan tetapi perlu dicatat karena terlalu banyak UI yang ditampilkan pada *screen* layar permainan akan berdampak mengurangi tingkat *immersive* dari permainan.
5. Percakapan guru dapat ditambahkan dengan suara asli untuk menambah *immersive* dari permainan



## DAFTAR PUSTAKA

- Anopas, D., Wongsawat, Y., 2014. *Virtual reality game for memory skills enhancement based on QEEG*. IEE Explore. Tersedia di: <<http://ieeexplore.ieee.org/document/7017399/>> [diakses 7 Februari 2018]
- Arjaka, Sugiyono., n.d. *Media Elektronik Berbasis Komputer Sebagai Media Dalam Proses Pembelajaran*. [online] Tersedia di: <http://jurnal-online.um.ac.id/data/artikel/artikelCE78C5B85EB8B1688FF2FC2D9760552D.pdf> [diakses 15 April 2018]
- Benward, Shaker, 2003. *Music: In Theory and Practice : Spiral*. New York: McGraw-Hill College.
- Brathwaite, B., Schreiber, I., 2009. *Challenges For Game Designers*. Boston: Course Technology PTR.
- Hanna, P., n.d. *Java Games Programming*. Belfsat: Queen's University.
- Heinich, Robert, 1982. *Instructional Media and New Technology of Instruction*. New York: John Wiley and Sons.
- Hunicke, R., Leblanc, M., Zubek, R., 2004. *MDA: A Formal Approach to Game Design and Game Research*. Research Gate. Tersedia di: <[https://www.researchgate.net/publication/228884866\\_MDA\\_A\\_Formal\\_Approach\\_to\\_Game\\_Design\\_and\\_Game\\_Research](https://www.researchgate.net/publication/228884866_MDA_A_Formal_Approach_to_Game_Design_and_Game_Research)> [diakses 17 Februari 2018]
- Insight Assesment. 2018. *How can I compare pretest to posttest scores in a small sample?*. [online] Tersedia di: <<https://insightassessment.com/FAQ/FAQs-Pretest-Posttest-Design/How-can-I-compare-pretest-to-posttest-scores-in-a-small-sample>>
- Kostolny, J., Bohacik, J., 2017. *Digital games in education and their development*. IEE Explore. Tersedia di: <<http://ieeexplore.ieee.org/document/8102494/>> [diakses 7 Februari 2018]
- Khan, Mohd, Ehmer., Khan, Farmeena., 2012. *A Comparative Study of White Box, Black Box and Grey Box Testing Techniques*. [pdf] International Journal of Advanced Computer Science and Applications. Tersedia di: <<http://thesai.org/Downloads/Volume3No6/Paper%203-A%20Comparative%20Study%20of%20White%20Box,%20Black%20Box%20and%20Grey%20Box%20Testing%20Techniques.pdf>> [diakses 2 Desember 2018]
- Krishnan, K, S., Saha, A., Ramachandran, S., Kumar, Shitij., 2017. *Recognition of human arm gestures using Myo armband for the game of hand cricket*. IEE Explore. Tersedia di: <<http://ieeexplore.ieee.org/document/8250154/>> [diakses 7 Februari 2018]
- Leow, F., 2014. *Interactive Multimedia Learning: Innovating Classroom Education In A Malaysian University*. TOJET. Tersedia di: <<http://tojet.net/articles/v13i2/13211.pdf>> [diakses 9 Februari 2018]



- Lozancic, Ana., 2017. *White Box Testing and Black Box Testing*. [online] Tersedia di: < <https://gauss-development.com/white-box-testing-black-box-testing/>> [diakses 2 Desember 2018]
- Nielsen, Jakob., 2009. *Why You Only Need to Test with 5 Users*. Nielsen Norman Group [online] Tersedia di: < <https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/>> [diakses 5 Desember 2018]
- Parisi, T., 2015. *Learning Virtual Reality Developing Immersive Experiences and Applications for Desktop, Web and Mobile*. Sebastopol: O'Really Media Inc.
- Sauro, Jeff., 2011. *10 Things To Know About Completion Rates*. [online] Tersedia di: < <https://measuringu.com/completion-rates/>> [diakses 4 Desember 2018]
- Sergius, L., 2018. *Panduan Super Belajar Gitar untuk Pemula*. [e-book] Tersedia di: < <http://www.slashkygitaris.com/2018/01/ebook-gratis-belajar-gitar.html>> [diakses 17 Februari 2018]
- Schreiber, I., *Game Design Concept*. [e-book] Tersedia di: < <http://gamedesignconcepts.pbworks.com/f/Game+Design+Concepts+-+An+experiment+in+game+design+and+teaching.pdf>> [diakses 17 Februari 2018]
- Wibawanto, W., 2017. *Desain dan Pemrograman Multimedia Pembelajaran Interaktif*. Jember: Penerbit Cerdas Ulet Kreatif.

