

BAB 3 METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Metode Penelitian yang dilakukan pada penelitian ini adalah sebagai berikut.

1. Studi Literatur

Pada tahap ini, dilakukan pengumpulan informasi yang dibutuhkan dalam penelitian ini. Informasi yang dicari berkaitan dengan Muay Thai, *game*, *virtual reality*, algoritma *sattolo shuffle*, *Game User Experience Satisfaction Scale* (GUESS), *waterfall model*, dan *black box testing*. Informasi tersebut akan digunakan sebagai acuan dalam pembangunan *game* yang akan dibuat pada penelitian ini.

2. Perancangan Sistem

Pada tahap perancangan sistem, dilakukan perancangan *game* yang ingin dibangun berdasarkan analisis kebutuhan dari penelitian, yaitu perancangan alur *game*, tampilan atau UI dari *game*, aset yang diperlukan untuk kebutuhan *game* yang akan dibuat, serta implementasi algoritma *sattolo shuffle* yang akan diterapkan pada *game*.

3. Pembuatan Aplikasi

Tahap ini merupakan proses pembangunan *game virtual reality* Pengenalan Muay Thai menggunakan *waterfall model*. Kegiatan yang dilakukan pada tahap ini adalah perancangan program utama, mengimplementasikan tampilan atau UI dari *game*, serta penggunaan aset yang telah dicari berdasarkan rancangan yang telah dilakukan pada tahap perancangan sistem. Lalu dilakukan juga implementasi dari gerakan Muay Thai yang telah ditentukan yaitu gerakan pukulan yang terdiri dari *jab*, *cross / straight*, *hook*, dan *uppercut* yang telah digabungkan dengan algoritma *sattolo shuffle* untuk melakukan pengacakan gerakan yang akan ditampilkan. Proses pembangunan *game* tersebut menggunakan aplikasi pembangun *game* yaitu Unity.

4. Pengujian

Pada tahap ini, *game* yang dihasilkan dari tahap pembuatan aplikasi diuji terlebih dahulu kualitas serta fungsionalitas dari *game* yang telah dibangun

menggunakan *black box testing*. Fungsionalitas terdiri atas fungsi-fungsi dasar yang dapat dilakukan oleh pengguna saat memainkan *game*, seperti saat pemain berada pada *main menu*, *mode selection*, *learn mode* dan *battle mode*. Lalu pengujian berikutnya adalah pengujian *game* yang dilakukan oleh pemain serta pengisian kusioner yang dibangun berdasarkan pemodelan GUESS untuk mendapatkan tingkat penerimaan dari *game* pemain mainkan.

5. Evaluasi

Tahap ini merupakan proses analisis dari hasil pengujian yang telah dilakukan sebelumnya. Hasil dari analisis yang didapatkan menggunakan pemodelan GUESS akan digunakan untuk mendapatkan nilai penerimaan dari pemain dan akan dijadikan sebagai kesimpulan dari penggunaan teknologi *virtual reality* dalam sebuah *game* terhadap pengenalan bela diri Muay Thai.

3.2 Analisa dan Perancangan Sistem

Dalam penelitian ini, aplikasi yang dibuat berupa *game* pembelajaran. *Game* ini memperkenalkan dasar dari gerakan bela diri Muay Thai dengan mengenalkan dasar gerakan dengan 2 mode, mode pertama adalah mengenalkan dasar gerakan bela diri Muay Thai dengan menampilkan nama dan deskripsi dari gerakan tersebut, lalu mode kedua adalah pengenalan gerakan dasar Muay Thai dengan melakukan pertarungan langsung dengan musuh yang telah disediakan dengan 3 tingkatan kesulitan yang dapat dipilih oleh pemain. Pemilihan menu dapat dilakukan pemain dengan menekan tombol "B" pada *controller*. Pada mode pertarungan, pemain diminta untuk melakukan gerakan bela diri Muay Thai berdasarkan gerakan yang diminta oleh sistem kepada pemain dengan menekan tombol tertentu, gerakan yang diminta oleh sistem tersebut akan diacak menggunakan algoritma *sattolo shuffle*.

3.2.1 Spesifikasi Game

Game yang dibangun merupakan *game* pengenalan Muay Thai menggunakan *virtual reality*. Pada *game* ini pemain dapat mempelajari gerakan dasar pukulan Muay Thai melalui contoh gerakan dengan diberikan nama serta deskripsi dari gerakan tersebut, maupun dilakukan dengan menggerakkan karakter untuk melakukan gerakan dasar Muay Thai dalam pertarungan langsung dengan musuh. Berikut merupakan *formal element* yang digunakan dalam membangun *game* pengenalan Muay Thai menggunakan *virtual reality*.

1. *Players*

(a) *Single Player*

Game ini hanya dapat dimainkan oleh seorang pemain.

(b) *Player vs Game*

Pemain memenangkan permainan dengan menyelesaikan rintangan dengan batasan yang diberikan dalam *game*.

2. *Objectives*

Pemain menyelesaikan *game* dengan waktu secepat mungkin agar mendapatkan hasil terbaik dalam permainan pertarungan, yaitu mendapatkan 3 bintang.

3. *Procedures*

(a) Saat *game* pertama kali dijalankan, maka pemain akan dibawa menuju *Main Menu*, pada main menu pemain dapat memilih *start* untuk menuju *mode selection*, *setting* untuk menghidupkan atau mematikan suara, serta *exit game* untuk keluar dari *game* yang dimainkan.

(b) Pada menu *mode selection*, pemain diminta untuk memilih menu *Learn* untuk mempelajari teori dasar dari gerakan dasar pukulan bela diri Muay Thai. Selain itu pemain juga dapat memilih menu *Battle* untuk melakukan pertarungan dengan musuh. Lalu pemain juga dapat memilih menu *Return Main Menu* untuk kembali ke *Main Menu*.

(c) Dalam mode *Learn*, pemain dapat memilih diantara 4 karakter untuk melakukan gerakan dasar pukulan bela diri Muay Thai, lalu saat karakter dipilih, pemain dapat menggerakkan karakter tersebut untuk melakukan gerakan dasar Muay Thai, dan memutar karakter untuk melihat gerakan secara keseluruhan. Selain itu pemain juga dapat melihat deskripsi dari gerakan yang dilakukan oleh karakter tersebut. Pemain dapat keluar dari mode *Learn* ini dengan menekan tombol "C" saat tidak memilih karakter apapun, lalu memilih *Exit Mode* untuk kembali ke menu *Mode Selection*.

(d) Dalam mode *Battle*, pemain diminta untuk memilih tingkat kesulitan terlebih dahulu. Lalu setelah tingkat kesulitan tersebut dipilih, maka pemain akan melihat musuh yang akan dilawan. Pada mode ini, pemain diminta untuk menyelesaikan runtutan gerakan yang telah diacak

oleh sistem menggunakan algoritma *sattolo shuffle*, gerakan tersebut dapat diselesaikan dengan pemain menekan tombol sesuai dengan gerakan yang diminta oleh sistem. Setiap gerakan yang salah maka pemain akan diberikan sanksi berupa pengurangan waktu dari sistem, selain itu untuk mendapatkan waktu tambahan, pemain diminta untuk menyelesaikan satu kali perulangan gerakan terlebih dahulu. Waktu yang disediakan, pengurangan serta penambahan waktu pemain ditambah dan dikurang berdasarkan tingkat kesulitan yang dipilih oleh pemain. Untuk mendapatkan bintang 3 (nilai tertinggi), pemain diminta untuk menyelesaikan permainan dalam kurun waktu tertentu. Permainan akan selesai apabila pemain dapat menyelesaikan seluruh gerakan yang telah diberikan oleh sistem, ataupun waktu yang dimiliki oleh pemain habis. Saat permainan berlangsung, pemain dapat melakukan *pause* dengan menggerakkan analog ke arah atas atau bawah, lalu pemain diberikan 3 pilihan berupa *Continue* untuk melanjutkan permainan, *Restart* untuk mengulang kembali permainan dengan tingkat kesulitan yang sama, dan *Exit Mode* untuk kembali ke menu pemilihan tingkat kesulitan permainan. Selain itu, saat permainan selesai, pemain akan diperlihatkan menu yang menampilkan hasil dari permainan pada tingkat kesulitan tersebut, pada menu ini pemain diberikan 2 pilihan berupa *Restart* dan *Exit Mode*.

4. Rules

Pada mode *Learn* dan *Battle*, permainan tidak akan dimulai sebelum pemain membaca cara melakukan interaksi dalam mode tersebut. Selain itu, pada mode *Battle*, pemain diharuskan untuk memilih tingkat kesulitan terlebih dahulu sebelum dapat memainkan permainan. Dalam mode *Battle*, pemain akan dibatasi oleh waktu yang diberikan dalam permainan. Untuk mendapatkan waktu tambahan, pemain harus menyelesaikan satu kali perulangan gerakan terlebih dahulu. Namun ketika pemain melakukan gerakan yang salah, maka waktu yang dimiliki oleh pemain akan dikurangkan.

5. *Resources*

Pemilihan menu dilakukan dengan menggerakkan kepala menuju menu yang diinginkan, kemudian menggunakan *controller* untuk berinteraksi dalam *game*.

6. *Conflict*

Pemain harus menggerakkan karakter sesuai dengan gerakan yang diminta oleh sistem secara akurat agar dapat menyelesaikan permainan dengan sempurna (mendapatkan 3 bintang).

7. *Boundaries*

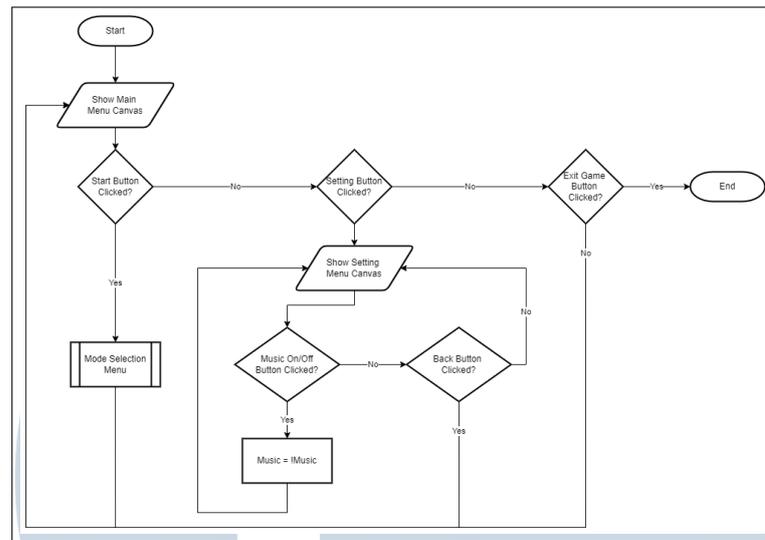
Pemain hanya dapat menggerakkan kepala untuk melihat sekitar dan memilih menu atau karakter lain dan tidak dapat menggerakkan karakter yang digunakan.

8. *Outcome*

Pada saat mode *Battle* selesai, sistem akan menunjukkan hasil yang didapatkan oleh pemain, yang diantaranya adalah menang dengan mendapatkan 1 sampai 3 bintang dan kalah dikarenakan kehabisan waktu.

3.2.2 Flowchart Main Menu

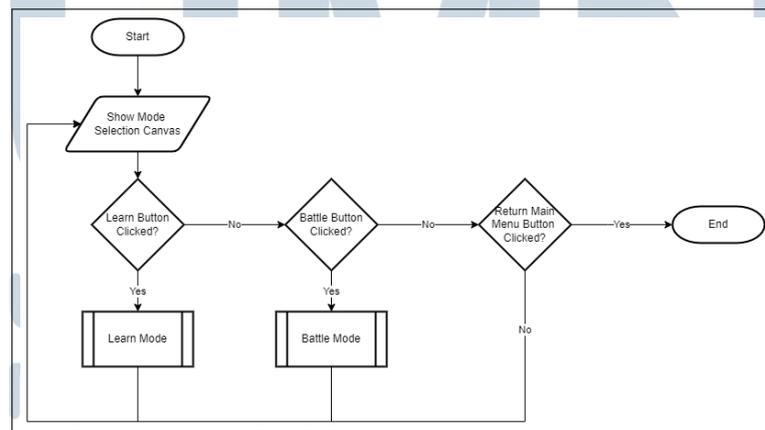
Pada saat *game* pertama kali dijalankan, maka *game* akan menampilkan main menu kepada pemain. *Main menu* ini berisikan menu *Start*, *Setting*, dan *Exit Game*. Pada menu *Start*, menu ini berfungsi untuk mengawali permainan, saat permainan dimulai maka akan menjalankan *Mode Selection*. Kemudian terdapat menu *Setting* yang berfungsi untuk menghidupkan atau mematikan musik yang terdapat pada *game*. Lalu yang terakhir adalah menu *Exit Game* yang berfungsi untuk menutup *game* yang dimainkan. *Flowchart* dari *main menu* terlihat seperti pada Gambar 3.1.



Gambar 3.1. *Flowchart Main Menu*

3.2.3 Flowchart Mode Selection

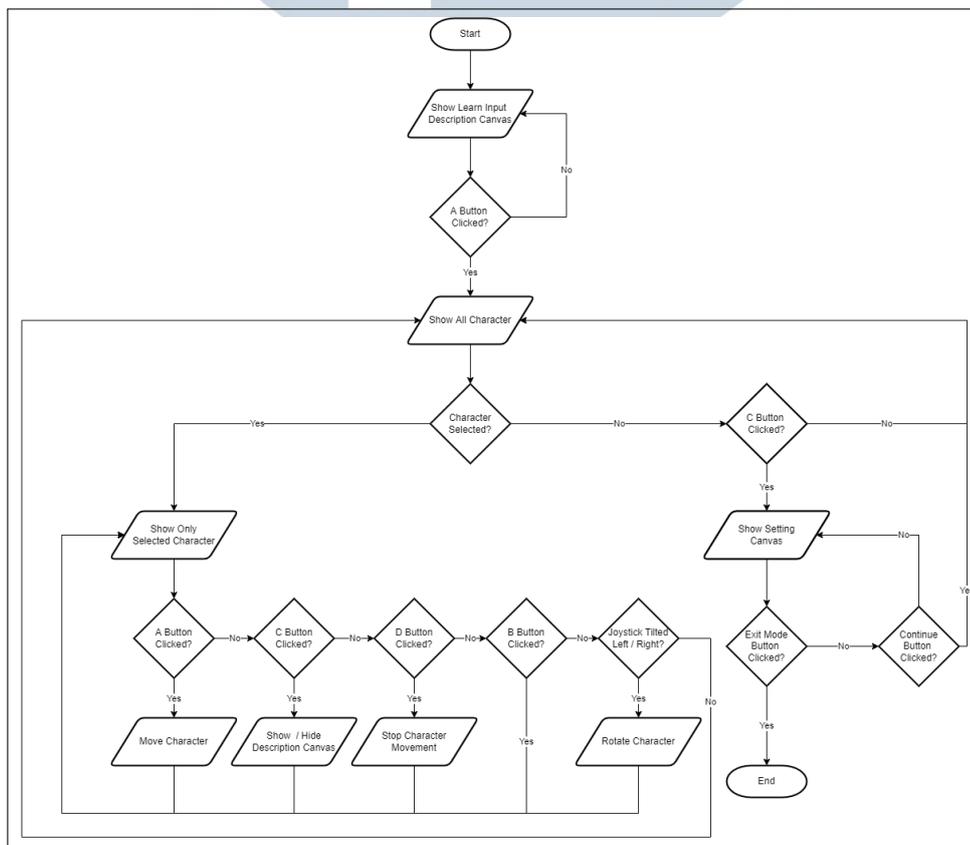
Setelah *game* dimulai dari main menu, maka akan ditampilkan menu *Mode Selection*. Pada menu ini ditampilkan 3 pilihan, yaitu menu *Learn*, *Battle* dan *Return to Main*. Saat menu *Learn* dipilih, maka akan dijalankan *Learn Mode*. Lalu, saat menu *Battle* dipilih, maka akan menjalankan *Battle Mode*. Selain itu, menu *Return to Main* berfungsi untuk kembali menuju main menu. *Flowchart mode selection* terlihat seperti pada Gambar 3.2.



Gambar 3.2. *Flowchart Mode Selection*

3.2.4 Flowchart Learn Mode

Learn Mode merupakan mode yang digunakan untuk mempelajari gerakan dasar pukulan yang terdapat pada Muay Thai. Pada saat *Learn Mode* dijalankan, maka akan ditampilkan deskripsi dari *input* yang dapat dilakukan oleh user. Setelah deksripsi tersebut ditutup, maka pemain akan ditunjukkan 4 karakter yang dapat melakukan salah satu gerakan dari seluruh gerakan pukulan dasar. Saat karakter tersebut dipilih, maka karakter yang tidak dipilih akan disembunyikan. Setelah karakter dipilih, maka pemain dapat melihat gerakan dasar pukulan yang dilakukan oleh karakter tersebut, dan melihat dari deskripsi gerakan yang dilakukan, seperti nama gerakan serta penjelasan dari gerakan tersebut. Kemudian saat pemain tidak memilih salah satu karakter yang disediakan, pemain dapat membuka menu setting yang terdiri dari 2 pilihan, yaitu *Continue* dan *Exit Mode*. *Continue* berfungsi untuk melanjutkan *Learn Mode* yang sedang dimainkan, dan *Exit Mode* berfungsi untuk membawa pemain kembali menuju *Mode Selection*. *Flowchart learn mode* terlihat seperti pada Gambar 3.3



Gambar 3.3. *Flowchart Learn Mode*

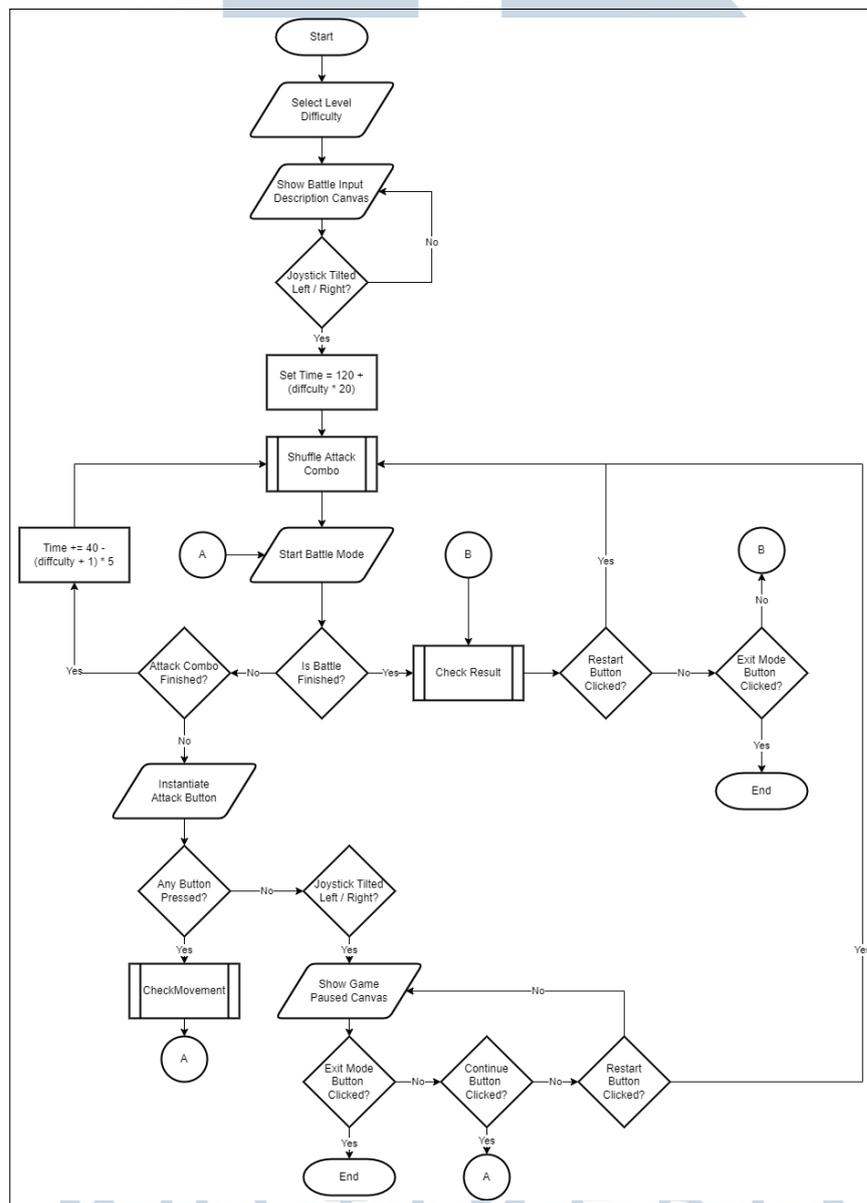
3.2.5 Flowchart Battle Mode

Battle Mode merupakan mode yang digunakan untuk mempelajari gerakan dasar pukulan dengan melawan musuh yang telah disediakan. Pada mode ini, pemain akan diminta untuk melakukan gerakan dasar pukulan berdasarkan gerakan yang diminta oleh sistem. Saat mode ini pertama kali dijalankan maka pemain akan diminta untuk memilih tingkat kesulitan dari mode yang ingin dimainkan. Setelah pemilihan tingkat kesulitan sudah selesai, maka deskripsi *input* yang dapat dilakukan pada mode tersebut akan ditampilkan kepada pemain. Ketika deksripsi tersebut ditutup, maka sistem akan melakukan pengacakan gerakan yang akan diselesaikan oleh pemain terlebih dahulu di *Shuffle Attack Combo*. Setelah pengacakan selesai, maka mode pertarungan akan dimulai. Saat permainan dimulai, gerakan yang harus dilakukan oleh pemain akan dimunculkan terlebih dahulu, lalu pemain akan diminta untuk melakukan gerakan yang diminta. Saat pemain melakukan gerakan maka akan dilakukan *Check Movement*. Terdapat 5 jenis gerakan yang terdapat pada permainan ini, yaitu *jab*, *hook right*, *hook left*, *cross / straight* dan *uppercut*. Pemain akan diminta untuk melakukan 5 gerakan tersebut sesuai dengan deskripsi yang telah dijelaskan sebelumnya, bila pemain melakukan gerakan yang salah maka pemain akan dikenakan sanksi pengurangan waktu dan musuh akan menyerang pemain, sedangkan untuk mendapatkan tambahan waktu pemain diharuskan untuk menyelesaikan satu set gerakan terlebih dahulu. Penambahan waktu berkurang sesuai dengan tingkat kesulitan yang dimainkan dengan perhitungan sebagai berikut:

$$Time = Time + (40 - (difficulty + 1) * 5) \quad (3.1)$$

Berdasarkan perhitungan 3.1, waktu tambahan yang akan diberikan adalah sebanyak 40 dikurangi dengan tingkat kesulitan ditambah 1 kemudian di kalikan dengan 5. Hal ini rancang supaya semakin sulit kesulitan, maka semakin sedikit waktu tambahan yang diberikan. Selama permainan berlangsung pemain juga dapat melakukan *pause game*, pada menu *pause game* ini terdapat 3 pilihan yang dapat dipilih oleh pemain, yaitu *Continue*, *Restart* dan *Exit Mode*. *Continue* digunakan untuk melanjutkan *game* yang telah *dipause* oleh pemain, lalu *Restart* digunakan untuk mengulang *Battle Mode* dengan tingkat kesulitan yang sama, dan yang terakhir adalah *Exit Mode* yang digunakan untuk kembali ke *Mode Selection*. Lalu, saat pemain menyelesaikan atau gagal menyelesaikan permainan, maka hasil

permainan akan cek melalui proses *Check Result*. Pada menu hasil permainan ini, pemain diberikan 2 pilihan yaitu *Restart* dan *Exit Mode* dengan fungsi yang sama seperti saat pemain melakukan *pause game*. *Flowchart battle mode* terlihat seperti pada Gambar 3.4.

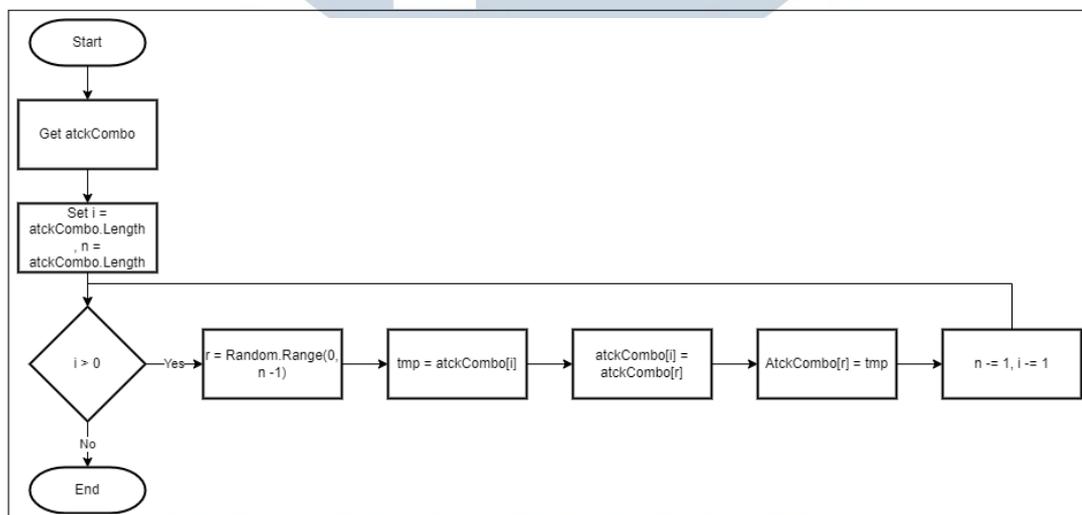


Gambar 3.4. *Flowchart Battle Mode*

MULTIMEDIA
NUSANTARA

3.2.6 Flowchart Shuffle Attack Combo

Shuffle Attack Combo merupakan proses pengacakan gerakan yang dilakukan pemain menggunakan algoritma pengacakan *sattolo shuffle*. Pengacakan ini berfungsi untuk mengacak gerakan yang perlu dilakukan oleh pemain untuk menyelesaikan permainan *Battle Mode*. Pada pengacakan ini, sistem akan mengambil *array* *atkCombo* yang berisikan daftar gerakan yang harus dilakukan oleh pemain, kemudian akan diambil panjang *array* tersebut dan ditampung ke dalam variabel *n* dan *i*. Kemudian dilanjutkan dengan mencari nilai random yang dimasukkan ke variabel *r* dan akan dijadikan target untuk dilakukan pertukaran dengan posisi *array* *i*. Nilai random yang akan dicari merupakan nilai dari 0 sampai dengan nilai $(n - 1)$, lalu nilai *i* dan *n* akan dikurang dengan 1 saat pertukaran telah selesai dilakukan. Proses perulangan pengacakan tersebut dilakukan sebanyak $(i - 1)$, setelah perulangan selesai dilakukan maka *array* tersebut akan menghasilkan kombinasi gerakan yang berbeda dengan sebelumnya. *Flowchart shuffle attack combo* terlihat seperti pada Gambar 3.5.



Gambar 3.5. *Flowchart Shuffle Attack Combo*

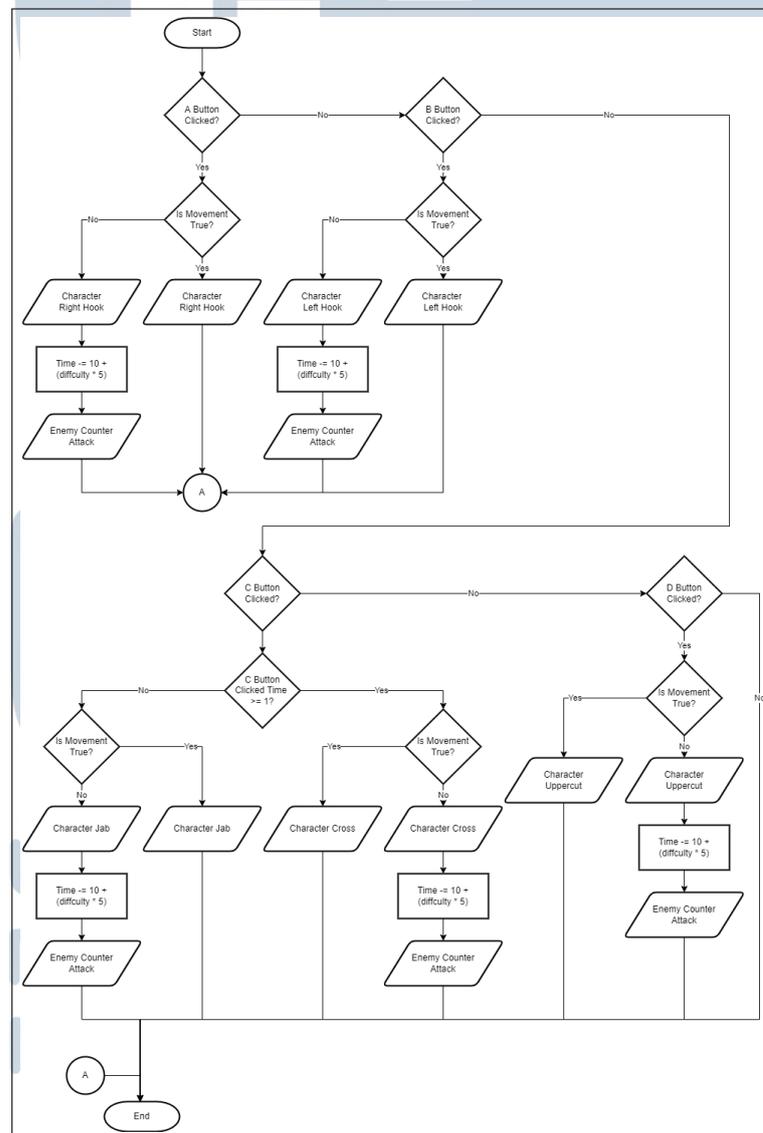
3.2.7 Flowchart Check Movement

Check Movement merupakan proses pengecekan dari *input* gerakan yang dilakukan oleh pemain. Bila *input* yang diberikan pemain benar, maka karakter pemain akan menyerang musuh saja. Namun, bila *input* yang diberikan salah, maka setelah karakter pemain menyerang, maka karakter musuh akan kembali menyerang

karakter pemain, dan pemain akan diberikan sanksi yaitu pengurangan waktu tersisa dalam permainan. Perhitungan dari pengurangan waktu bertambah berdasarkan tingkat kesulitan yang dihitung sesuai dengan perhitungan berikut:

$$Time = Time - (10 + (difficulty * 5)) \quad (3.2)$$

Berdasarkan perhitungan 3.2, pengurangan yang dilakukan akan meningkat berdasarkan tingkat kesulitan yang diambil pemain, yaitu 10 ditambahkan dengan tingkat kesulitan dikalikan dengan 5. Pada Gambar 3.6, merupakan gambaran dari *Flowchart check movement*.



Gambar 3.6. *Flowchart Check Movement*

3.2.8 Flowchart Check Result

Check Result merupakan fungsi yang digunakan untuk melihat hasil akhir dari permainan yang dilakukan oleh pemain. Baik menang maupun kalah yang dilihat dari sisa waktu yang ada. Jika waktu sama dengan 0, maka pemain akan ditampilkan hasil bahwa mereka telah gagal menyelesaikan permainan. Kemudian bila waktu yang dimiliki lebih dari 0, maka akan dicek kembali apakah waktu lebih dari ketentuan tertentu untuk menentukan bintang yang didapatkan. Perhitungan yang dilakukan untuk menghitung bintang adalah sebagai berikut:

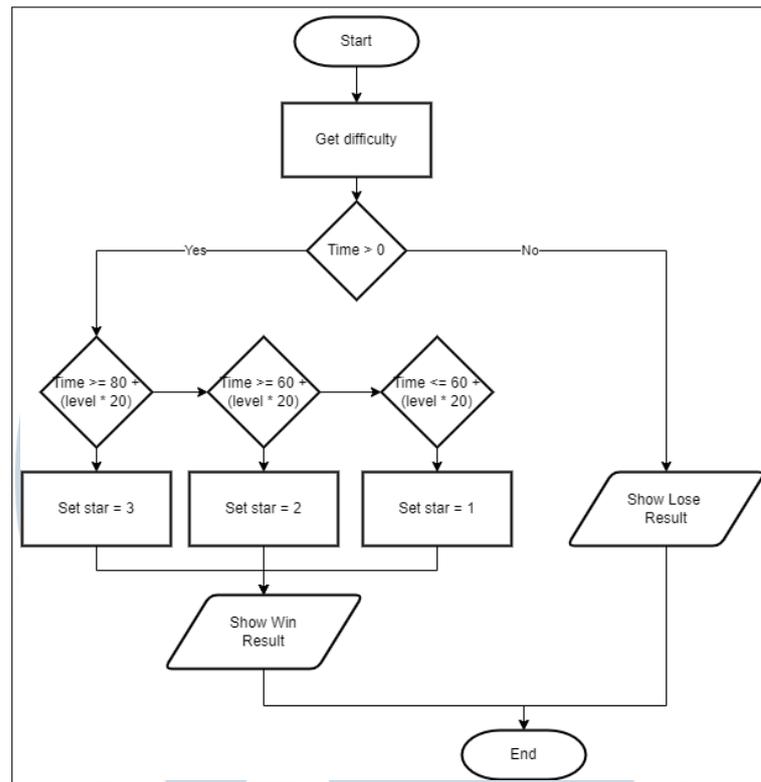
$$Time \geq 80 + (difficulty * 20) \quad (3.3)$$

$$Time \geq 60 + (difficulty * 20) \quad (3.4)$$

$$Time \leq 60 + (difficulty * 20) \quad (3.5)$$

Pada perhitungan 3.3, merupakan perhitungan yang dilakukan untuk mendapatkan 3 bintang dengan mengecek waktu tersisa lebih besar dari 80 ditambah dengan tingkat kesulitan dikalikan dengan 20. Kemudian pada perhitungan 3.4, merupakan perhitungan yang dilakukan untuk mendapatkan 2 bintang dengan mengecek waktu tersisa lebih besar dari 60 ditambah dengan tingkat kesulitan dikalikan dengan 20. Dan yang terakhir yaitu perhitungan 3.5 yang merupakan perhitungan untuk mendapatkan 1 bintang dengan mengecek waktu tersisa lebih kecil dari 60 ditambah dengan tingkat kesulitan dikalikan dengan 20. Waktu yang dibutuhkan untuk menyelesaikan permainan dengan hasil tertentu akan bertambah berdasarkan tingkat kesulitan yang dimainkan. Kemudian hasil tersebut akan ditampilkan kepada pemain. *Flowchart check result* terlihat seperti pada Gambar 3.7.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



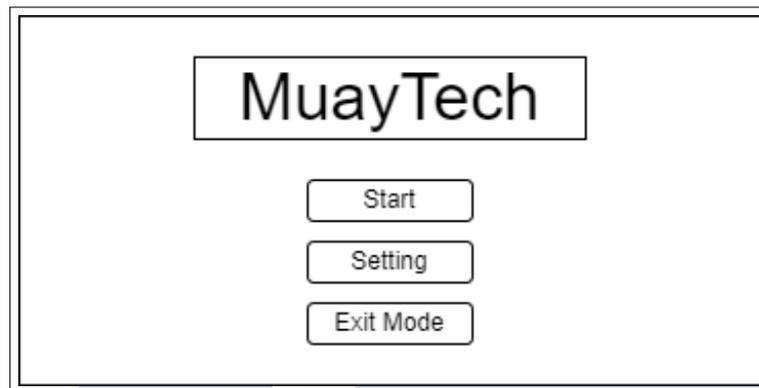
Gambar 3.7. Flowchart Check Result

3.3 Perancangan Tampilan / UI Game

Pada tahap ini, dilakukan perancangan dari tampilan / UI yang digunakan dalam *game*. Tampilan / UI yang terdapat pada *game* ini terdiri atas *Home*, *Setting*, *Mode Selection*, *Difficulty Selection*, *Learn Mode*, *Battle Mode*, *Pause*, dan *Result*. Perancangan yang dilakukan dalam perancangan *game* adalah sebagai berikut.

3.3.1 Rancangan Tampilan / UI Home

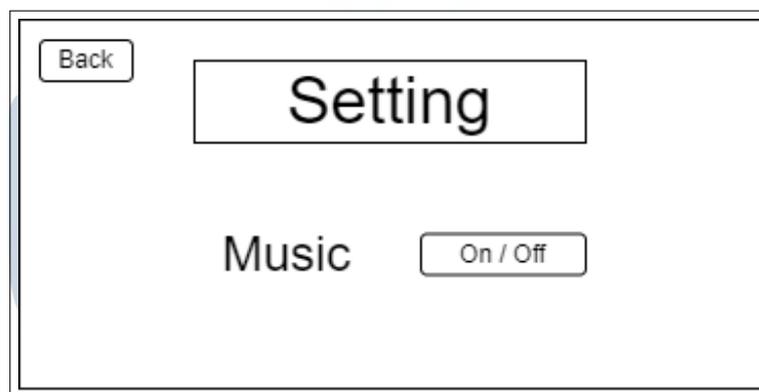
Tampilan / UI *home* merupakan tampilan yang pertama kali dilihat oleh pemain saat *game* dijalankan. Terdapat 3 tombol utama yang terdapat pada *Home*, yaitu *Start*, *Setting*, dan *Exit Game*. Tombol *Start* berfungsi untuk membawa pemain menuju *Mode Selection*. Kemudian terdapat tombol *Setting* yang berfungsi untuk menghidupkan atau mematikan musik yang terdapat pada *game*. Lalu yang terakhir adalah tombol *Exit Game* yang berfungsi untuk menutup *game* yang dimainkan. Rancangan tampilan / UI *home* terlihat seperti pada Gambar 3.8.



Gambar 3.8. Rancangan Tampilan / UI *Home*

3.3.2 Rancangan Tampilan / UI *Setting*

Tampilan / UI *setting* merupakan tampilan yang dilihat oleh pemain saat memilih tombol *Setting* pada Home. Pada tampilan ini, terdapat 2 tombol yang dapat diakses oleh pemain, yaitu tombol *On / Off*, dan *Back*. Tombol *On / Off* berfungsi untuk menghidupkan atau mematikan musik yang terdapat pada *game*, kemudian tombol *Back* berfungsi untuk kembali ke tampilan Home. Rancangan tampilan / UI *setting* terlihat seperti pada Gambar 3.9.

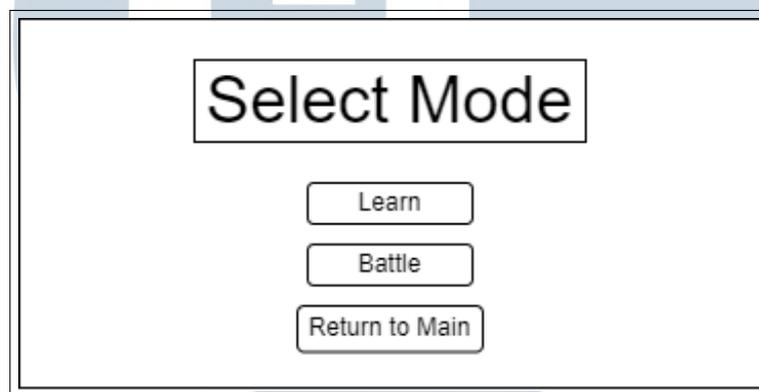


Gambar 3.9. Rancangan Tampilan / UI *Setting*

UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.3.3 Rancangan Tampilan / UI Mode Selection

Tampilan / UI *mode selection* merupakan tampilan yang dapat dilihat saat pemain memilih tombol *Start* pada *Home*. Tampilan ini berisikan pilihan mode yang dapat dimainkan oleh pemain, yang terdiri dari mode *Learn* dan mode *Battle*. Terdapat 3 tombol pada tampilan ini, yaitu *Learn*, *Battle*, dan *Return to Main*. Tombol *Learn* untuk menjalankan *Learn Mmode*, *Battle* untuk menjalankan *Battle Mode*, dan *Return to Main* untuk kembali ke *Home*. Rancangan tampilan / UI *mode selection* terlihat seperti pada Gambar 3.10.

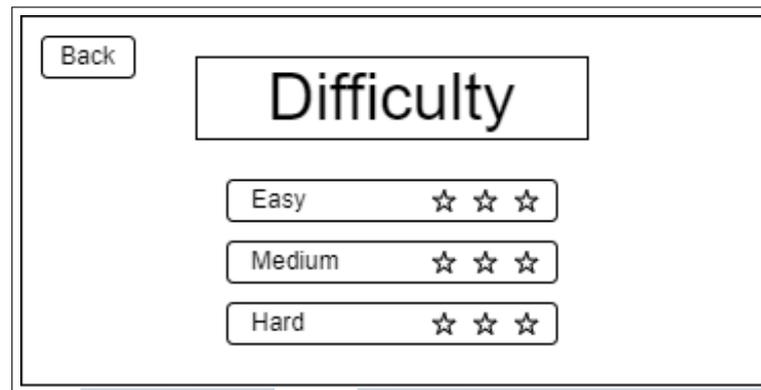


Gambar 3.10. Rancangan Tampilan / UI *Mode Selection*

3.3.4 Rancangan Tampilan / UI Difficulty Selection

Tampilan / UI *difficulty selection* merupakan tampilan yang berfungsi sebagai tempat memilih tingkat kesulitan dari *Battle Mode* yang akan dimainkan oleh pemain. Terdapat 3 tingkatan kesulitan yang terdapat pada *Battle Mode*, yaitu *Easy*, *Medium*, dan *Hard*. Selain itu terdapat juga tombol *Back* untuk kembali ke *Mode Selection*. Rancangan tampilan / UI *difficulty selection* terlihat seperti pada Gambar 3.11.

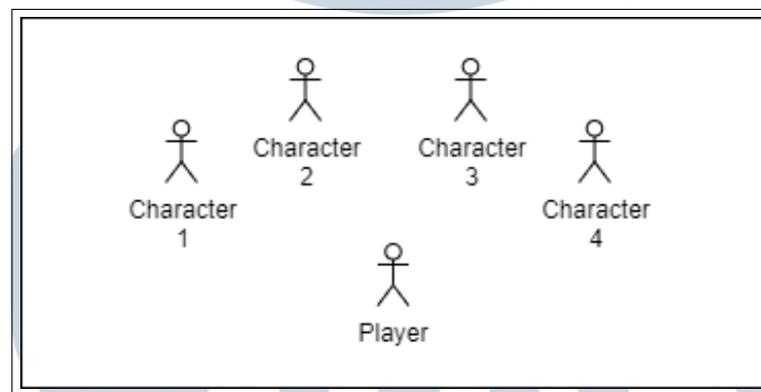
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.11. Rancangan Tampilan / UI *Difficulty Selection*

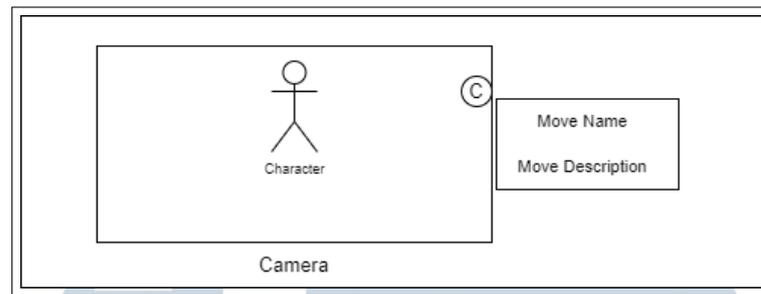
3.3.5 Rancangan Tampilan / UI *Learn Mode*

Tampilan / UI *learn mode* merupakan tampilan yang dilihat oleh pemain saat pemain menjalankan *Learn Mode*. Pemain akan melihat 4 karakter yang dapat dipilih untuk melakukan gerakan. Penempatan pemain serta 4 karakter tersebut terlihat pada Gambar 3.12.



Gambar 3.12. Rancangan Penempatan Pemain dan Karakter *Learn Mode*

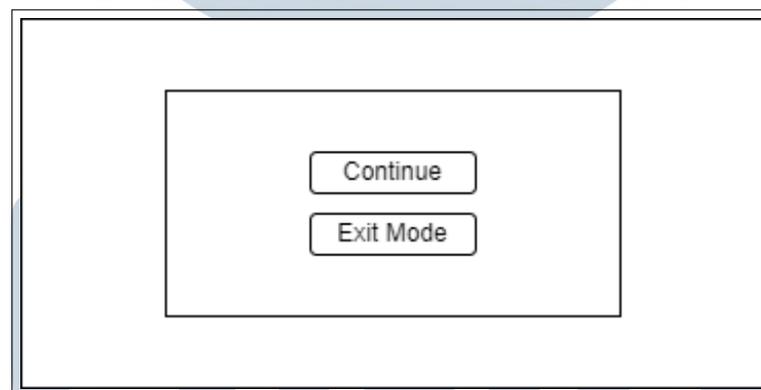
Setelah pemain memilih salah satu karakter dari 4 karakter yang disediakan, maka pemain dapat menampilkan deskripsi dari gerakan yang dilakukan oleh karakter tersebut. Rancangan tampilan / UI *learn mode* terlihat seperti pada Gambar 3.13.



Gambar 3.13. Rancangan Tampilan / UI *Learn Mode*

3.3.6 Rancangan Tampilan / UI *Learn Mode Setting*

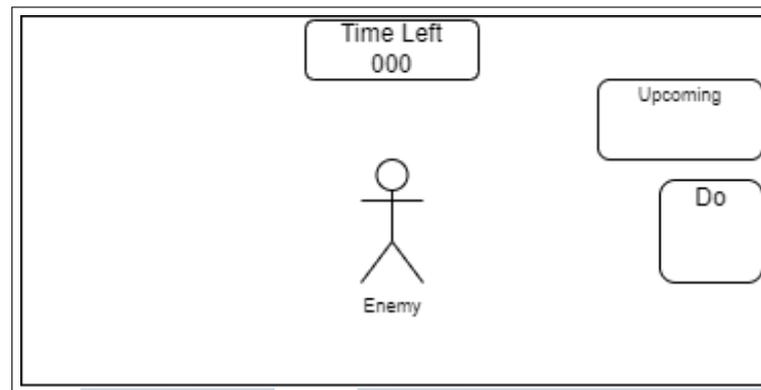
Tampilan / UI *learn mode setting* merupakan tampilan yang dilihat oleh pemain saat pemain memanggil tampilan *setting* saat berada dalam *Learn Mode*. Pemain diberikan 2 pilihan, yaitu *Continue* untuk melanjutkan *Learn Mode*, dan *Exit Mode* untuk kembali ke *Mode Selection*. Rancangan tampilan / UI *learn mode setting* terlihat pada Gambar 3.14.



Gambar 3.14. Rancangan Tampilan / UI *Learn Mode Setting*

3.3.7 Rancangan Tampilan / UI *Battle Mode*

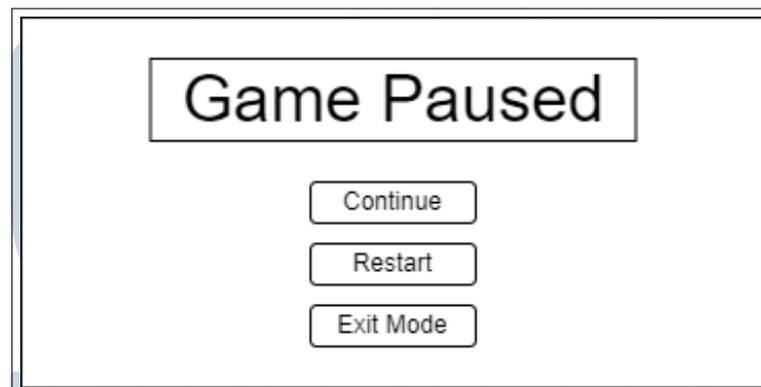
Tampilan / UI *battle mode* merupakan tampilan yang dapat dilihat oleh pemain saat menjalankan *Battle Mode*. Tampilan ini menunjukkan waktu pertarungan yang tersisa, gerakan yang harus dilakukan serta gerakan yang akan muncul selanjutnya, serta musuh yang akan dilawan oleh pemain. Rancangan tampilan / UI *battle mode* terlihat seperti pada Gambar 3.15.



Gambar 3.15. Rancangan Tampilan / UI *Battle Mode*

3.3.8 Rancangan Tampilan / UI Pause

Tampilan / UI *pause* merupakan tampilan yang muncul saat pemain melakukan *pause* saat memainkan *Battle Mode*. Pada tampilan ini, pemain akan diberikan 3 pilihan, yaitu *Continue*, *Restart*, dan *Exit Mode*. *Continue* berfungsi untuk melanjutkan pertarungan yang berlangsung, *Restart* berfungsi untuk mengulang kembali pertarungan dengan tingkat kesulitan yang sama, serta *Exit Mode* untuk keluar dari pertarungan dan kembali ke *Difficulty Selection*. Rancangan tampilan / UI *pause* terlihat seperti pada Gambar 3.16.



Gambar 3.16. Rancangan Tampilan / UI *Pause*

3.3.9 Rancangan Tampilan / UI Input Description

Tampilan / UI *input description* merupakan tampilan yang muncul saat pemain pertama kali menjalankan *Learn Mode* dan *Battle Mode*. Tampilan ini akan menampilkan daftar input yang dapat dilakukan oleh pemain serta fungsi dari input

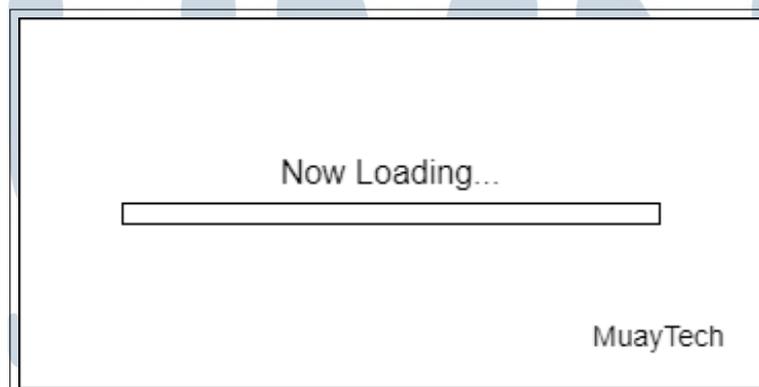
yang mereka lakukan. Rancangan tampilan / UI *input description* terlihat seperti pada Gambar 3.17.



Gambar 3.17. Rancangan Tampilan / UI *Input Description*

3.3.10 Rancangan Tampilan / UI Loading Screen

Tampilan / UI *loading screen* merupakan tampilan yang muncul saat dilakukan transisi antara perpindahan *scene* pada *Home* menuju *Mode Selection* maupun sebaliknya. Perpindahan *scene* ini dilakukan karena *scene* yang digunakan merupakan 2 *scene* yang berbeda. Rancangan tampilan / UI loading screen terlihat seperti pada Gambar 3.18.



Gambar 3.18. Rancangan Tampilan / UI Loading Screen

3.3.11 Rancangan Tampilan / UI Result

Tampilan / UI *result* merupakan tampilan yang muncul saat pemain menyelesaikan permainan, baik menang ataupun kalah. Saat pemain memenangkan per-

mainan, pemain akan ditunjukkan hasil dari pertarungan yang dilakukan yang terlihat dari total bintang yang didapatkan. Sedangkan saat pemain kalah dalam permainan, pemain tidak diperlihatkan bintang karena pemain gagal dalam memenangkan permainan. Selain itu pemain diberikan 2 pilihan pada tampilan ini, yaitu *Restart*, dan *Exit Mode*. Rancangan tampilan / UI *result* terlihat seperti pada Gambar 3.19.



Gambar 3.19. Rancangan Tampilan / UI *Result*

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

3.4 Perancangan Penggunaan Aset Game

Dalam perancangan penggunaan aset *game*, seluruh aset yang digunakan merupakan aset yang gratis untuk digunakan dan didapatkan melalui berbagai sumber. Berikut adalah penjelasan mengenai seluruh aset yang digunakan dalam perancangan *game* yang terlihat seperti pada Tabel 3.1.

Tabel 3.1. Daftar Aset Permainan

No	Nama	Fungsi	Sumber Aset
1	Sci-Fi Styled Modular Pack	<i>Terrain</i>	Unity Asset Store
2	Karakter pemain dan animasi	Karater pemain	Mixamo
3	Thai Boxing Animset	Karater pemain dan animasi	Unity Asset Store
4	Wall Texture	Tekstur tembok	Unity Asset Store
5	Skybox Series	<i>Skybox game</i>	Unity Asset Store
6	Meshtint Free Boxing Ring Mega Toon Series	<i>Ring boxing game</i>	Unity Asset Store
7	Fighting Game Level - Basic Training Stage	<i>Terrain</i>	Unity Asset Store
8	iTween	<i>Animation system</i>	Unity Asset Store
9	Progress Bar Pack	<i>Loading bar</i>	Unity Asset Store
10	Sci-Fi UI Pack	<i>Game UI</i>	Unity Asset Store
11	Google VR	<i>VR development envirointment</i>	Google VR
12	BGM & SFX	Musik <i>game</i> dan <i>sound effect</i>	Mixkit & Bendsound

UNIVERSITAS
MULTIMEDIA
NUSANTARA

3.5 Perancangan Kuesioner

Tahap perancangan kuesioner merupakan tahapan yang merancang sekumpulan pertanyaan menggunakan pemodelan GUESS. Kuesioner yang dirancang akan digunakan sebagai instrumen untuk mengukur tingkat kepuasan video game yang telah dibuat. Pertanyaan yang diberikan akan berisi mengenai 9 faktor, yaitu *Usability/Playability*, *Narratives*, *Play Engrossment*, *Enjoyment*, *Creative Freedom*, *Audio Aesthetics*, *Personal Gratification*, *Social Connectivity*, dan *Visual Aesthetics*. 9 faktor ini berisikan masing-masing dua pertanyaan sesuai dengan panduan yang terdapat pada pemodelan GUESS, yang terdiri seperti pada Tabel 3.2.

Tabel 3.2. Daftar Pertanyaan Pemodelan GUESS

Subscale Item	Statement
<i>Usability 1</i>	<i>I find the controls of the game to be straightforward</i>
<i>Usability 2</i>	<i>I find the game's interface to be easy to navigate</i>
<i>Narratives 1</i>	<i>I am captivated by the game's story from the beginning</i>
<i>Narratives 2</i>	<i>I enjoy the fantasy or story provided by the game</i>
<i>Play Engrossment 1</i>	<i>I feel detached from the outside world while playing the game</i>
<i>Play Engrossment 2</i>	<i>I do not care to check events that are happening in the real world during the game</i>
<i>Enjoyment 1</i>	<i>I think the game is fun</i>
<i>Enjoyment 2</i>	<i>I feel bored while playing the game</i>
<i>Creative Freedom 1</i>	<i>I feel the game allows me to be imaginative</i>
<i>Creative Freedom 2</i>	<i>I feel creative while playing the game</i>
<i>Audio Aesthetics 1</i>	<i>I enjoy the sound effects in the game</i>
<i>Audio Aesthetics 2</i>	<i>I feel the game's audio (e.g., sound effects, music) enhances my gaming experience</i>
<i>Personal Gratification 1</i>	<i>I am very focused on my own performance while playing the game</i>
<i>Personal Gratification 2</i>	<i>I want to do as well as possible during the game</i>
Lanjut pada halaman berikutnya	

Tabel 3.2 Daftar Pertanyaan Pemodelan GUESS (lanjutan)

Subscale Item	Statement
<i>Social Connectivity 1</i>	<i>I find the game supports social interaction (e.g., chat) between players</i>
<i>Social Connectivity 2</i>	<i>I like to play this game with other players</i>
<i>Visual Aesthetics 1</i>	<i>I enjoy the game's graphics</i>
<i>Visual Aesthetics 2</i>	<i>I think the game is visually appealing</i>

Berdasarkan tabel diatas, 3 diantaranya tidak digunakan sehingga hanya diberikan 15 pertanyaan. Pertanyaan yang tidak termasuk merupakan pertanyaan pada *subscale Narratives 2*, *Social Connectivity 1* dan *2*, hal ini dikarenakan 3 poin tersebut tidak termasuk dalam *game* yang dibuat. Selain itu, setiap pertanyaan memiliki *7-point rating scale* untuk mengukur tingkat kepuasan pemain yang diantaranya adalah:

- (a) Sangat tidak setuju dengan nilai 1.
- (b) Tidak setuju dengan nilai 2.
- (c) Agak tidak setuju dengan nilai 3.
- (d) Biasa saja dengan nilai 4.
- (e) Agak setuju dengan nilai 5.
- (f) Setuju dengan nilai 6.
- (g) Sangat setuju dengan nilai 7.

Namun, terdapat pertanyaan yang memiliki *7-point rating scale* yang terbalik, sehingga penilaian yang diberikan bertolak belakang dengan penilaian sebelumnya. Pengukuran tingkat kepuasan pemain tersebut terdiri atas:

- (a) Sangat tidak setuju dengan nilai 7.
- (b) Tidak setuju dengan nilai 6.
- (c) Agak tidak setuju dengan nilai 5.

- (d) Biasa saja dengan nilai 4.
- (e) Agak setuju dengan nilai 3.
- (f) Setuju dengan nilai 2.
- (g) Sangat setuju dengan nilai 1.

Setelah pembuatan soal dan penentuan *7-point rating scale*, responden diharapkan untuk mengisi pertanyaan-pertanyaan tersebut berdasarkan *rating scale* yang telah disediakan. Selanjutnya, hasil dari responden tersebut akan diakumulasikan sehingga mendapatkan nilai tingkat kepuasan yang dibagi menjadi beberapa tingkatan, yaitu:

- (a) Sangat tidak puas dengan nilai 0% sampai 14%.
- (b) Tidak puas dengan nilai 15% sampai 28%.
- (c) Agak tidak puas dengan nilai 29% sampai 42%.
- (d) Biasa saja dengan nilai 43% sampai 56%.
- (e) Agak puas dengan nilai 57% sampai 70%.
- (f) Puas dengan nilai 71% sampai 84%.
- (g) Sangat puas dengan nilai diatas 84%.

3.6 Pembuatan Aplikasi

Dalam proses pembuatan aplikasi, tahapan-tahapan yang dilakukan dalam pengembangan aplikasi adalah dengan menggunakan metode *waterfall*. Tahapan-tahapan dari metode *waterfall* tersebut terdiri atas:

1. *Requirements Analysis*

Pada tahap ini, dilakukan pengembangan sistem yang diperlukan pada *game* yang akan dibangun. Pada *game* yang dibangun merupakan *game* pembelajaran bela diri Muay Thai, dengan dua mode utama yaitu metode pembelajaran langsung melalui teori dan metode pertarungan dengan melawan musuh yang telah disediakan.

2. *System Design*

Tahap ini merupakan tahap perancangan sistem yang akan dibangun berdasarkan *requirement analysis* yang telah dilakukan. Perancangan sistem dilakukan dengan membuat perancangan *flowchart*, tampilan / UI, serta perancangan aset yang akan digunakan dalam membangun *game*.

3. *Implementation*

Pada tahap ini dilakukan proses pembangunan *game* berdasarkan tahap *system design* sebelumnya. Pembangunan *game* dilakukan menggunakan bahasa pemrograman *C#* dengan aplikasi Unity berdasarkan perancangan *flowchart*, tampilan / UI, dan perancangan aset yang dilakukan pada tahap sebelumnya.

4. *Integration & Testing*

Tahap ini merupakan tahap penggabungan hasil implementasi yang dilakukan pada tahap sebelumnya, kemudian dilakukan pengujian untuk mengetahui *game* yang dibangun telah sesuai dengan rancangan serta fungsi yang terdapat didalamnya berjalan dengan baik atau tidak.

5. *Operation & Maintenance*

Pada tahap ini, dilakukan pemeliharaan *game* yang telah berhasil dibangun, termasuk memperbaiki kesalahan yang tidak ditemukan pada tahap sebelumnya.

3.7 Pengujian Aplikasi

Tahap pengujian aplikasi merupakan tahap pengujian fungsionalitas yang terdapat pada *game*. Pengujian tersebut menggunakan pengujian *black box testing*, yaitu pengujian yang berfokus pada fungsi-fungsi yang terdapat pada *game*. Pengujian ini akan dilakukan pengujian secara detail dari main menu, mode selection, learn mode, dan battle mode yang terdapat dalam *game*. Pengujian yang dilakukan terlihat seperti pada Tabel 3.3.

Tabel 3.3. Pengujian Black Box

Fungsi Utama	Detail Fungsi	Hasil yang Diharapkan	Hasil Pengujian
<i>Main Menu</i>	<i>Start</i>	Saat dipilih maka <i>game</i> akan membawa pemain menuju <i>mode selection</i> .	Berhasil
<i>Main Menu</i>	<i>Setting</i>	Saat dipilih maka <i>game</i> menampilkan <i>menu setting</i> dan pemain dapat mengatur pengaturan musik pada <i>game</i> .	Berhasil
<i>Main Menu</i>	<i>Exit Game</i>	Saat dipilih maka pemain akan keluar dari <i>game</i> yang dimainkan.	Berhasil
<i>Mode Selection</i>	<i>Learn Mode</i>	Saat dipilih maka <i>game</i> akan menjalankan <i>learn mode</i> .	Berhasil
<i>Mode Selection</i>	<i>Battle Mode</i>	Saat dipilih maka <i>game</i> akan menjalankan <i>battle mode</i> .	Berhasil
<i>Mode Selection</i>	<i>Return to Main Menu</i>	Saat dipilih maka <i>game</i> akan membawa pemain menuju <i>main menu</i> .	Berhasil
<i>Learn Mode</i>	<i>Close Description</i>	Pemain dapat menutup deskripsi input yang ditampilkan pada <i>learn mode</i> .	Berhasil
<i>Learn Mode</i>	<i>Idle</i>	Pemain dapat memilih karakter yang terdapat pada <i>learn mode</i> , serta pemain dapat menampilkan <i>menu setting</i> .	Berhasil
Lanjut pada halaman berikutnya			

Tabel 3.3 Pengujian Black Box (lanjutan)

Fungsi Utama	Detail Fungsi	Hasil yang Diharapkan	Hasil Pengujian
<i>Learn Mode</i>	<i>Character Selected</i>	Pemain dapat menggerakkan atau menghentikan karakter yang dipilih untuk melakukan gerakan dasar Muay Thai, memutar karakter, menampilkan deskripsi dari gerakan Muay Thai yang dilakukan karakter serta unselect karakter yang dipilih.	Berhasil
<i>Learn Mode</i>	<i>Setting</i>	Pemain dapat melanjutkan <i>learn mode</i> atau keluar dari <i>learn mode</i> dan akan kembali menuju <i>mode selection</i> .	Berhasil
<i>Battle Mode</i>	<i>Difficulty Selection</i>	Pemain akan diminta untuk memilih tingkat kesulitan pada <i>battle mode</i> yang akan dimainkan, maupun kembali ke <i>mode selection</i> .	Berhasil
<i>Battle Mode</i>	<i>Close Description</i>	Pemain dapat menutup deskripsi input yang ditampilkan pada <i>battle mode</i> .	Berhasil
<i>Battle Mode</i>	<i>Attack Shuffle</i>	Game melakukan pengacakan gerakan yang akan dilakukan oleh pemain.	Berhasil
Lanjut pada halaman berikutnya			

Tabel 3.3 Pengujian Black Box (lanjutan)

Fungsi Utama	Detail Fungsi	Hasil yang Diharapkan	Hasil Pengujian
<i>Battle Mode</i>	<i>Battle</i>	Pemain dapat menggerakan karakter untuk melakukan gerakan <i>jab</i> , <i>hook</i> , <i>cross / straight</i> , dan <i>uppercut</i> yang diminta oleh sistem, serta dapat melakukan <i>pause</i> saat <i>battle</i> berlangsung.	Berhasil
<i>Battle Mode</i>	<i>Move Attack</i> <i>Button</i>	Sistem dapat menggerakan <i>button</i> yang digunakan untuk menunjukkan gerakan yang harus dilakukan selanjutnya, menghapus <i>button</i> yang sudah digunakan serta melakukan <i>instantiate button</i> gerakan selanjutnya.	Berhasil
<i>Battle Mode</i>	<i>Pause</i>	<i>Game</i> akan menampilkan <i>menu pause</i> yang digunakan pemain untuk menghentikan <i>battle</i> , pemain dapat memilih untuk melanjutkan <i>battle</i> , mengulang kembali <i>battle</i> , serta keluar dari <i>battle</i> dan kembali menuju <i>difficulty selection</i> .	Berhasil