

BAB 3

METODOLOGI PENELITIAN

3.1 Metodologi Penelitian

Dalam proses perancangan dan pembangunan gim *turn-based* dengan *procedural content generation*, metodologi penelitian yang adalah:

1. Identifikasi Masalah

Tahap identifikasi masalah dilakukan untuk menemukan permasalahan yang ada dan akan diselesaikan saat merancang dan membangun gim *turn-based* dengan *procedural content generation*

2. Studi Literatur

Tahap studi literatur meliputi pengumpulan informasi terhadap *procedural content generation* beserta dengan metode-metode yang dapat digunakan dan pembelajaran metode perancangan dan pengembangan gim *turn-based*

3. Perancangan gim

Tahap perancangan gim meliputi proses perancangan konsep awal gim, rancangan *flow* dari *gameplay* menggunakan *flowchart*, dan pembuatan/pencarian aset-aset yang perlu digunakan dalam gim.

4. Pembuatan gim

Tahap pembuatan gim meliputi proses pembuatan sistem-sistem/komponen-komponen dalam gim serta pembahasan potongan kode/*script* yang dibuat menggunakan bahasa pemrograman *GDS*cript milik game engine godot.

5. Uji coba gim

Tahap uji coba meliputi proses penyebaran gim ke para pemain melalui sosial media untuk diuji dan dinilai oleh pemain. Penilaian oleh pemain menggunakan *google forms* dengan pertanyaan-pertanyaan dari *GUESS* yang disesuaikan dengan fitur-fitur dari gim

6. Evaluasi

Tahap evaluasi meliputi pemrosesan data yang didapat dari para pemain serta menarik kesimpulan terhadap gim menggunakan data yang didapatkan.

3.2 Perancangan gim

Rancang bangun gim *turn-based* dimulai dengan penyusunan struktur gim yang berisi *formal elements* dan *dramatic elements* dalam gim. Setelah gim memiliki struktur, pencarian aset-aset yang ingin digunakan dalam gim dilakukan. Aset-aset tersebut dimasukkan ke dalam tabel 3.1 yang berisi cuplikan gambar dari aset, tipe aset, dan pembuat aset. *Flowchart* yang ditampilkan menjelaskan langkah-langkah yang pemain lakukan saat mereka berinteraksi dengan gim dan juga menjelaskan langkah-langkah gim memproses interaksi dari pemain.

3.2.1 Struktur gim

Judul Game: TTB Project

TTB Project adalah sebuah gim ber-genre *turn-based* yang memiliki tujuan untuk mengalahkan musuh kuat/*boss* di ruangan terakhir. Sebelum mencapai ruangan terakhir, pemain harus melewati ruangan-ruangan sebelumnya yang dapat berisi musuh, toko atau ruang kemah.

Formal elements yang terdapat dalam gim TTB Project adalah sebagai berikut:

- Player

gim TTB Project merupakan sebuah gim *singleplayer* yang memungkinkan gim untuk memuat satu pemain.

- Objectives

Pemain bertujuan untuk mengalahkan *boss* akhir setelah pemain sudah bertambah kuat melalui ruangan-ruangan sebelumnya.

- Procedures

1. Pemain membuka aplikasi gim dalam bentuk .exe untuk mencapai layar *main menu* gim.

2. Di dalam *main menu* terdapat empat tombol yang berfungsi sebagai tombol mulai, tombol daftar nama aset, tombol penghapusan data tersimpan, dan tombol keluar.

3. Saat pemain menekan tombol mulai, pemain akan masuk ke dalam *scene tutorial* yang memiliki dua halaman. Halaman pertama berisi cuplikan

layar pertarungan dengan beberapa petunjuk bermain dan tombol untuk melanjutkan ke halaman selanjutnya. Halaman kedua berisi susunan gambar yang menunjukkan tujuan akhir dan arti dari gambar-gambar ruangan yang perlu dilewati serta tombol untuk masuk ke dalam *scene* peta.

4. Saat *scene* peta pertama ditampilkan dan pemain tidak memiliki data yang tersimpan sebelumnya gim menghasilkan satu ruangan pertarungan. saat pemain menekan tombol ruangan tersebut pemain akan ditampilkan *scene* pertarungan.
5. Saat layar pertarungan ditampilkan, gim akan memilih antara dua musuh yang ada dalam data musuh, lalu karakter musuh akan dibuat dan ditampilkan pada bagian kanan layar sementara karakter pemain ditampilkan pada bagian kiri layar. Bagian bawah layar memiliki tiga tombol yang berfungsi sebagai pengendali aksi yang akan dilakukan karakter pemain. Saat karakter musuh atau karakter pemain memiliki *hitpoint* kurang dari nol, sebuah *dialog box* akan muncul dengan tombol yang berfungsi untuk melanjutkan permainan.
 - Jika karakter pemain yang memiliki *hitpoint* kurang dari nol, tombol akan berfungsi untuk menampilkan *scene* akhir gim
 - Jika karakter musuh yang memiliki *hitpoint* kurang dari nol, tombol akan berfungsi untuk mengembalikan pemain ke *scene* peta dan memberikan pemain sejumlah uang
6. Saat pemain kembali ke *scene* peta setelah memasuki ruangan, gim akan menghasilkan dua sampai empat ruangan baru secara acak di atas ruangan yang sebelumnya dipilih dan meningkatkan hitungan ruangan. Pemain lalu dapat memilih ruangan yang telah dihasilkan.
7. Setiap kali peta menghasilkan sepuluh ruangan, generasi map selanjutnya akan menghasilkan satu ruangan ekstra dalam map yang merupakan ruangan *boss*. Pemain dapat memilih ruangan *boss* tersebut untuk melawan *boss* dan menyelesaikan sesi permainan. Tombol yang sebelumnya muncul untuk mengembalikan pemain ke *scene* peta sekarang berfungsi untuk menampilkan *scene* akhir gim.
8. Dalam *scene* akhir gim, terdapat hitungan ruangan tertinggi yang dicapai pemain, rasio jumlah input sempurna menyerang dibanding

serangan yang dilakukan oleh karakter, dan rasio jumlah input sempurna berlindung dibanding serangan yang didapatkan oleh karakter. Terdapat juga tombol yang digunakan untuk mengembalikan pemain ke *scene main menu*, menghapus data yang tersimpan, serta menyimpan tangkapan *scene* ke dalam folder aplikasi gim.

- Rules

Aturan-aturan yang terdapat dalam gim adalah

- Pemain mengendalikan pemain dengan menekan tombol UI, karakter musuh dan tombol spasi *keyboard*
- Pemain dan musuh hanya dapat menyerang satu sama lain saat giliran mereka
- Pemain dapat meningkatkan atribut-atribut karakter menggunakan mata uang yang didapatkan di akhir pertarungan.
- Pemain dapat menyembuhkan karakter melalui ruangan kemah atau menangkis serangan saat menggunakan aksi *parry stance*

- Resource

Resource yang digunakan oleh pemain dalam gim adalah

- Player hitpoints
hitpoint merupakan *resource* utama yang perlu dijaga pemain agar tidak menyentuh nol
- Attack stat
Atribut *attack* digunakan untuk menandakan tinggi nilai serangan
- Defense stat
Atribut *defense* digunakan dalam kalkulasi pengurangan nilai serangan musuh saat pemain berlindung dan kalkulasi nilai penyembuhan saat pemain menangkis serangan
- Currency
Currency digunakan untuk meningkatkan atribut *attack*, *defense* atau *hitpoint* maksimum

- Conflict

Pemain harus mengalahkan musuh-musuh untuk dapat meningkatkan atribut karakter agar *boss* dapat lebih mudah dikalahkan.

- **Boundaries**

Pemain hanya dapat memilih ruangan-ruangan yang telah dibuat menggunakan *content-selection procedural content generation*.

- **Outcome**

- Gim berakhir saat *hitpoint* karakter pemain menyentuh nol atau pemain berhasil mengalahkan *boss*.

Dramatic elements dari gim TTB Project adalah:

- **Challenge**

Challenge yang dimiliki gim adalah untuk mengalahkan *boss* setelah melewati ruangan-ruangan yang dipilih secara acak, sehingga kesempatan mendapat nilai atribut yang optimal menjadi kecil.

- **Play**

Gameplay dari gim menggunakan sistem *turn-based* yang membatasi serangan pemain saat giliran mereka. Gim juga menggunakan sistem *timed input* untuk pilihan berlindung saat giliran musuh berlangsung, meningkatkan daya serangan saat karakter pemain menyerang, dan pemilihan peningkatan atribut saat menggunakan aksi *self buff*

- **Premise**

Premise dari gim mengambil tema *dark-fantasy* dengan suasana yang gelap dan misterius.

- **Characters**

Karakter-karakter yang terdapat dalam gim adalah:

- **Hooded Figure**

Hooded figure adalah seseorang bertudung yang akan menjadi karakter yang dapat pemain kendalikan.

- **Slime**

Slime adalah sebuah makhluk kecil yang menjadi musuh pertama saat pemain melakukan pertarungan

- Dark samurai

Dark samurai adalah karakter *humanoid* menggunakan topi dan pedang yang menjadi musuh alternatif selain *slime*

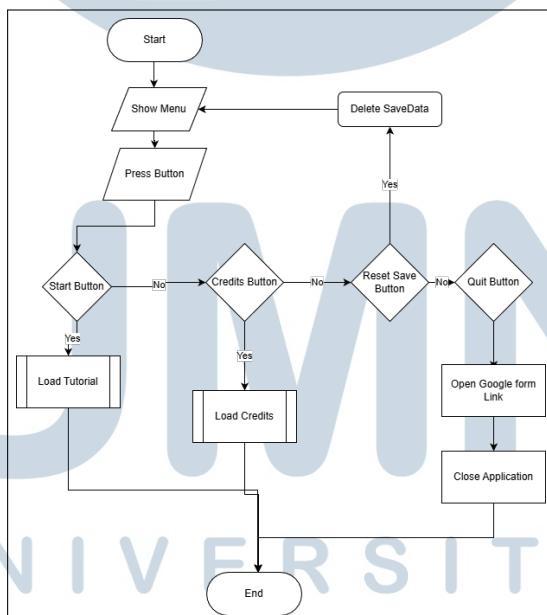
- Executioner

Executioner adalah makhluk bayangan yang menjadi *boss* yang perlu dikalahkan untuk mengakhiri sesi permainan.

Karena gim berfokus kepada sistem *procedural content generation* dan *turn-based combat*, gim tidak memiliki komponen naratif yang mengakibatkan tidak adanya komponen *story* pada *dramatic elements*

3.3 Alur Gim

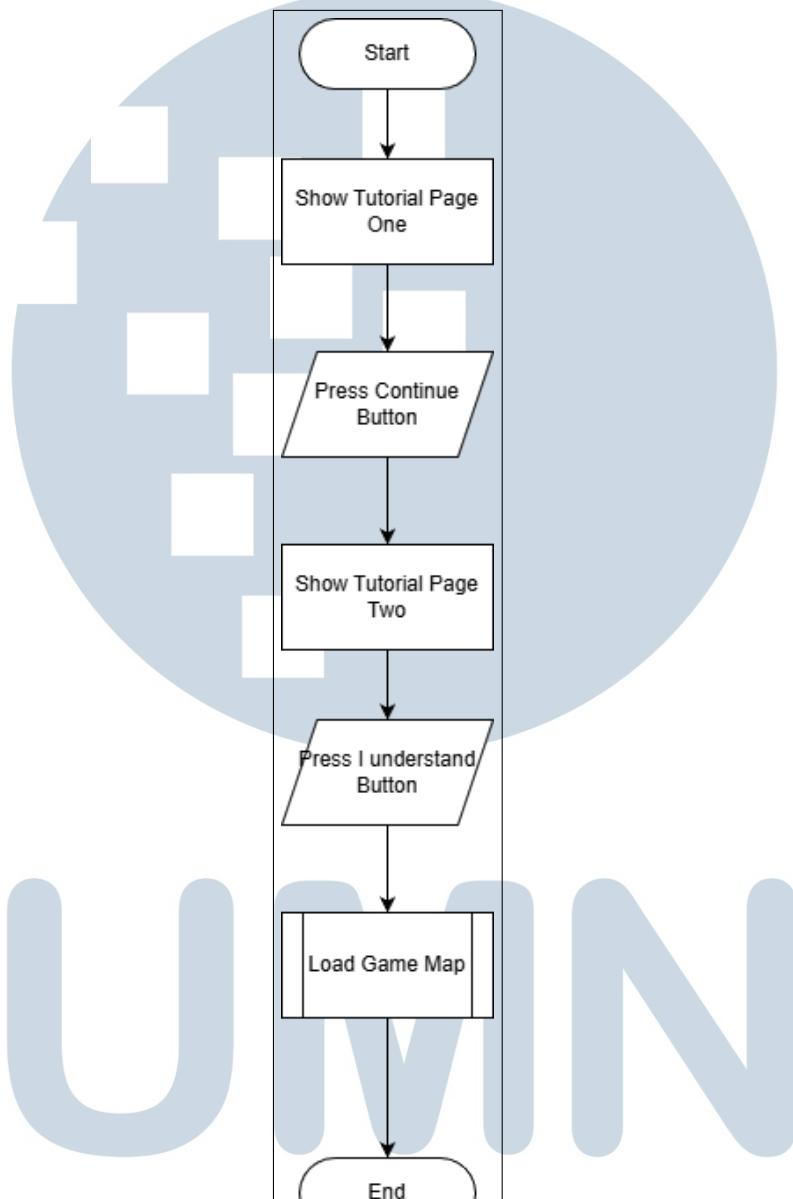
Alur dari gim digambarkan menggunakan beberapa *flowchart*. *Flowchart* - *flowchart* tersebut akan berisi bagaimana sistem dari gim bekerja dan menjelaskan perjalanan pemain saat pemain memulai atau membuka gim.



Gambar 3.1. *Flowchart Main Menu*

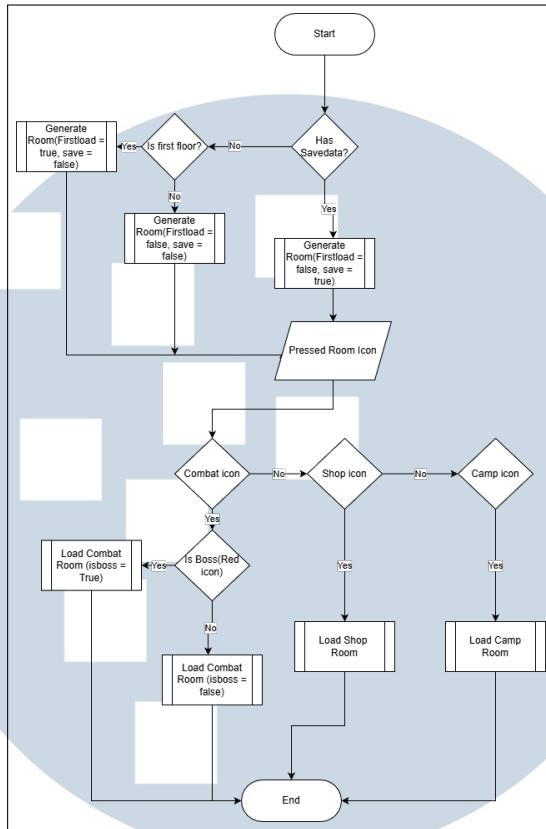
Flowchart pada Gambar 3.1 menampilkan alur yang dimulai saat pemain membuka aplikasi gim. Dalam *main menu* terdapat empat tombol yaitu sebagai tombol *play* yang digunakan untuk memulai gim, tombol *credits* untuk menampilkan sumber-sumber dari asset yang digunakan, tombol *reset progress*

untuk menghapus data yang tersimpan dan tombol *quit* untuk menutup aplikasi gim serta membuka halaman kuesioner di *browser*.



Gambar 3.2. Flowchart Tutorial

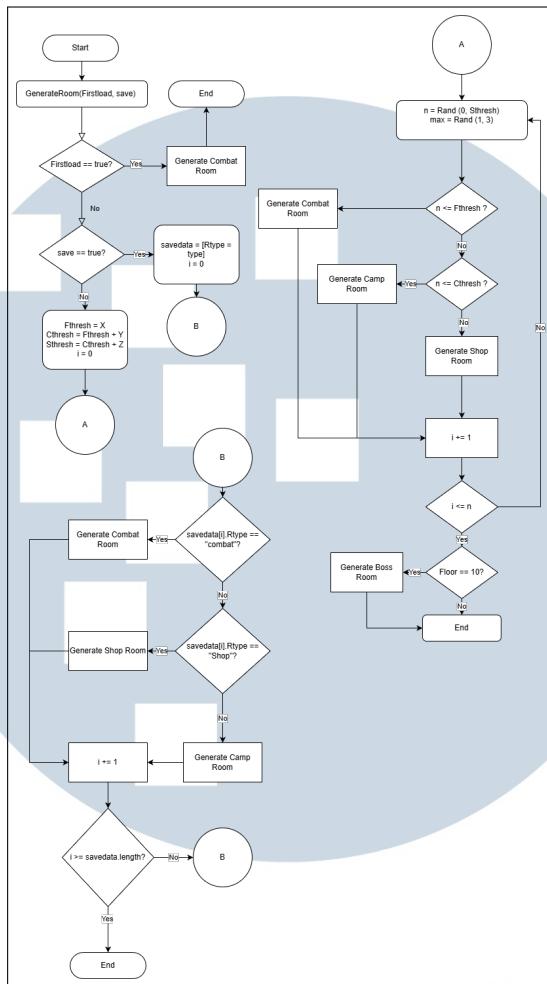
Flowchart pada Gambar 3.2 menampilkan alur sebelum pemain memasuki peta gim. *Tutorial* dari gim terdiri dari dua halaman, halaman pertama menampilkan potongan gambar yang menggambarkan tampilan pertarungan serta penjelasan singkat terhadap mekanisme gim, halaman kedua menampilkan kumpulan gambar yang menjelaskan tujuan akhir gim, tantangan yang akan dilewati, dan bantuan yang dapat digunakan untuk mempermudah pemain dalam menyelesaikan gim.



Gambar 3.3. Flowchart loadmap

Flowchart pada Gambar 3.3 menampilkan alur pembuatan peta dalam gim. Saat peta pertama ditampilkan jika pemain belum memainkan gim sebelumnya dan tidak memiliki data yang tersimpan, peta akan menghasilkan satu ruangan pertarungan. Setiap kali pemain menyelesaikan suatu tujuan dari ruangan, peta akan menghasilkan satu sampai empat ruangan lain yang dapat pemain pilih.

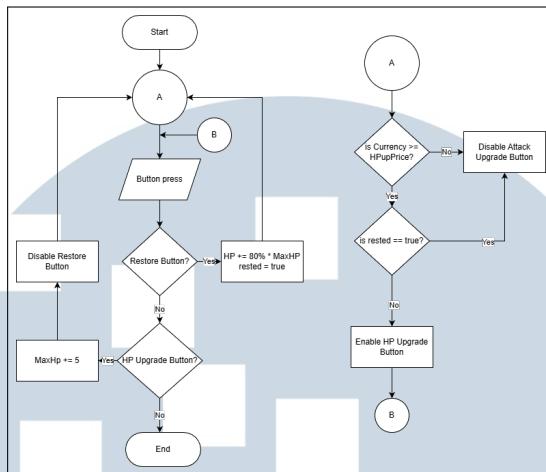
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.4. Flowchart Generate Room

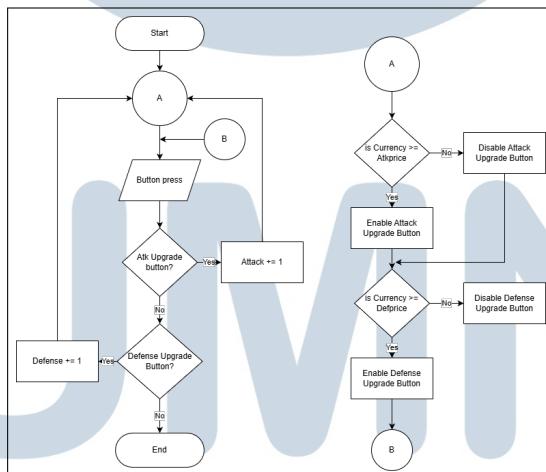
Flowchart pada Gambar 3.4 menampilkan cara kerja *procedural content generation* pada peta gim. Cara kerja *room generation* pada peta adalah dengan menggunakan *random number generation* bersamaan dengan *weighted probability* untuk menentukan tipe ruangan yang akan dihasilkan. Setiap kali peta melakukan generasi ruangan, sebuah variabel akan ditingkatkan lalu disimpan yang akan menjadi tanda dari jumlah ruangan yang pemain sudah selesaikan. Setiap sepuluh ruangan yang dibuat, generasi ruangan selanjutnya akan menghasilkan satu ruangan *boss* tambahan.

**UNIVERSITAS
MULTIMEDIA
NUSANTARA**



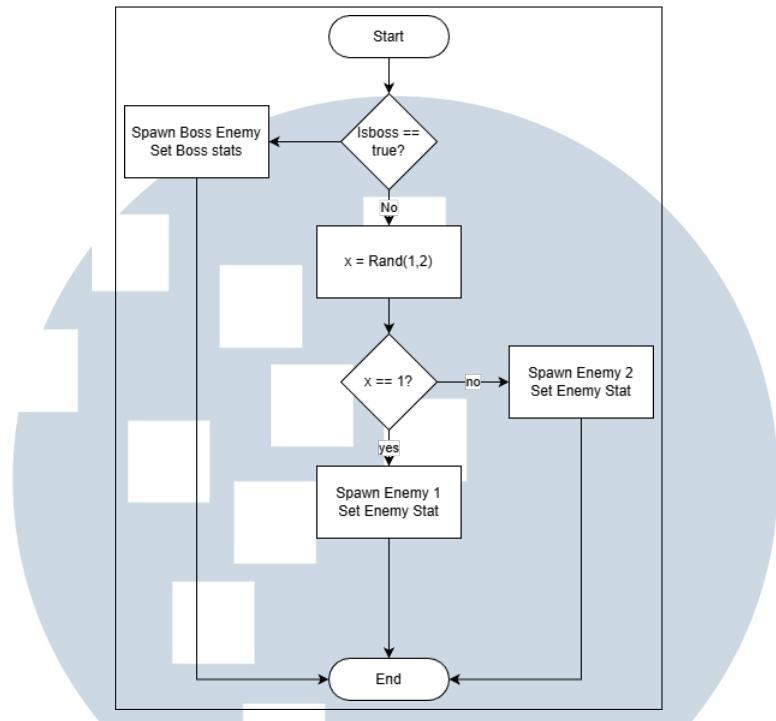
Gambar 3.5. Flowchart Camp Room

Gambar 3.5 menampilkan alur peningkatan *hitpoint* maksimum atau mengembalikan *hitpoint* yang hilang saat pemain memilih ruangan kemah. Tombol peningkatan *hitpoint* maksimum akan diaktifkan jika pemain memiliki mata uang yang cukup



Gambar 3.6. Flowchart Shop Room

Gambar 3.6 menampilkan alur peningkatan atribut karakter saat pemain memilih ruangan toko. Tombol untuk melakukan peningkatan atribut akan diaktifkan jika pemain memiliki mata uang yang cukup.



Gambar 3.7. *Flowchart Combat Room*

Gambar 3.7 menampilkan *flowchart* yang menggambarkan alur persiapan ruangan pertarungan. Ruangan akan memilih secara acak antara dua objek musuh saat pemain memilih ruangan yang bukan ruangan *boss*, lalu menetapkan atribut musuh. Sementara objek musuh *boss* akan selalu terpilih saat tipe ruangan yang dipilih adalah ruang pertarungan *boss*.

3.4 Game Assets

Aset-aset yang digunakan dalam pembangunan game adalah sebagai berikut:

Tabel 3.1. Tabel Aset-Aset Grafis Gim

Nama	Gambar	Sumber
Panah		Aset Pribadi
Lanjut pada halaman berikutnya		

Tabel 3.1 Tabel Aset-Aset Gim (lanjutan)

Nama	Gambar	Sumber
Ikon plus		Aset Pribadi
Karakter player		Aset Pribadi
Musuh 1		Aset Pribadi
Ikon Tangga		Aset Pribadi
Pedang		Aset Pribadi
Tangan		Aset Pribadi
Ikon Kantong		OpenGameArt
Gelembung		OpenGameArt
Ikon Kemah		OpenGameArt
Ikon Ledakan		OpenGameArt
Ikon Perisai		OpenGameArt
Ikon Penjual		OpenGameArt
Lanjut pada halaman berikutnya		

Tabel 3.1 Tabel Aset-Aset Gim (lanjutan)

Nama	Gambar	Sumber
Ikon Pedang		OpenGameArt
<i>Demon Woods</i>		https://aethrall.itch.io/demon-woods-parallax-background
Musuh 2		https://aimmaga.itch.io/darksamurai
Musuh Boss		https://darkpixel-kronovi.itch.io/undead-executioner

Tabel 3.1 merupakan tabel yang berisi aset-aset yang digunakan untuk membangun gim. Aset pribadi dibuat menggunakan aplikasi *Aseprite*.

Tabel 3.2. Tabel Aset-Aset Audio Gim

Judul	Keterangan	Sumber
<i>The Waiting Room</i>	Musik di <i>main menu</i>	Waterflame
<i>Dark Seaside</i>	Musik di <i>scene</i> utama(peta, pertarungan, kemah dan toko)	Waterflame
<i>Hit Hurt</i>	Efek suara saat karakter terkena serangan	Profmudkip/Freesound.org
Lanjut pada halaman berikutnya		

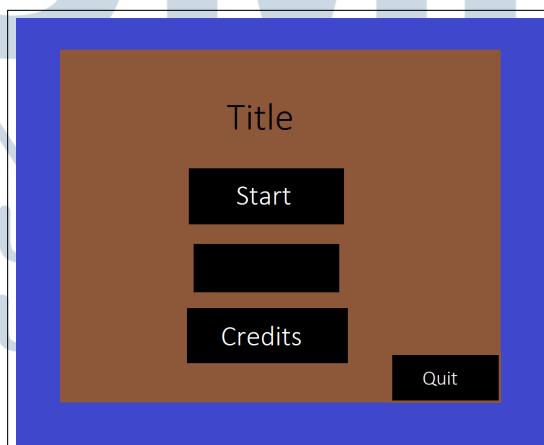
Tabel 3.2 Tabel Aset-Aset Gim (lanjutan)

Nama	Gambar	Sumber
<i>Sword Pull</i>	Efek suara saat karakter pemain berlindung dari atau menangkis serangan	JoBro/Freesound.org
<i>Metallic Ting</i>	Efek suara untuk menandai input sempurna saat karakter pemain menyerang	Gamagool/Freesound.org
<i>Spellcast</i>	Efek suara saat karakter pemain memulai aksi <i>self buff</i>	Toby Fox
<i>Power</i>	Efek suara saat karakter pemain menyelesaikan aksi <i>self buff</i>	Toby Fox

Tabel 3.2 berisi aset-aset suara yang digunakan dalam pembangunan gim.

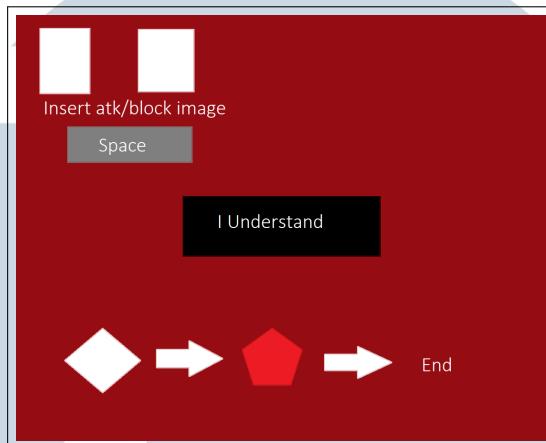
3.5 Perancangan UI dan Tampilan Gim

Rancangan awal untuk tampilan serta UI dari gim akan ditampilkan menggunakan *mock up* atau draf.



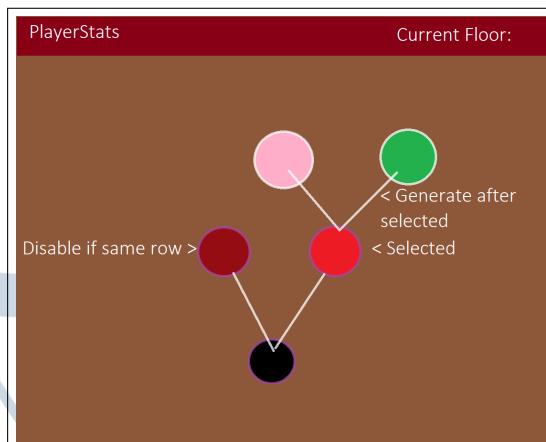
Gambar 3.8. Mockup Main Menu

Gambar 3.8 menampilkan draf awal dari menu awal gim. Menu memiliki empat tombol dengan fungsinya tertera untuk membuka/melanjutkan ke *scene* tertentu.



Gambar 3.9. *Mockup Tutorial*

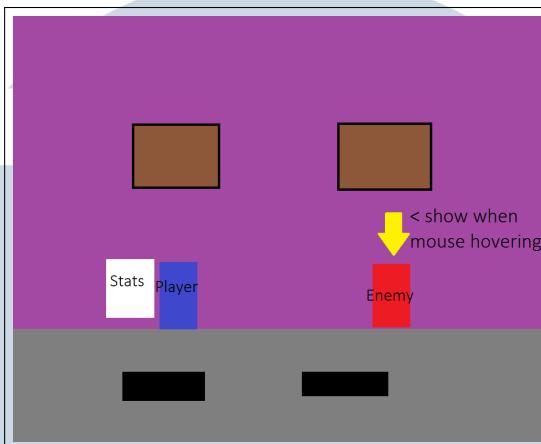
Gambar 3.9 menampilkan *mockup* dari *scene tutorial* dalam gim. Gim menggunakan teks yang minimal untuk *tutorial* sehingga *scene*/halaman *tutorial* mengandalkan *sprite-sprite* atau aset gambar untuk memberikan informasi kepada pemain.



Gambar 3.10. *Mockup Mapscreen*

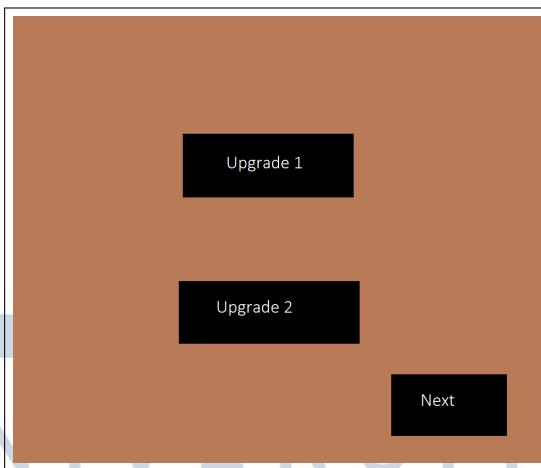
Gambar 3.10 menampilkan *mockup* dari peta yang akan dilalui oleh pemain. Kondisi gim dalam gambar adalah setelah pemain menyelesaikan ruangan kedua dan memilih ruangan kanan sebagai ruangan tujuan. Terlihat bahwa ruangan kedua kiri lebih gelap dibanding ruangan kanan untuk menandakan ruangan yang tidak

dipilih oleh pemain, dan ruangan yang di generasi selanjutnya berada di atas posisi ruangan kedua kiri serta hanya tersambung pada ruang tersebut.



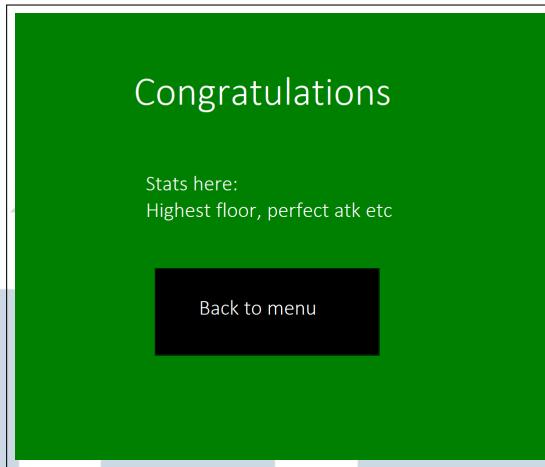
Gambar 3.11. *Mockup Combat*

Gambar 3.11 menampilkan *mockup scene* pertarungan saat pemain memilih ruangan pertarungan. Karakter pemain memiliki *stats* yang akan ditampilkan saat pemain meng-*hover* mouse ke karakter pemain. Pemain dapat mengendalikan karakter menggunakan dua tombol UI yang berada di bagian bawah layar.



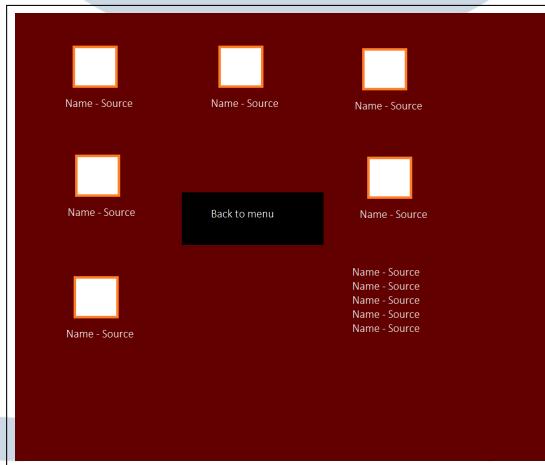
Gambar 3.12. *Mockup Camp* dan *Shop*

Gambar 3.12 menampilkan *mockup* untuk *scene* kemah dan *scene* toko. Kedua *scene* tersebut menggunakan layout yang serupa tetapi dengan fungsionalitas yang berbeda, *scene* kemah digunakan untuk meningkatkan dan mengembalikan *hitpoint* sementara *scene* toko digunakan untuk meningkatkan atribut dari karakter pemain.



Gambar 3.13. *Mockup Endscreen*

Gambar 3.13 menggambarkan layar terakhir saat pemain berhasil mengalahkan *boss* atau gagal dengan menghabiskan *hitpoint* mereka. *End screen* berisi pesan akhir, menunjukkan jumlah ruangan terbanyak yang pemain selesaikan serta *stat* pemain terakhir setelah menyelesaikan sesi permainan.



Gambar 3.14. *Mockup Credits*

Gambar 3.14 menampilkan *mockup* untuk *scene credits*. *Scene credits* menampilkan sumber dari aset-aset yang digunakan dalam gim.

UNIVERSITAS
MULTIMEDIA
NUSANTARA