

## BAB 3 METODOLOGI PENELITIAN

Metodologi penelitian yang digunakan dalam melakukan perancangan dan pembangunan *game horror* berbasis *virtual reality* dengan mengimplementasikan *procedural content generation* menggunakan algoritma *cellular automata* adalah sebagai berikut.

### 3.1 Telaah Literatur

Pada tahap ini akan dilakukan pengumpulan informasi terhadap teori-teori yang nantinya akan digunakan dalam melakukan penelitian. Teori-teori yang dibutuhkan dalam melakukan penelitian ini bersangkutan dengan *procedural content generation* serta pendalaman mengenai algoritma *cellular automation* dan metode pengujian *Game User Experience Satisfaction Scale* untuk mengukur tingkat kepuasan pemain terhadap *game* yang dibuat.

### 3.2 Perancangan Game

Pada tahap ini akan dilakukan pembuatan *game design document* yang berisi tentang rancangan *game* yang nantinya akan dibuat seperti struktur permainan, alur *game* yang akan dibuat.

#### 3.2.1 Struktur Permainan

Judul: Berun Plague

Berun Plague adalah *game FPS Horror* dimana pemain akan menjelajahi map untuk mencari barang yang diminta oleh NPC.

##### 1. Player

Berun Plague merupakan *game single player* yang hanya bisa dimainkan oleh satu orang.

##### 2. Objectives

Objektif dalam *game* berun plague adalah untuk menyelesaikan *quest* yang diberikan NPC yaitu mencari tanaman obat yang berada dalam goa yang dipenuhi oleh *zombie*.

### 3. Procedures

- (a) Ketika pemain menjalankan *game* maka akan muncul halaman *splash screen*.
- (b) Setelah pemain melewati halaman *splash screen* akan muncul halaman menu yang berisi 4 pilihan yaitu *start*, *how to play*, *credit* dan *quit*.
- (c) Apabila pemain memilih *start* maka *game* akan dimulai dan akan muncul halaman prolog yang berisi cerita dari *game*.
- (d) Apabila pemain menekan *start* pada halaman prolog, pemain akan di arahkan ke kota.
- (e) Di kota pemain dapat berbicara pada NPC untuk mengambil *quest* dan membuka pintu yang tertutup.
- (f) Setelah pintu sudah terbuka pemain dapat masuk kedalam pintu dan nanti akan diarahkan menuju *scene* berisi map yang dibuat menggunakan PCG, di *scene* ini pemain dapat mulai untuk menyelesaikan *quest*.
- (g) Untuk menyelesaikan *game* pemain harus menyelesaikan *quest* yang diberikan yaitu mencari 3 tanaman obat.
- (h) Apabila darah pemain 0 maka pemain akan diarahkan ke dalam halaman *game over* yang berisi pilihan *retry* dan *quit*, dimana jika pemain memilih *retry* maka pemain akan mengulang kembali *game*, jika pemain memilih *quit* maka pemain akan diarahkan ke halaman main menu.

### 4. Rules

- (a) Pemain harus menggunakan *Google Cardboard* untuk menjalankan *game*.
- (b) Pemain dapat menggunakan *VR Controller* atau *Joystick* sebagai *input* dalam permainan.
- (c) Pemain tidak bisa keluar ketika sudah berada di dalam *game*.
- (d) Pemain harus mengumpulkan 3 tanaman obat untuk menyelesaikan permainan.
- (e) Pemain akan mati jika darah telah mencapai 0.
- (f) *Enemy* akan mati jika darah telah mencapai 0.

- (g) Jika pemain berhasil mengumpulkan 3 tanaman maka permainan selesai dan permainan selesai.

#### 5. Resources

- (a) Player Health: Menunjukkan indikator darah yang dimiliki oleh pemain, dimana darah akan berkurang jika terkena serang *enemy* dan akan bertambah jika pemain mengambil *healthpacks* yang berada pada **game**.
- (b) Player Ammo: Menunjukkan indikator peluru yang dimiliki pemain, dimana peluru akan berkurang jika *player* menembak dan jumlah peluru bisa ditambah dengan cara mengambil *item Ammo*.
- (c) Quest : Menunjukkan *progress quest* yang sedang dijalankan pemain.

#### 6. Conflict

Pemain harus menemukan tanaman obat yang berada pada map sambil mempertahankan hidupnya dari kejaran para *zombie*.

#### 7. Boundaries

Pemain hanya bisa berkeliling pada map yang sudah dibuat, serta pemain tidak bisa keluar di tengah permainan jika permainan sudah dimulai.

- 8. Outcome Pemain dapat dikatakan menang jika pemain sudah berhasil mengumpulkan 3 tumbuhan obat.

Dramatic Elements:

- (a) Challenge

Tantangan dalam *game* ini adalah bagaimana caranya pemain untuk tetap mempertahankan hidupnya sambil mencari *item* tanaman obat sambil di kejar oleh para *zombie*.

- (b) Play

Pada *game* ini *player* diberikan kebebasan dalam melakukan eksplorasi terhadap map yang dibangun sambil mencari tanaman obat yang dibutuhkan NPC.

- (c) Premise

Seorang tentara pemerintah yang bernama Joe yang ditus pemerintah untuk menyelesaikan permasalahan desa berun.

(d) Character

*Character* pada *game* ini adalah seorang tentara pemerintahan yang diutus ke suatu desa untuk membantu Dr.Sisca dalam menyelesaikan permasalahan wabah disana.

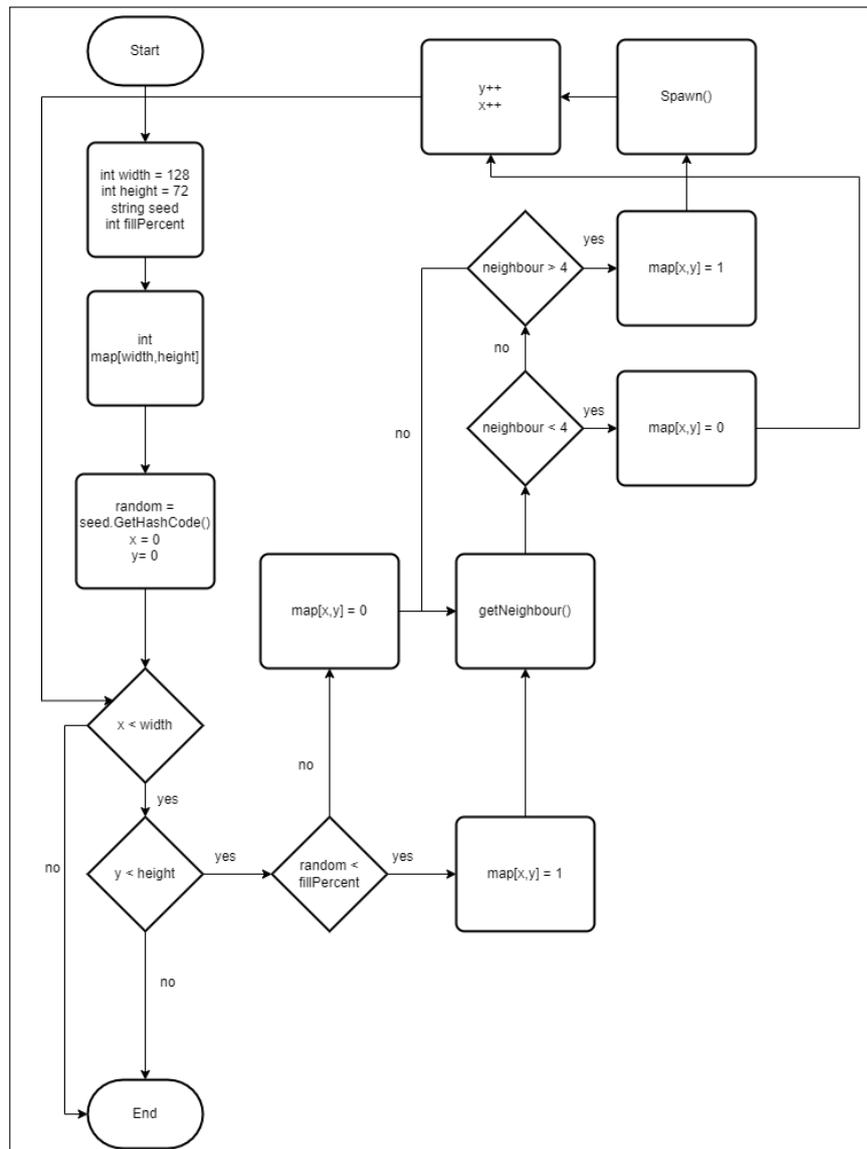
(e) Story

Pada tahun 1980 di desa Berun terjadi wabah yang berbahaya dimana wabah ini dapat menimbulkan kematian pada siapa saja yang terinfeksi, wabah ini hanya dapat disembuhkan menggunakan tanaman solanum yang diolah menjadi obat dimana keberadaan tanaman ini hanya berada pada goa mordor yang terdapat banyak *zombie*, lalu ada seorang doktor dari desa tersebut yang bernama dr.sisca menelpon kepada lembaga pemerintahan untuk mengirimkan seseorang dalam membantu doktor tersebut mencari tanaman solanum tersebut, lalu diutus lah seorang prajurit bernama joe untuk membantu dr.sisca dalam menemukan tanaman solanum.

### 3.2.2 Flowchart

Berikut merupakan *flowchart* yang digunakan untuk membangun *game horror* menggunakan algoritma *cellular automata*.

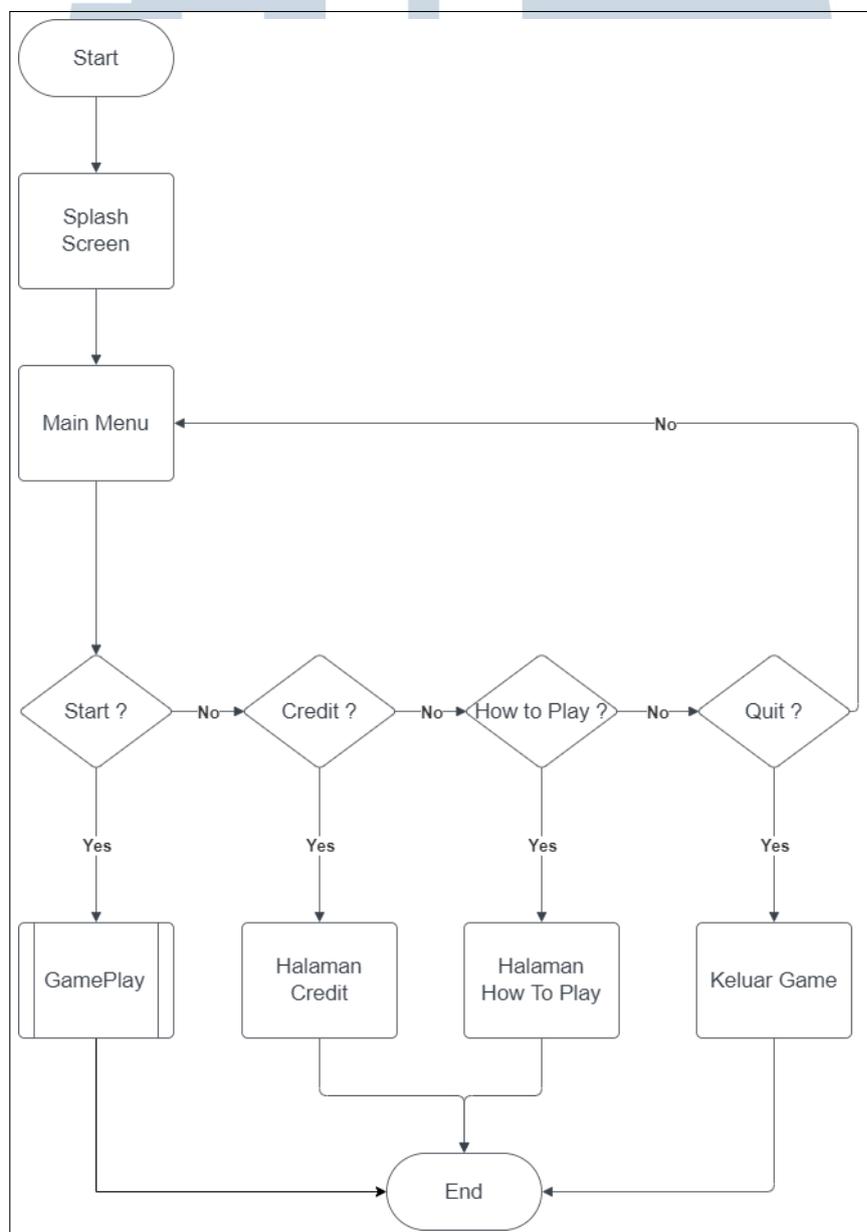
UMMN  
UNIVERSITAS  
MULTIMEDIA  
NUSANTARA



Gambar 3.1. Flowchart Map Generator

Pada Gambar 3.1 adalah *flowchart* dari proses pembuatan map menggunakan algoritma *cellular automata*, dimana proses diawali dengan inialisasi variable yang akan digunakan dalam pembuatan map, variable yang digunakan dalam pembuatan map antara lain *width*, *height* yang digunakan untuk mendefinisikan lebar dan tinggi map, lalu variable *seed* yang digunakan untuk perandoman pada map, variable *fillPercent* yang digunakan untuk presentase pengisian map, serta variable map yang memiliki struktur data array yang digunakan untuk merepresentasikan cell pada algoritma *cellular automata*, setelah inialisasi variable selanjutnya adalah inialisasi nilai variable random yang di dapatkan dari fungsi

seed.GetHashCode() dan looping yang digunakan untuk mengisi variable map, pada saat looping akan dilakukan pengecekan nilai random terhadap nilai *fillPercent* jika nilai random kurang dari *fillPercent* maka  $map[x][y]$  akan bernilai 1 jika tidak maka akan bernilai 0 setelah itu akan dilakukan pengecekan terhadap tetangga menggunakan fungsi *GetNeighbour*, jika jumlah tetangga lebih dari 4 maka *state cell* akan bernilai 1 dan jika kurang dari 4 maka akan bernilai 0, setelah itu akan dilakukan *spawn prefab* terhadap *cell-cell* yang memiliki nilai *state* 1 dengan menggunakan fungsi *spawn*.



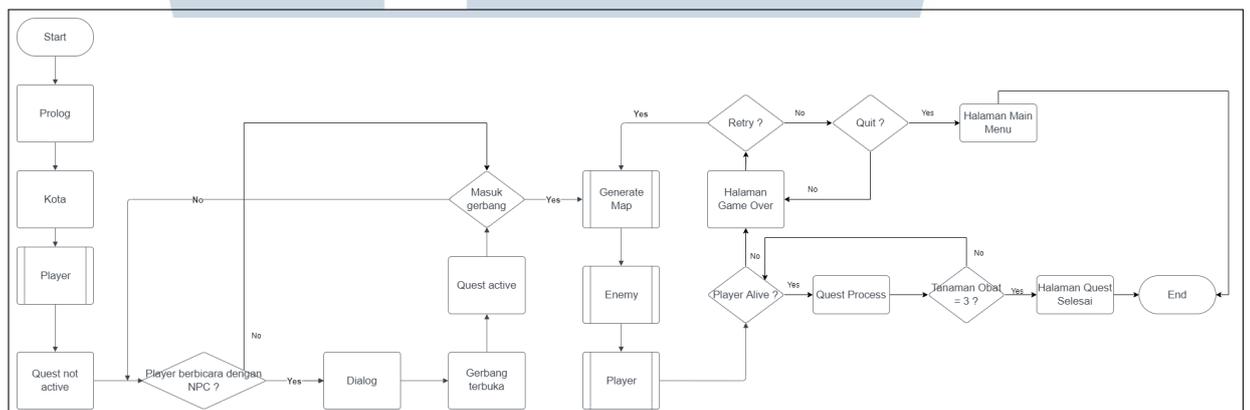
Gambar 3.2. Flowchart Main Menu

Pada Gambar 3.2 menunjukkan alur proses yang terjadi pada main menu di mana pada main menu pemain akan di sajikan beberapa pilihan yang dapat diakses yaitu *start*, *credit*, *how to play*, dan *quit*, dimana ketika pemain memilih *start* maka pemain akan dibawa kedalam modul *gameplay* yang artinya pemain akan mulai memainkan *game*.

Jika pemain memilih *credit* maka pemain akan diarahkan menuju halaman *credit* yang berisi nama dari pembuat *game*.

Jika pemain memilih halaman *how to play* maka pemain akan diarahkan menuju halaman *how to play* yang berisi *control* yang digunakan untuk bermain *game*.

Jika pemain memilih *quit* maka pemain akan keluar dari *game*.

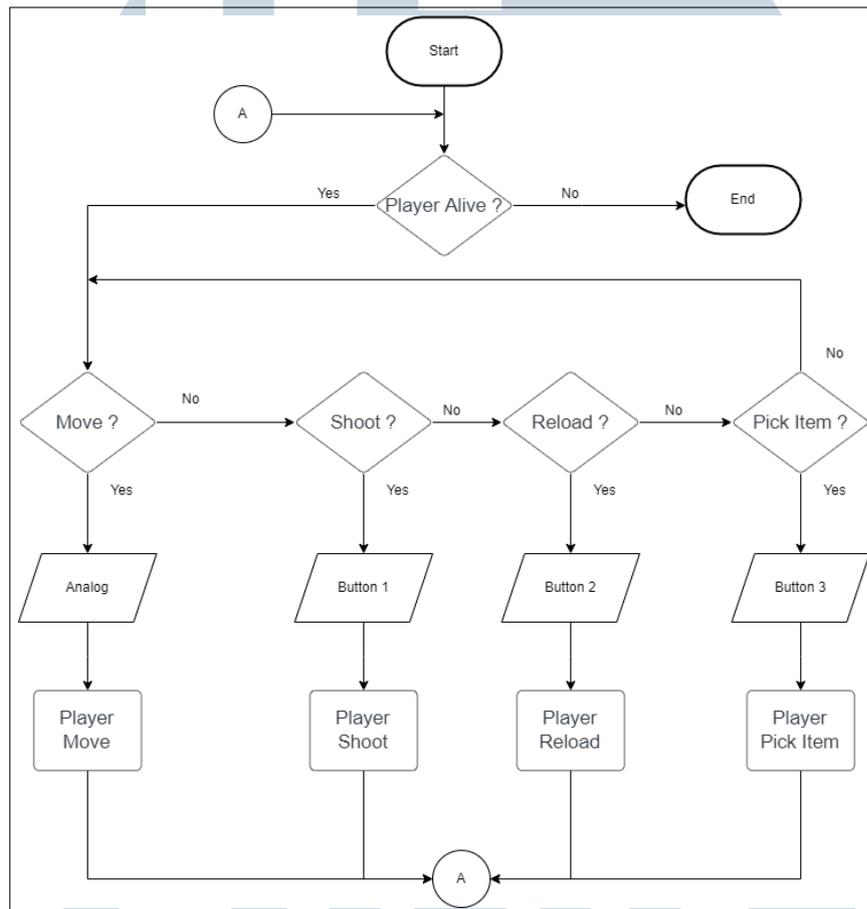


Gambar 3.3. Flowchart Gameplay

Pada Gambar 3.3 merupakan alur dari *gameplay*, dimana pada *gameplay* pemain pertama kali akan ditampilkan prolog dari *game* setelah prolog selesai selanjutnya pemain akan diarahkan ke *scene town* dimana pada *scene* ini pemain dapat mengambil tugas dengan cara berinteraksi dengan NPC, setelah pemain sudah berbicara pada NPC maka gerbang akan terbuka dan *quest* akan aktif, setelah gerbang terbuka pemain dapat masuk kedalam nya.

Ketika pemain masuk kedalam gerbang maka modul map generator akan dijalankan untuk pembuatan map, setelah map jadi maka akan dilanjutkan dengan modul *enemy* lalu modul *player* setelah modul *player* berjalan maka selanjutnya adalah pengecekan apakah pemain masih hidup atau tidak jika pemain masih hidup maka akan lanjut ke *quest process* yang artinya *player* mulai menjalankan *quest* setelah *player* menjalankan *quest* maka akan ada pengecekan apakah *item* tanaman obat sudah berjumlah 3 atau belum jika tanaman obat sudah berjumlah 3 maka pe-

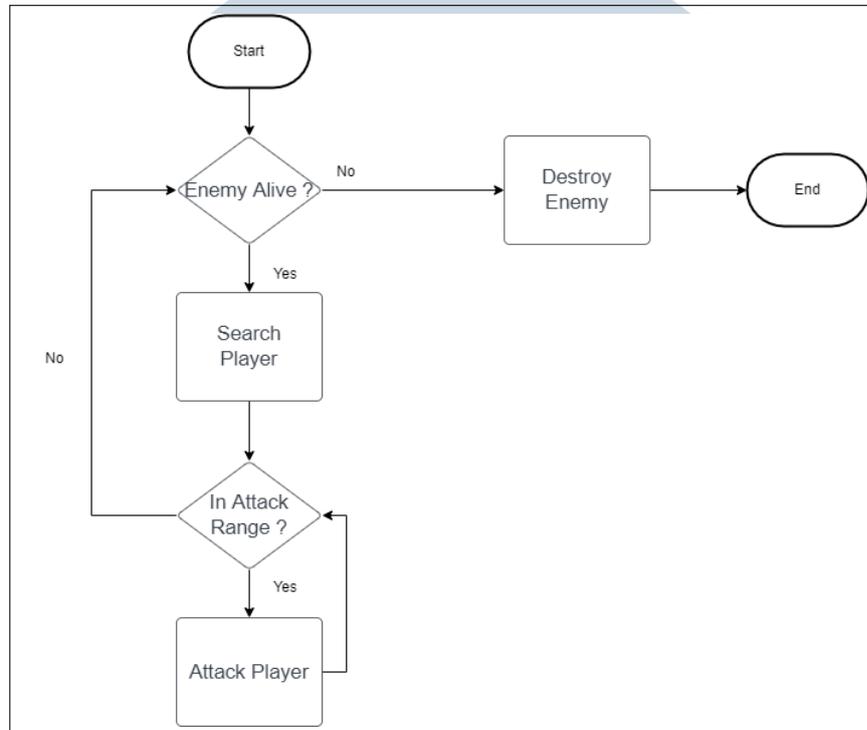
main akan diarahkan ke halaman quest selesai, jika pemain mati saat sedang menjalankan quest maka akan muncul halaman *game over*, setelah halaman *game over* muncul akan ada 2 pilihan yaitu *retry* dan *quit* jika pemain memilih *retry* maka akan diarahkan kembali ke map generator dan permainan akan di mulai kembali jika pemain memilih *quit* maka pemain akan diarahkan ke halaman main menu.



Gambar 3.4. Flowchart Player

Pada Gambar 3.4 adalah gambar dari flowchart *player*, dimana alur yang terjadi pada objek *player* akan diawali dengan pengecekan apakah *player* hidup atau tidak, jika *player* hidup maka *player* akan memiliki beberapa pilihan aktivitas yang dapat dilakukan yaitu *move*, *shoot*, *reload* dan *pick item* dimana jika *player* ingin bergerak maka *player* bisa menggunakan input analog untuk bergerak, jika *player* ingin menembak *player* bisa melakukan input button 1 untuk menembak, jika *player* ingin reload *player* bisa melakukan input button 2 untuk melakukan reload, jika *player* ingin mengambil *item* *player* melakukan input button 3 untuk mengambil *item*, jika *player* sudah melakukan salah satu aktivitas tersebut maka

akan dilakukan pengecekan kembali apakah *player* masih hidup atau tidak karena jika *player* sudah tidak hidup maka *player* tidak bisa melakukan aktivitas seperti bergerak, menembak, reload dan mengambil *item*.

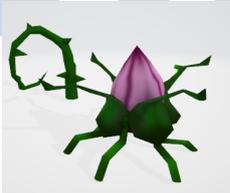


Gambar 3.5. Flowchart Enemy

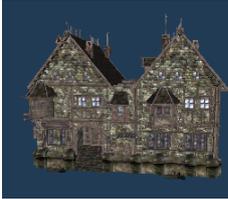
Pada Gambar 3.5 adalah *flowchart* dari *enemy*, dimana alur yang terjadi pada objek *enemy* akan diawali dengan pengecekan *enemy* hidup atau tidak jika *enemy* hidup maka *enemy* akan melakukan proses serach *player* untuk mencari *player*, dalam proses pencarian *enemy* akan melakukan pengecekan apakah *player* berada dalam jangkauan serang *enemy* atau tidak, jika *player* dalam jangkauan serang maka *enemy* akan melakukan serangan pada *player* jika *player* sudah tidak berada dalam jangkauan serang maka akan dilakukan kembali pengecekan apakah *enemy* hidup atau tidak jika *enemy* sudah tidak hidup atau mati maka *enemy* akan di destroy dan jika *enemy* masih hidup maka *enemy* akan melakuka kembali proses pencarian *player*.

### 3.2.3 Penggunaan Asset

Tabel 3.1. Daftar Aset Gambar

No	Nama	Gambar	Deskripsi	Sumber
1	3D Model NPC		NPC yang berada pada kota	assetstore.unity.com
2	3D Model Ammo		Item untuk menambah ammo	assetstore.unity.com
3	3D Model Flower		Item quest untuk menyelesaikan permainan	assetstore.unity.com
Lanjut pada halaman berikutnya				

Tabel 3.1 Daftar Asset (lanjutan)

No	Nama	Gambar	Deskripsi	Sumber
4	3D Model Health-packs		Item untuk menambah darah pemain	assetstore.unity.com
5	3D Model Pistol		Pistol yang digunakan pemain	assetstore.unity.com
6	3D Model Town		Kota yang digunakan dalam game	assetstore.unity.com
7	3D Model Zombie		Musuh dalam game	assetstore.unity.com
8	Blood Effect		Efek darah setiap musuh tertembak	assetstore.unity.com

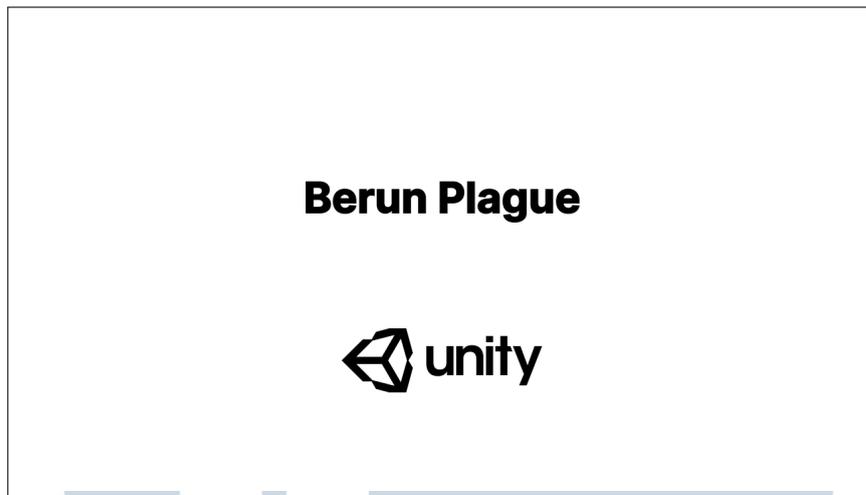
Lanjut pada halaman berikutnya

Tabel 3.1 Daftar Asset (lanjutan)

No	Nama	Gambar	Deskripsi	Sumber
9	MuzzleFlash Effect		Efek tembakan ketika pemain menembak	assetstore.unity.com
10	Gun Reload Sound Effect	-	Suara ketika reload	youtube.com
11	Gun Shot Dry Fire Sound Effect	-	Suara ketika pemain menembak tapi peluru habis	youtube.com
12	Cinematic Trailer Background Music	-	Background music pada game	youtube.com
13	Realistic Gunshot Sound Effect	-	Suara ketika pemain menembak	youtube.com
14	Zombie Sound Effects (free)	-	Suara zombie	youtube.com

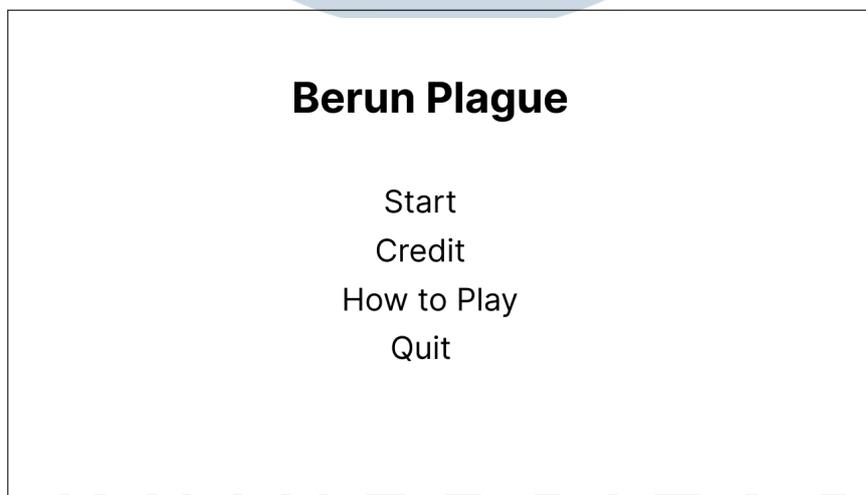
### 3.2.4 Desain Mockup

Berikut adalah desain *mockup* yang akan digunakan sebagai rujukan dalam melakukan pembuatan *game*. Pada Gambar 3.6 merupakan tampilan *splash screen* yang akan muncul pertama kali ketika pemain menjalankan *game*.



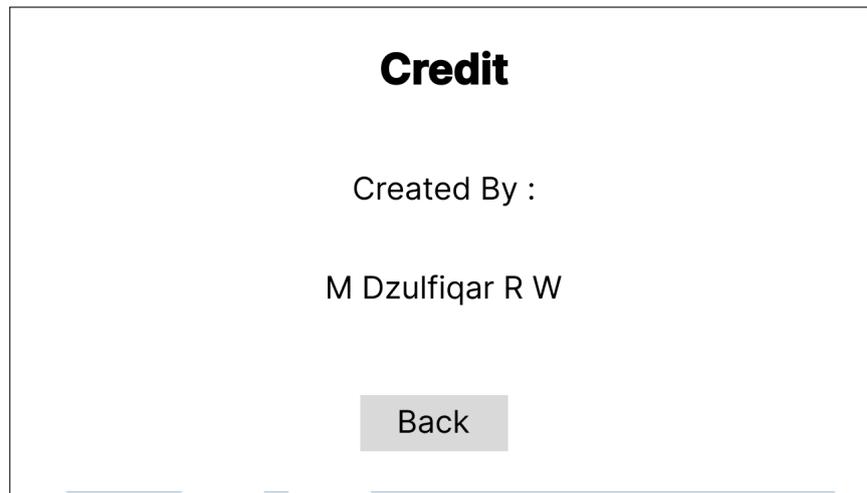
Gambar 3.6. Mockup Splash Screen

Setelah pemain melewati halaman *splash screen* pemain akan diarahkan menuju halaman main menu. Pada gambar 3.7 merupakan tampilan main menu dimana pada main menu akan terdapat 4 pilihan yaitu *start*, *credit*, *how to play*, dan *quit*.



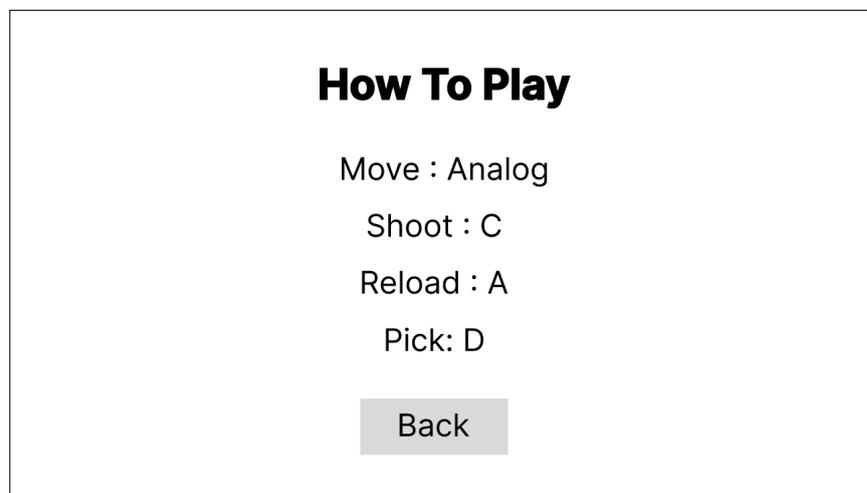
Gambar 3.7. Mockup Main Menu

Jika pemain memilih untuk menekan tombol *credit* maka pemain akan diarahkan kedalam halaman kredit seperti yang bisa dilihat pada Gambar 3.8 dimana pada halaman *credit* berisi nama pembuat *game* serta tombol *back* untuk kembali ke main menu.



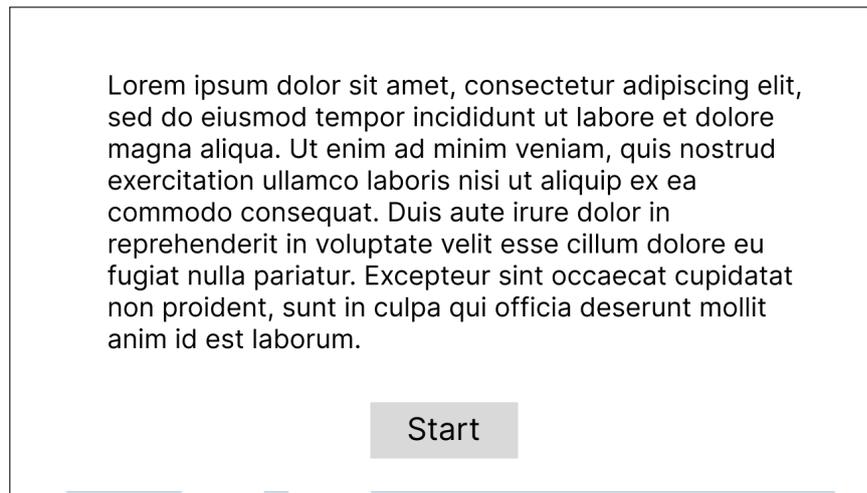
Gambar 3.8. Mockup Credit

Jika pemain menekan tombol *how to play* maka pemain akan diarahkan kedalam halaman *how to play* seperti yang bisa dilihat pada Gambar 3.9 dimana pada halaman *how to play* berisi informasi mengenai *controller* untuk memainkan *game* seperti analog digunakan untuk bergerak, tombol C untuk menembak, tombol A untuk *reload*, dan tombol D untuk mengambil *item* serta terdapat tombol *back* untuk kembali ke halaman main menu.



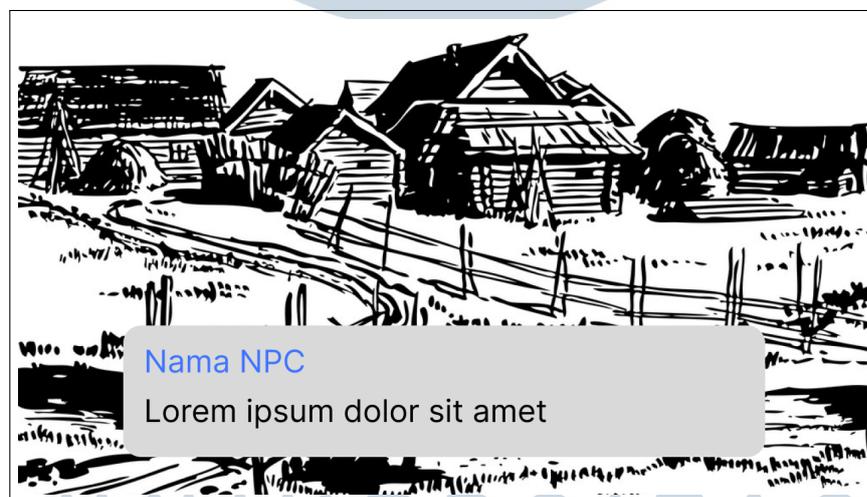
Gambar 3.9. Mockup How To Play

Jika pemain memilih untuk menekan tombol *start* maka pemain akan diarahkan kedalam halaman prolog dimana halaman prolog berisi tentang cerita awal dari *game*, halaman prolog dapat dilihat pada Gambar 3.10



Gambar 3.10. Mockup Prolog

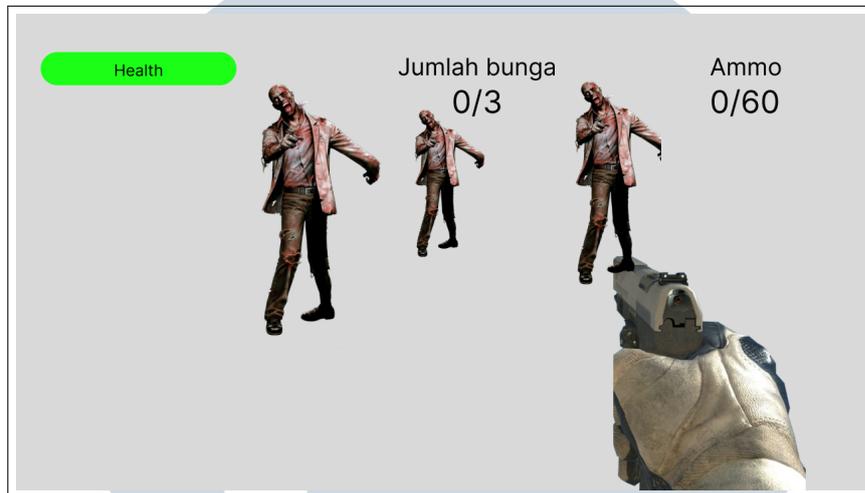
Jika pemain menekan tombol *start* pada halaman prolog maka pemain akan diarahkan ke halaman *town* dimana pada halaman *town* ini *game* sudah dimulai dan pemain diharuskan berbicara kepada NPC dan akan muncul dialog antara pemain dan NPC untuk dialog dapat dilihat pada Gambar 3.11.



Gambar 3.11. Mockup Dialogue

Setelah pemain melakukan dialog dengan NPC maka pemain akan diarahkan kepada *scene gameplay* dimana pada *scene gameplay* ini pemain sudah mulai dapat bermain dan mencari tanaman obat yang disuruh oleh NPC, *scene gameplay* dapat dilihat pada Gambar 3.12 dimana pada *scene gameplay* terdapat beberapa informasi yang digunakan untuk bermain yaitu *health bar* yang digunakan untuk mengetahui jumlah darah yang dimiliki oleh pemain, jumlah bunga yang berguna

untuk mengetahui berapa jumlah tanaman yang telah didapatkan, dan *ammo* yang berguna untuk mengetahui jumlah *ammo* serta sisa *ammo* yang berada dalam pistol pemain.



Gambar 3.12. Mockup Gameplay

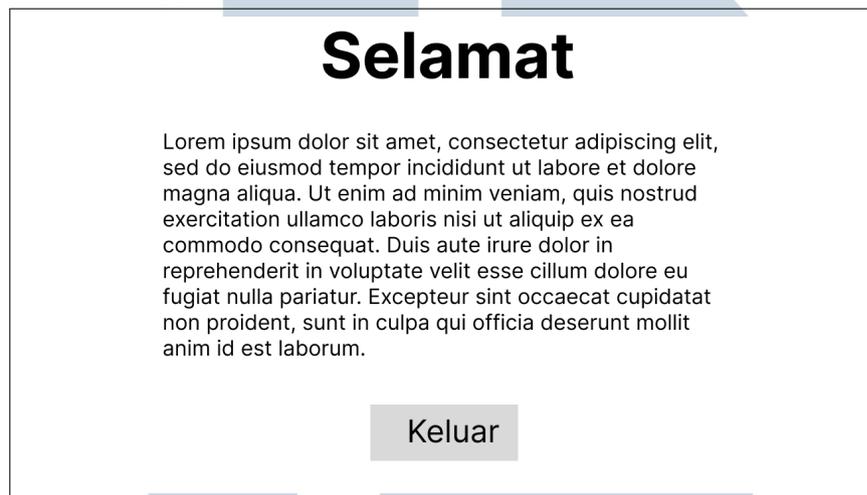
Jika di dalam permainan pemain mengalami Kematian akibat darah habis maka pemain akan diarahkan kepada halaman *game over*, Pada halaman *game over* terdapat dua pilihan tombol yaitu *retry* dan *quit* dimana jika pemain memilih *retry* maka pemain akan kembali menuju *scene town* dan mengulangi permainan, jika pemain memilih *quit* maka pemain akan diarahkan ke dalam halaman main menu, *scene game over* dapat dilihat pada Gambar 3.13



Gambar 3.13. Mockup Game Over

Jika pemain berhasil menemukan ketiga tanaman maka pemain akan di-

arahkan kedalam *scene Game Finish* dimana pada *scene* ini ada tulisan selamat dan juga epilog dari cerita yang menceritakan lanjutan cerita dari *game*, dan terdapat juga tombol keluar untuk kembali ke main menu, tampilan *game finish* dapat dilihat pada Gambar 3.14



Gambar 3.14. Mockup Game Finish

### 3.3 Pembuatan Game

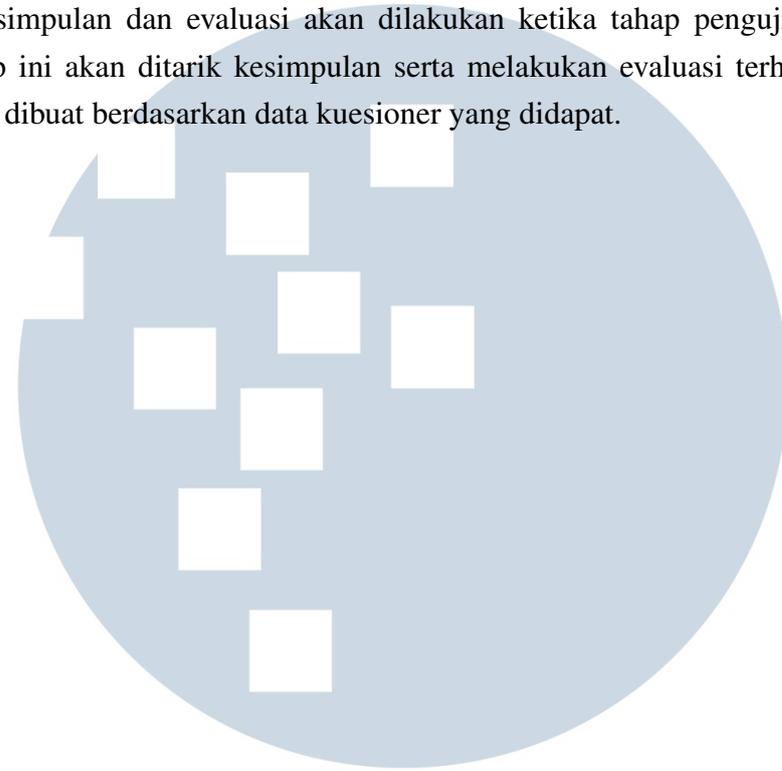
Pada tahap ini akan dilakukan pembuatan *game* menggunakan *game engine* unity dan bahasa pemrograman C# sesuai dengan rancangan yang telah dibuat pada tahap perancangan.

### 3.4 Pengujian Game

Pengujian akan dilakukan dengan cara memberikan para responden *game* yang telah dibuat untuk dimainkan, setelah *game* telah dimainkan responden akan diminta untuk mengisi kuesioner yang telah dibuat sesuai dengan metode *Game User Experience Satisfaction Scale* dimana kuesioner yang telah diisi oleh para pemain nantinya akan digunakan untuk mengukur tingkat kepuasan pemain terhadap *game* yang dibangun dengan *procedural content generation* menggunakan *cellular automata*, jumlah sample yang akan diambil pada proses pengujian *game* ini adalah 30 sample sesuai dengan metode penelitian kombinasi yang telah dilakukan oleh Sugiono[25].

### 3.5 Kesimpulan dan Evaluasi

Kesimpulan dan evaluasi akan dilakukan ketika tahap pengujian selesai, pada tahap ini akan ditarik kesimpulan serta melakukan evaluasi terhadap *game* yang telah dibuat berdasarkan data kuesioner yang didapat.



# UMMN

UNIVERSITAS  
MULTIMEDIA  
NUSANTARA