

BAB V. IMPLEMENTASI DAN PENGUJIAN

1.1 Implementasi

Pengembangan pada tahap awal lebih difokuskan ke pengumpulan asset model 3D. Model-model ini kemudian dimasukkan ke dalam project Unity baru. Sebelum pengimplementasian model pemain dan model musuh, dibuatlah *terrain* lingkungan dengan menambahkan objek *terrain* ke dalam project Unity. Objek *terrain* ini “dilukis” sehingga tampak bukit-bukitan dan sebuah parit. Setelah itu, barulah model pemain dan model musuh dimasukkan.

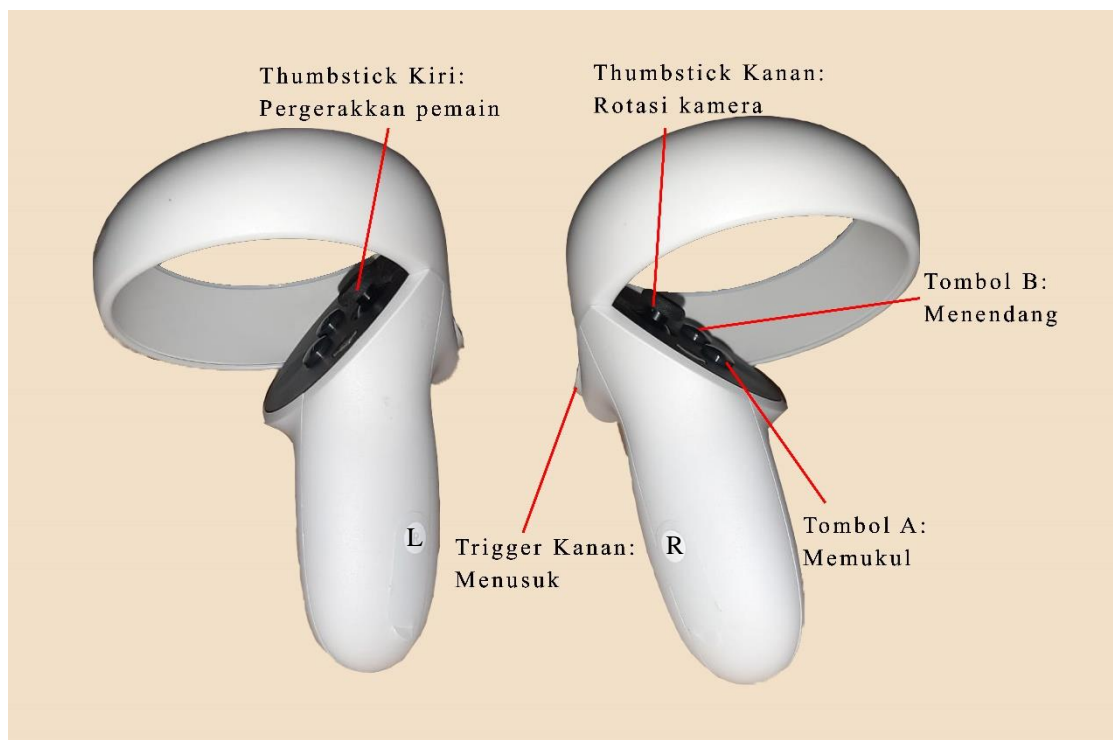
Di samping itu, diterapkanlah animator untuk kedua model tersebut. Animator merupakan alur animasi yang dapat disesuaikan berdasarkan kondisi. Masing-masing model memiliki animator tersendiri. Untuk model pemain, animasi terdiri dari animasi berdiri atau *idle*, animasi berjalan, animasi menusuk, animasi menendang, animasi memukul, dan animasi mati. Untuk model musuh, animasi terdiri dari animasi lari, animasi terpukul, animasi menyerang balik, animasi berjalan lemah, dan animasi mati.

Setelah penerapan model dan animator untuk model, perhatian dialihkan ke interaksi *User Interface*. Untuk menghindari ukuran file yang terlalu besar, maka dibuatlah dua *scene* yang membedakan tampilan menu-menu utama dari tampilan permainan dan tampilan *Game Over*. Interaksi antar satu *scene* dengan yang lainnya diatur oleh dua buah *custom script* yang saling memanggil dan menerima informasi dari satu sama lain. Di antara fungsi yang diterapkan dari *custom script* ini selain dari menampilkan dan menyembunyikan opsi yang tidak diperlukan ialah pemunculan data-data skor tertinggi yang diambil dari beberapa sesi permainan berdasarkan tingkat kesulitannya, pengaturan untuk volume *background music* dan *sound effects*, pengaturan penampilan timer dan bercak darah, dan penentuan fungsional permainan yang didasarkan dari opsi tingkat kesulitan yang dipilih.

Karena model musuh akan muncul secara *random* tergantung tingkatan, dibuatlah suatu *custom script* yang mendiktekan jumlah generasi dan posisi individual model musuh. Jumlah nyawa musuh juga didiktekan oleh tingkat kesulitan serta jenis serangan apa yang digunakan oleh model pemain. Jika nyawa musuh mendekati nol, animasi berlari model akan berubah menjadi animasi

berjalan lemah sebelum akhirnya memainkan animasi mati ketika nyawa mencapai nol.

Untuk kamera dan kontrol karakter, diimplementasikan prefab `OVRPlayerController` yang didapat dari men-download Oculus Integration di Unity Asset Store. `OVRPlayerController` sudah memiliki *built-in* kontrol pada kamera, di mana kamera ini memiliki prefab sendiri yaitu prefab `OVRCameraRig` yang merupakan *child* langsung dari prefab `OVRPlayerController`. Supaya model pemain ikut bergerak bersama dengan kamera, model pemain dijadikan sebagai *child* dari prefab `OVRPlayerController`. Sebuah *custom script* dibuat untuk mengontrol interaksi model seperti menendang atau memukul.



Gambar 5.1 Foto kontroller *Oculus Quest 2* beserta fungsi setiap tombol

Sesuai dengan Gambar 5.1, tombol A yang terletak pada kontroler kanan berfungsi sebagai tombol memukul. Tombol B di kontroler yang sama berfungsi untuk membuat karakter pemain menendang, sementara tombol *Trigger* kontroler kanan membuat karakter pemain menusuk dengan bayonette. *Thumbstick* kontroler kanan berfungsi untuk memutar posisi pandang pemain. *Thumbstick* kontroler kiri berfungsi untuk menjalankan pemain.

1.2 Screenshot



Gambar 5.2 Tampilan Main Menu

Halaman pertama yang akan menyambut pemain tentunya adalah menu utama dengan tampilan seperti yang dapat dilihat pada Gambar 5.2. Dalam menu utama ini, terdapat enam tombol dengan fungsinya masing masing. Tombol-tombol tersebut ialah *Play* (bermain), *Controls* (Kontrol), *Option* (Pengaturan), *Scoreboard* (Papan skor), *Credits*, dan *Exit* (Keluar). Fungsi dari setiap tombol akan dijabarkan pada gambar-gambar selanjutnya kecuali untuk tombol *Exit*. Tombol ini berfungsi untuk keluar dari game.



Gambar 5.3 Tampilan menu pilihan tingkat kesulitan

Ketika memilih tombol *Play* pada menu utama, akan muncul tampilan pemilihan tingkat kesulitan yang dapat dilihat pada Gambar 5.3. Tingkatan kesulitan yang ada ialah *Easy* (mudah), *Medium* (sedang), dan *Hard* (sulit). Tingkat kesulitan akan mempengaruhi jumlah musuh yang ada, banyak nyawa yang dimiliki oleh musuh, jumlah skor yang didapat, dan lama waktu bermain. Tombol *Back* (kembali) berfungsi untuk mengembalikan ke menu utama.

Untuk spesifikasi perbedaan antara setiap tingkat kesulitan, dapat dilihat pada Tabel 5.1 di bawah ini.

No.	Tingkat Kesulitan	Jumlah Musuh	Nyawa Musuh	Skor yang didapat				Timer
				Memukul	Menendang	Menusuk	Musuh Mati	
1	<i>Easy</i> (Mudah)	30	5000 HP	1000 pt	2000 pt	5000 pt	10000 pt	7 Menit
2	<i>Medium</i> (Sedang)	20	10000 HP	500 pt	1000 pt	1500 pt	5000 pt	5 Menit

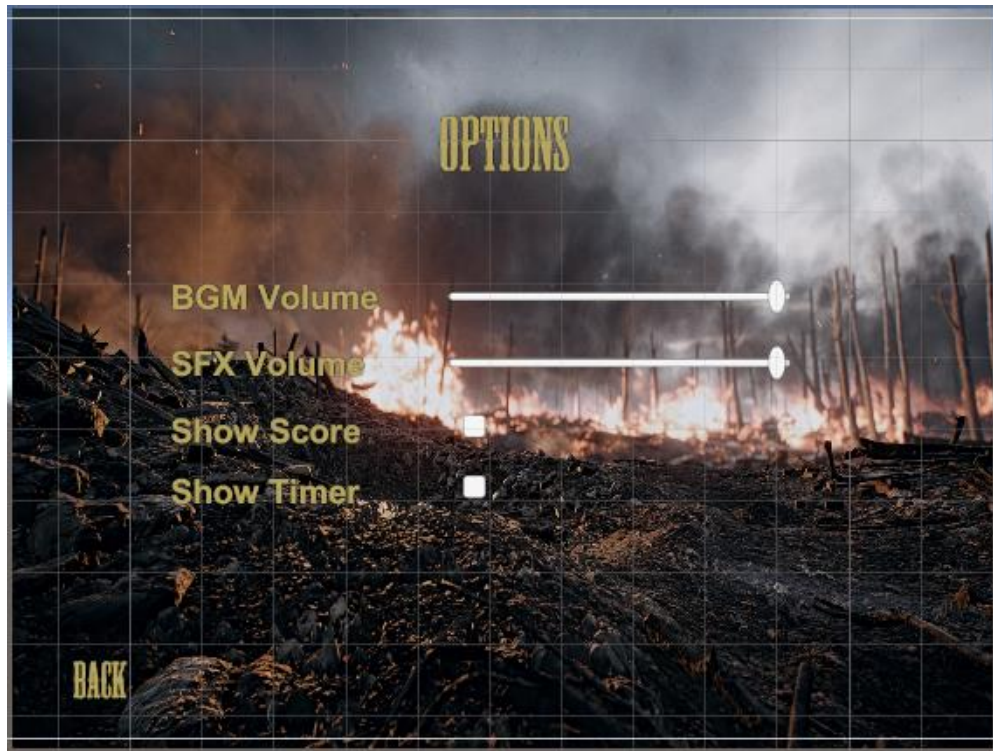
3	<i>Hard</i> (Sulit)	15	15000 HP	100 pt	200 pt	500 pt	1000 pt	3 Menit
---	------------------------	----	-------------	--------	--------	--------	------------	---------

Tabel 5.1 Spesifikasi perbedaan setiap tingkat kesulitan



Gambar 5.4 Tampilan panduan

Untuk tombol *Controls*, akan ditampilkan instruksi dalam memainkan game ini. Pada Gambar 5.4, dapat dilihat bahwa tombol A pada controller *Oculus Quest 2* kanan digunakan untuk memukul, tombol B untuk menendang, tombol *trigger* untuk menusuk, dan *thumbstick* kanan untuk rotasi kamera. Pada controller *Oculus Quest 2* kiri, *thumbstick* digunakan untuk berjalan. Tombol *Back* (kembali) berfungsi untuk mengembalikan ke menu utama.



Gambar 5.5 Tampilan menu pengaturan

Pada Gambar 5.5, ada 4 pengaturan yang dapat disesuaikan. *Slider BGM Volume* dan *SFX Volume* digunakan untuk mengatur volume *background music* dan *sound effects*. *Toggle Show Score* digunakan untuk menampilkan atau menghilangkan tampilan UI *score* saat memainkan game. *Toggle Show Timer* memiliki fungsi yang sama tetapi sebagai ganti *score*, *toggle* ini mengatur UI *timer* saat bermain. Tombol *Back* (kembali) berfungsi untuk mengembalikan ke menu utama.



No.	Name	Date	Score
1.	AAAAAA	01/01/2021	000000
2.	AAAAAA	01/01/2021	000000
3.	AAAAAA	01/01/2021	000000
4.	AAAAAA	01/01/2021	000000
5.	AAAAAA	01/01/2021	000000
6.	AAAAAA	01/01/2021	000000
7.	AAAAAA	01/01/2021	000000

BACK

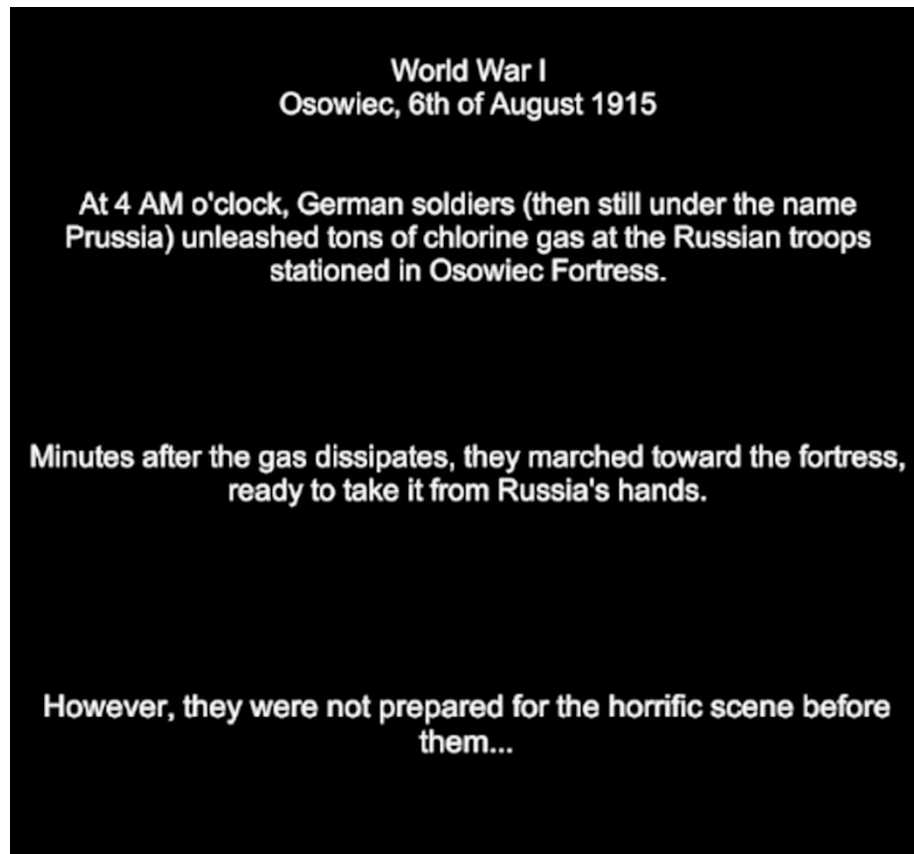
Gambar 5.6 Tampilan *Scoreboard*

Tampilan *Scoreboard* pada Gambar 5.6 menunjukkan papan skor untuk tingkat kesulitan mudah. Selain papan skor untuk tingkat mudah, terdapat juga papan skor untuk tingkat kesulitan sedang dan sulit. Tombol *Back* (kembali) berfungsi untuk mengembalikan ke menu utama.



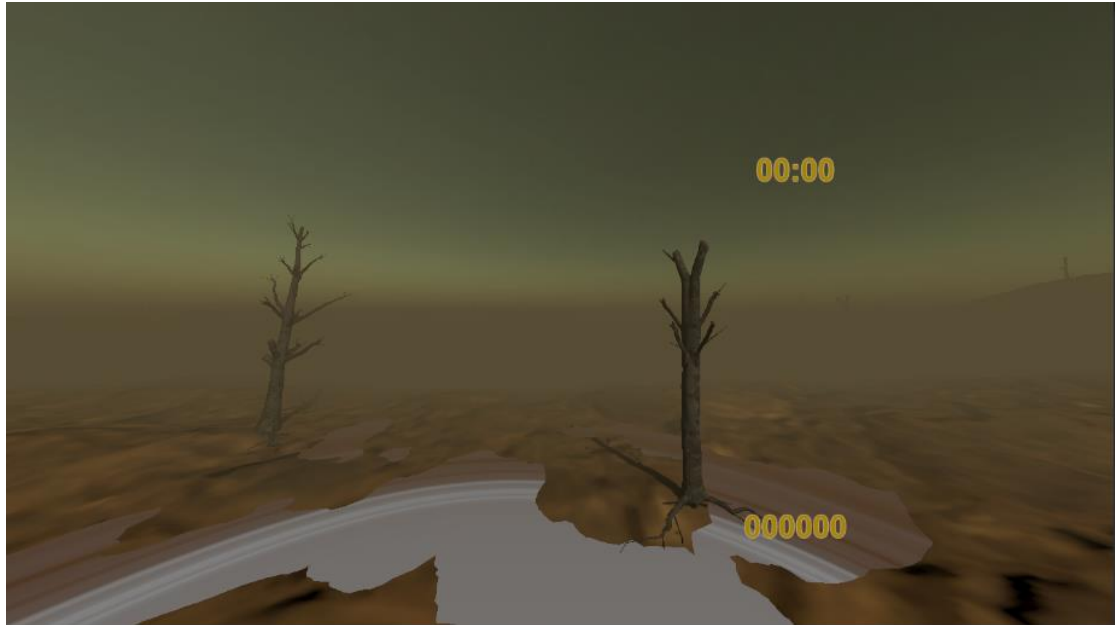
Gambar 5.7 Tampilan *Credits*

Credits pada **Gambar 5.7** menampilkan nama pembuat game beserta sumber-sumber dari asset-asset yang dipakai dalam game ini. Tombol *Back* (kembali) berfungsi untuk mengembalikan ke menu utama.



Gambar 5.8 Narasi *intro*

Gambar 5.8 menampilkan narasi awal sebelum bermain sebagai semacam pengenalan ke peristiwa sejarah yang menjadi latar belakang dari game ini, yaitu peristiwa *Attack of the Dead Men*. Narasi ditunjukkan menggunakan 3 “*slide*” yang akan *fade in* dan *fade out* sebelum akhirnya memulai permainan.



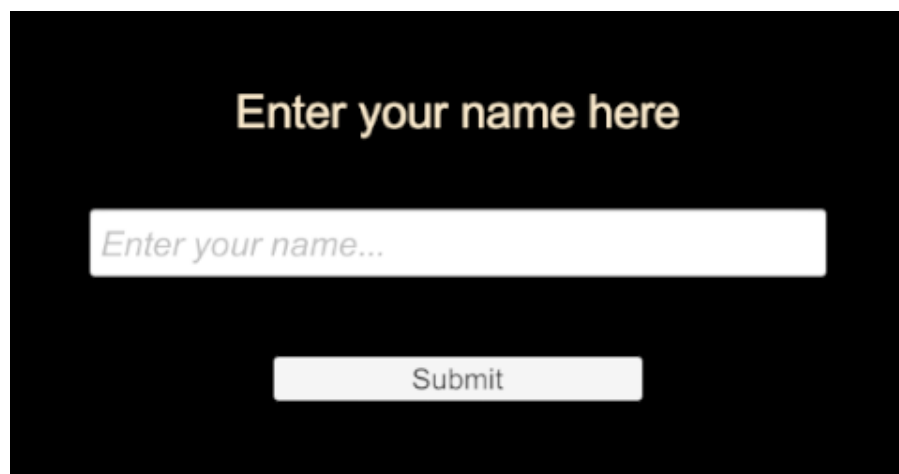
Gambar 5.9 Tampilan saat bermain

Gambar 5.9 merupakan tampilan saat bermain dengan *timer* dan *score* yang dimunculkan. *Environment* memiliki kesan berkabut untuk mensimulasikan kabut gas klorin yang diluncurkan oleh pihak Jerman ke pihak Russia. *Landscape* yang berbatu sesuai dengan keadaan perang pada saat itu.



Gambar 5.10 Narasi *outro*

Sama seperti Gambar 5.8 yang menampilkan narasi *intro*, Gambar 5.10 menampilkan narasi *outro* dengan menggunakan metode *fade in* dan *fade out*. Narasi ini akan muncul setelah *timer* permainan mencapai 0.



Gambar 5.11 Tampilan input nama

Setelah narasi *outro* selesai ditampilkan, tampilan pada Gambar 5.11 akan muncul. Pemain diminta untuk memasukkan nama mereka sebelum melakukan submit. Nama ini akan tersimpan di data yang akan tampil di papan skor.



Gambar 5.12 Tampilan Game Over

Setelah memasukkan nama dan *submit*, akan muncul tampilan seperti yang ada pada Gambar 5.12 yang merupakan tampilan *Game Over*. Pada tampilan ini, total skor yang didapat akan ditampilkan beserta dua buah tombol. Tombol-tombol ini adalah tombol *Retry* dan tombol *Return to Main Menu*. Tombol *Retry* mengulang permainan dengan menggunakan tingkat kesulitan yang sama. Tombol *Return to Main Menu* mengembalikan ke menu utama.

1.3 Kendala

Pada awal pengembangan game, terjadi kendala pada rigging model. Secara spesifik, model-model yang didapat tidak memiliki rig atau rig tidak sesuai dengan asset animasi sehingga model memiliki kesan *stretchy*. Selain itu, model untuk musuh tidak berada di posisi *anchor* pusat sehingga model memanjang dari bentuk aslinya. Untuk terrain, pembuatan masih sangat amatir sehingga terkesan simple. Menanggapi ini, saya pada akhirnya harus membeli model-model dan terrain baru

dikarenakan keterbatasan waktu dan pengetahuan dalam rigging dan membuat terrain.

Kendala selanjutnya terjadi ketika mengimplementasikan kamera VR. Pada awalnya, kamera berada di belakang model, sehingga seakan-akan kamera bertindak sebagai sudut pandang ketiga sementara seharusnya kamera bertindak sebagai sudut pandang pertama (posisi kamera persis di depan mata/kepala model). Menanggulangi ini, model pemain akhirnya dijadikan sebagai *child* dari kamera VR. Sebagai perbandingan sebelumnya, kamera VR merupakan *child* dari model pemain.

Kendala juga terjadi pada *movement* model pemain. Animasi dan rotasi model pemain ketika di-*test* menggunakan *device* VR berfungsi dengan baik namun *movement* dari model tidak berfungsi sama sekali. Pada awalnya, pergerakan model menggunakan *script* kode manual yang memanipulasikan input dari *thumbstick* pada kontroler VR. Namun, hal tersebut tidak menyelesaikan masalah yang dihadapi. Sebaliknya, terjadi *clipping* pada model di mana model jatuh tembus dari *terrain* tanah yang ada. Solusi selanjutnya adalah mengimplementasikan prefab *OVRPlayerController* yang didapat dari *men-download* Oculus Integration yang tersedia di web Unity Asset Store. Prefab *OVRPlayerController* ini memang diperuntukan sebagai kontrol *movement* yang “disambungkan” ke *joystick* kontroler VR. Selain mengimplementasikan prefab *OVRPlayerController*, komponen *Rigidbody* dan *Capsule Controller* dimasukkan dalam prefab *OVRPlayerController* dan model pemain untuk menghindari terjadinya *clipping* dari tanah.

1.4 *Testing* dan Testimoni Reponden

Untuk tahap awal testing, dilakukan testing dengan metode *Black Box* untuk menguji apakah ada error atau tidak. Selama *testing*, terdapat beberapa error yang umumnya terjadi pada objek kamera VR. Hal ini terjadi karena beberapa faktor seperti komposisi *Rigidbody* dan *Collider* yang salah, pembuatan objek model pemain sebagai *child* dari kamera VR, serta pemosisian kamera VR yang selalu berpindah dari posisi yang seharusnya. Dari sini, dicarilah solusi untuk menghilangkan penyebab error tersebut.



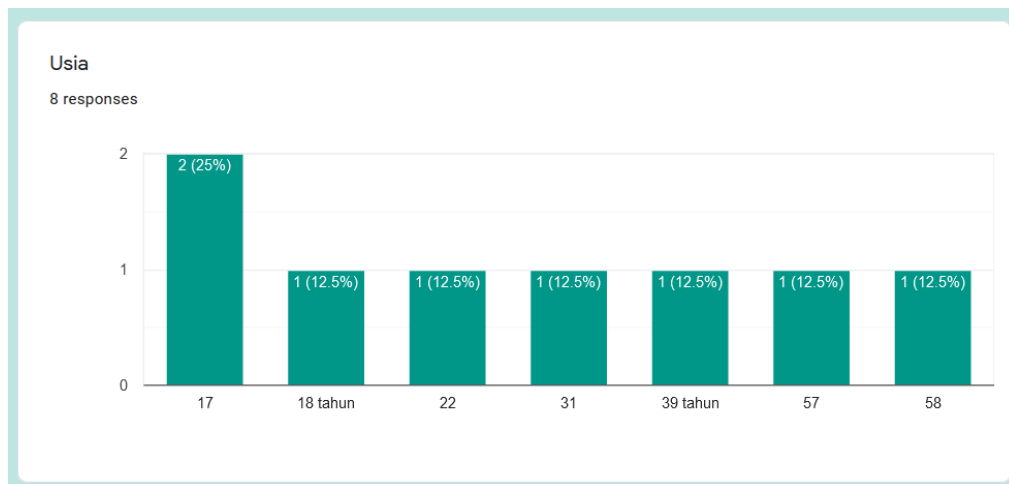
Gambar 5.13 Dua responden sedang bergantian dalam menggunakan alat VR



Gambar 5.14 Seorang responden sedang mencoba menggunakan alat VR untuk bermain game *Attack of the Dead Men*

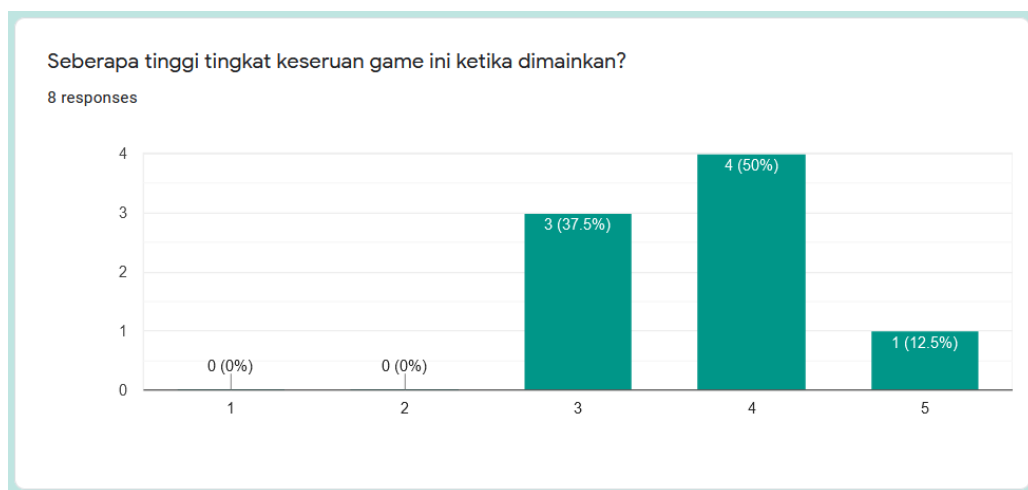
Setelah dilakukan *Build* dan install file .apk ke alat VR, maka dilakukanlah semacam *Beta Testing* ke beberapa responden tertentu dengan rentang usia 17-60 tahun, baik yang sudah pernah menggunakan alat VR maupun yang belum. Dokumentasi *Beta Testing* dapat dilihat pada Gambar 5.13 dan Gambar 5.14. Setelah responden-responden tersebut sudah memainkan game, dibagikanlah kuesioner berupa *Google Form* untuk mendapatkan data *feedback* dari responden.

Berikut ini hasil dari pengumpulan *feedback* responden-responden game ini:

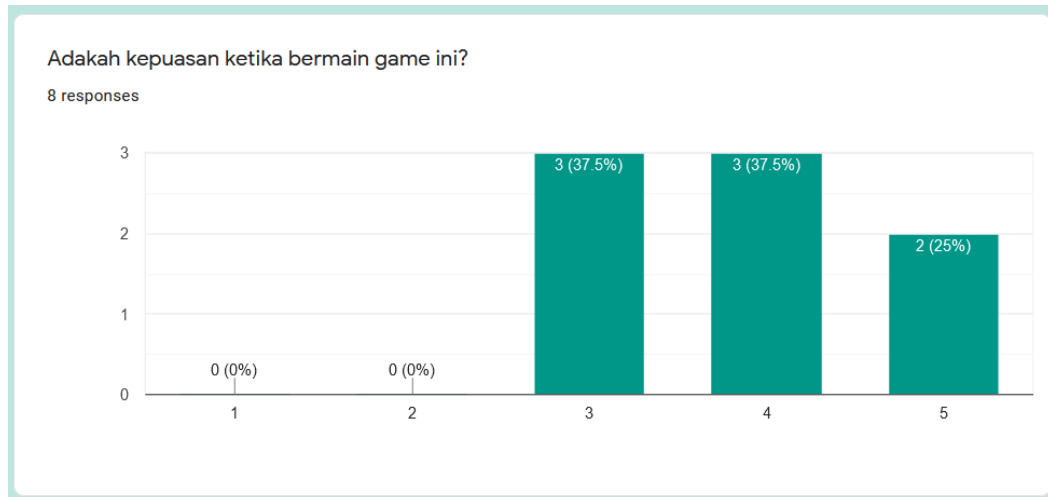


Gambar 5.15 Statistika usia

Berdasarkan statistika pada Gambar 5.15, mayoritas responden berada pada rentang usia antara 13 tahun sampai 35 tahun.

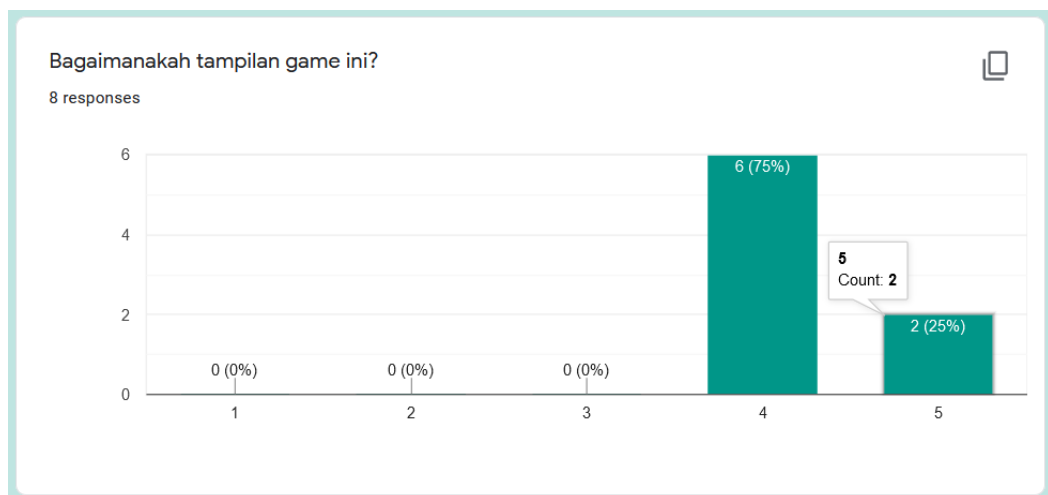


Gambar 5.16 Statistika tingkat keseruan



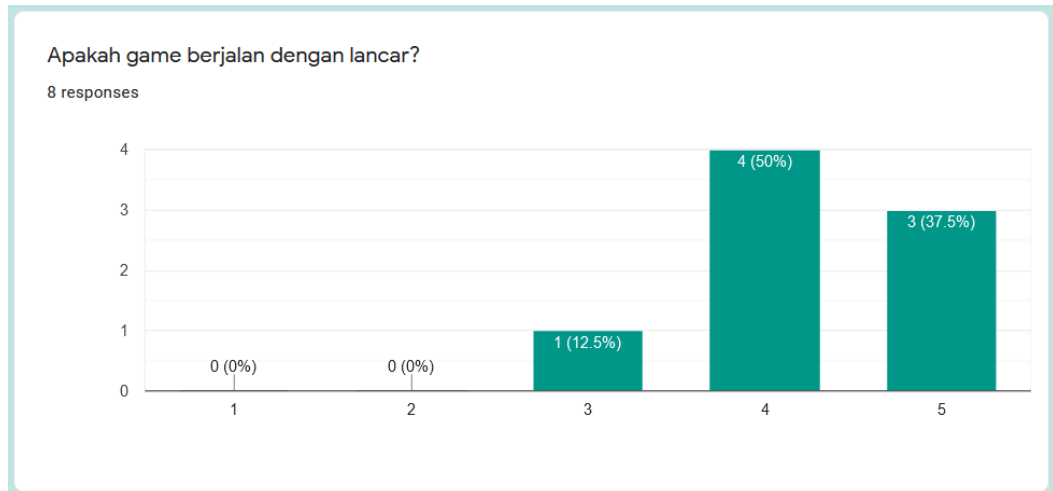
Gambar 5.17 Statistika tingkat kepuasan

Statistika pada Gambar 5.16 dan Gambar 5.17 memiliki korelasi dengan tingkat kejenuhan responden saat bermain. Di sini terlihat bahwa tingkat kejenuhan relatif rendah, dengan 4 responden menyatakan seru dan 1 responden menyatakan sangat seru sementara 3 responden menganggap biasa saja. Dari sisi kepuasan, 3 responden menyatakan puas dan 2 lagi menyatakan sangat puas sementara 3 responden merasa biasa saja.



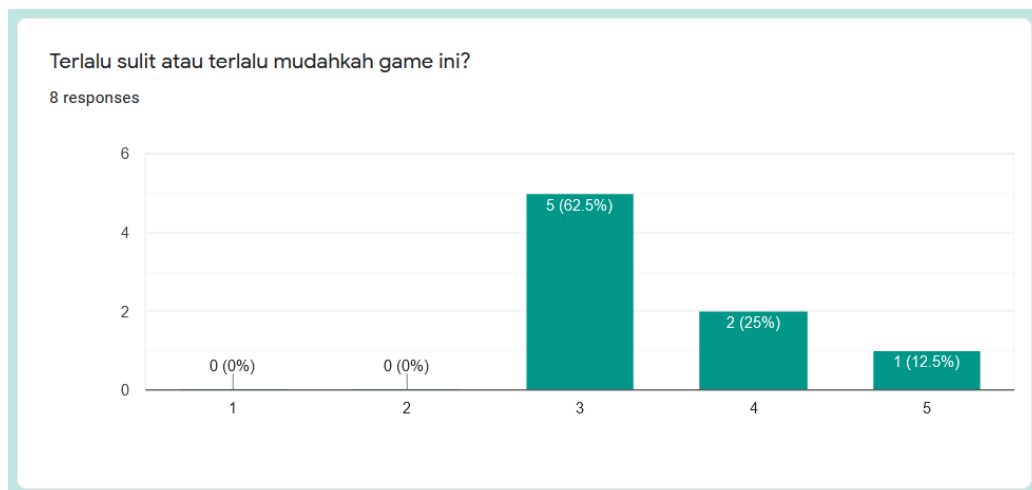
Gambar 5.18 Statistika tampilan

Dari sisi tampilan sesuai dengan Gambar 5.18, 6 responden menganggap tampilan bagus dan 2 lainnya menganggap sangat bagus.



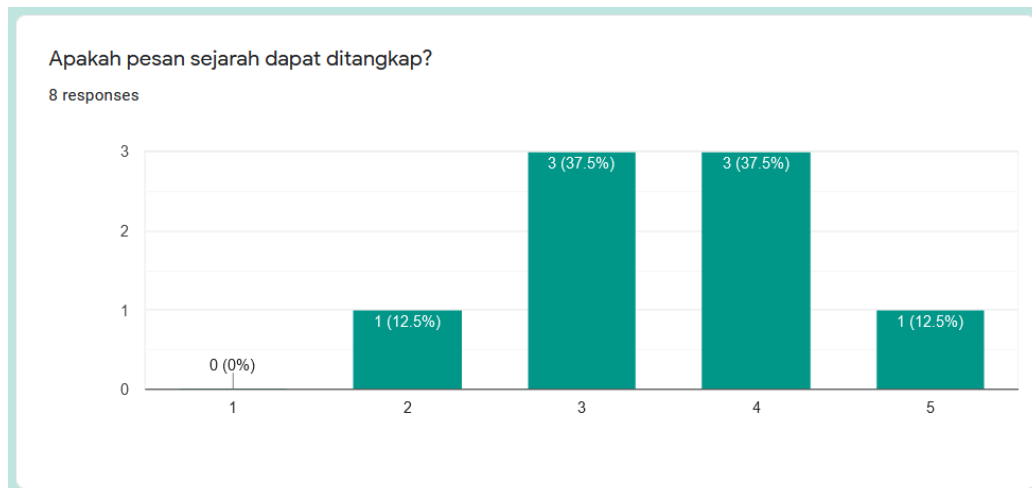
Gambar 5.19 Statistika kelancaran game

Game berjalan dengan cukup lancar sesuai dengan testimoni pada statistika Gambar 5.19 di mana hanya 1 responden menyatakan kelancaran game setengah-setengah. 4 responden menganggap lancar dan 3 lainnya menganggap sangat lancar.



Gambar 5.20 Statistika tingkat kesulitan game

Mayoritas responden menganggap tingkat kesulitan game sudah pas sesuai dengan Gambar 5.20 di mana 5 responden menganggap game tingkat kesulitan sudah pas, 2 responden merasa game mudah, dan 1 responden merasa game terlalu mudah.



Gambar 5.21 Statistika penangkapan pengetahuan sejarah



Gambar 5.22 Statistika pengetahuan sejarah sebelum bermain game

Pada sisi pemahaman sejarah, responden memiliki tanggapan yang berbeda-beda. Berdasarkan Gambar 5.21, 1 responden mengaku mengalami kesulitan dalam memahami sejarah yang disampaikan, 3 responden ragu-ragu, 3 lagi merasa mudah menanggapi pesan sejarah, dan 1 responden menganggap pesan sejarah sangat mudah untuk ditangkap. Untuk pemahaman kejadian sejarah itu sendiri, berdasarkan Gambar 5.22, hanya 1 responden yang mengaku sudah pernah mendengar kejadian *Attack of the Dead Men*. Sisanya belum pernah mendengar kejadian tersebut.



Gambar 5.23 Statistika error dalam game

Pada saat bermain, 2 responden mengaku terjadi error saat bermain sementara 6 responden tidak menemukan error. Hal ini sesuai dengan statistika pada Gambar 5.23.



Gambar 5.24 Statistika efek rasa pusing saat menggunakan VR

Menurut Gambar 5.24, 1 responden mengaku merasa pusing ketika memakai alat VR. Sisanya tidak menemukan kendala tersebut saat bermain dengan VR.

