

## PERANCANGAN GAME PLATFORMER “GRAPPLE” PADA PC

Gery Trenalady<sup>1</sup>, Darius Andana Haris<sup>2</sup>, Jeanny Pragantha<sup>3</sup> & Carlene Lim<sup>4</sup>

<sup>1</sup>Program Studi Teknik Informatika, Universitas Tarumanagara Jakarta

*Email: gery.535180060@stu.untar.ac.id*

<sup>2</sup>Fakultas Teknologi Informasi, Universitas Tarumanagara Jakarta

*Email: dariush@fti.untar.ac.id*

<sup>3</sup>Fakultas Teknologi Informasi, Universitas Tarumanagara Jakarta

*Email: jeannyp@fti.untar.ac.id*

<sup>4</sup>Program Studi Teknik Informatika, Universitas Tarumanagara Jakarta

*Email: carlene.825190058@stu.untar.ac.id*

### ABSTRACT

"Grapple" is a platformer game designed using Unity and targeted to be played on PC. Sprites and assets used in this game were created with Pixilart. Players can choose between 2 level packs where each level pack consists of 6 levels. The goal in the game "Grapple" is to pass all available obstacles with the help of a grappling gun and get the highest score possible from each played level. Design & Planning carried out using game design methods. Testing is carried out using black box testing.

**Keywords:** Platformer, grapple, unity

### ABSTRAK

"Grapple" merupakan *game platformer* yang dirancang dengan Unity dan ditargetkan untuk dimainkan pada PC. Sprite dan aset yang digunakan pada game ini dibuat dengan Pixilart. Pemain dapat memilih antara 2 level pack dimana masing-masing level pack terdiri dari 6 level. Level yang dimainkan memiliki rintangannya sendiri dengan tingkat kesulitan yang semakin tinggi dari level sebelumnya. Tujuan dalam game "Grapple" adalah melewati semua rintangan yang tersedia dengan alat bantuan grappling gun dan mendapatkan skor setinggi mungkin dari setiap level yang dimainkan. Untuk tahapan perancangan menggunakan metode *game design*. Untuk pengujian akan dilakukan dengan *black box testing*.

**Kata Kunci:** Platformer, grapple, unity

## 1. PENDAHULUAN

Perkembangan teknologi saat ini semakin pesat, sehingga kebutuhan manusia seperti kemudahan penggunaan fasilitas, kemudahan beban kerja, komunikasi, hiburan, dan lainnya menjadi semakin dinamis. Salah satu kebutuhan manusia yang sangat penting adalah hiburan untuk menghilangkan stres atau penat di tempat kerja. *Game* masih menjadi hiburan yang populer hingga saat ini. *Game* dapat didefinisikan sebagai aktivitas atau kompetisi secara fisik atau mental dengan aturan dan dilakukan orang untuk kesenangan mereka sendiri (Ar-Raqib & Roche, 2010). *Game* berkembang dengan cepat dan memiliki tren yang dinamis seiring perkembangan teknologi. *Game* sangat diminati oleh kalangan anak-anak, remaja, orang dewasa, dan orang tua. *Game* juga memiliki banyak jenis seperti RPG, *action*, *adventure*, *arcade*, FPS, dan masih banyak lagi (Vince, 2022). Salah satu jenis *game* yang mengandung unsur ini adalah *game platformer*. *Platformer* merupakan sebuah *subgenre* dari *game* bergenre *action* yang menampilkan grafik dua atau tiga dimensi dimana pemain mengendalikan karakter yang melompat atau memanjat ke *platform* yang berbeda (iD Tech, 2022).

Di dalam *game platformer*, pemain bergerak baik secara vertikal maupun horizontal melalui lingkungan yang dibuat, melompat dari satu *platform* ke *platform* lain, menghindari rintangan, dan melawan musuh. Pemain memiliki kemampuan untuk melompat secara supernatural dan umumnya tidak dapat terluka jika jatuh jarak jauh,

kecuali jika pemain jatuh ke sesuatu yang berbahaya atau ke jurang yang tak berdasar. Kedua hal tersebut adalah fitur yang umum ditemukan pada *game platformer*. Tindakan yang dapat dilakukan pemain terdiri dari berlari dan melompat, dilengkapi dengan gerakan seperti berputar, memanjat, serta meluncur di dinding disertai fitur lingkungan seperti angin. *Game platformer* menggunakan fisika yang tidak realistis, seperti pemain dapat mengubah arah saat melompat di udara (Adams, 2014). Judul *game* yang dirancang adalah “Grapple”. *Game* ini dirancang dengan *Unity* dan dapat dimainkan pada PC. Perancangan *game* menggunakan *Unity* karena mendukung pembuatan *game* untuk PC. *Game* yang dirancang mendukung *platform* PC karena kontrol permainan lebih leluasa dan mudah digunakan pada PC. Pembuatan *game* serupa pernah dilakukan pada tahun 2021 pada penelitiannya, Pembuatan *Game Platformer* “Interval” dengan Fitur *auto jumping* (Subetha et al., 2021). Pada *game* interval, pemain harus menunggu selang waktu tertentu untuk dapat melompat secara otomatis. Berbeda dengan *game platformer* pada umumnya, di Grapple, pemain akan mengendalikan sebuah karakter yang terbangun pada sebuah ruangan dan menemukan alat *grappling gun* saat mengeksplorasi tempat karakter pemain berada. Oleh *grappling gun*, pemain dapat mengeksplor ke tempat yang biasanya tidak akan dapat ditelusuri tanpa alat tersebut. *Game* ini memiliki tampilan dua dimensi dan bersifat *single-player*.

## 2. METODE PENELITIAN

Penelitian dimulai dari tahap perancangan. Dasar teori diperlukan dalam merancang sebuah *game* sebagai pondasinya. Beberapa hal yang perlu dijabarkan dan dibahas lebih lanjut adalah tahapan perancangan, proses pembuatan, dan *genre game*.

Metode perancangan *game design* diperlukan sebagai titik referensi untuk pembuatan *game* nantinya. Metode perancangan juga menentukan ruang lingkup *game* yang akan dibuat. Beberapa hal yang perlu diperhatikan menurut Subetha et al (2021), yaitu: (a) *high concept* memberikan gambaran singkat tentang *game* yang akan dirancang dan mengarahkan pengembangan *game* agar tidak menyimpang dari rancangan yang sudah ditentukan; (b) *gameplay* menggambarkan apa yang dapat dilakukan dalam *game* dan bagaimana cara melakukannya. Poin-poin penting yang perlu diperhatikan yaitu, (a) alat atau cara untuk mengontrol karakter yang terdapat di dalam *game* dijelaskan pada desain kontrol; (b) hal-hal yang berkaitan dengan karakter utama maupun karakter musuh dijelaskan pada desain karakter; (c) objek, fungsi, dan hal-hal lainnya berkaitan dengan objek tersebut dijelaskan pada desain objek; (d) level yang akan diselesaikan pemain dijelaskan pada desain level; (e) suara yang digunakan untuk mengisi latar maupun efek dijelaskan pada desain suara; (f) perhitungan *score* yang didapatkan pemain saat menyelesaikan sebuah level dijelaskan pada desain *score*; (g) *story*, alur cerita dari *game* yang dirancang akan dituliskan pada bagian *story*; (h) *audience*, target pemain dari *game* yang dirancang, baik itu jenis kelamin atau usia; (i) perangkat keras dan perangkat lunak.

Pada bagian ini dijelaskan tentang spesifikasi minimum dari perangkat keras dan perangkat lunak yang dibutuhkan untuk memainkan *game* yang dibuat rancangan tampilan. Gambaran kasar dari tampilan *user interface* (UI) dari gambar yang dirancang dijelaskan pada rancangan tampilan pembuatan *game*. Setelah semua konsep yang dijabarkan pada rancangan, berikutnya akan dibuat menjadi *game* dengan mengumpulkan aset serta *scripting* atau pembuatan *testing*. Setelah *game* selesai dibuat, maka perlu dilakukan pengujian pada *game* tersebut. Pengujian dilakukan untuk menguji apakah *game* terdapat *error* atau *bug* dan apakah *game* yang dirancang sesuai dengan rancangan.

Tahap *testing* dapat dibagi menjadi: (a) *Alpha testing*; dan (b) *Beta testing*. *Alpha testing* merupakan bentuk pengujian *game* yang dilakukan oleh pihak pengembang ataupun ahli yang sudah mengerti betul dalam pembuatan *game*. Tujuan dilakukannya pengujian ini adalah meningkatkan kualitas *game* dan mempersiapkan *game* untuk *beta testing*. *Alpha testing* sering dianggap sebagai *internal acceptance testing* sebelum memasuki tahap *beta testing*. Setelah *alpha testing* dilakukan, berikutnya akan dilakukan *beta testing*. Berbeda dengan tahap *alpha testing*, *beta testing* adalah pengujian yang dilakukan oleh pemain di luar pihak pengembang untuk mendapatkan laporan *error* (*bug report*), masukan, atau saran untuk pengembang *game* yang tidak terpikirkan oleh pembuat *game* sebelumnya. *Beta testing* akan dibuka untuk umum supaya pihak pengembang bisa mendapatkan masukan sebanyak mungkin.

*Genre* atau jenis *game* digunakan untuk mengelompokkan *game* berdasarkan dari interaksi dan *gameplay*. *Game* “Grapple” termasuk dalam *genre platformer*. Tujuan dari *game* ini adalah melewati semua rintangan yang dihadapi dengan menggunakan alat bantuan *grappling gun* secepat mungkin untuk mendapatkan skor setinggi mungkin.

Alur aplikasi. *Game* “Grapple” mempunyai *genre platformer* dengan tampilan 2 dimensi. Tujuan dari *game* ini adalah melewati seluruh rintangan yang dihadapi dengan menggunakan alat bantuan *grappling gun* secepat mungkin untuk mendapatkan skor setinggi mungkin. Setiap *level* memiliki jumlah ruangan bervariasi dengan tingkat kesulitan yang akan meningkat dari satu ruangan ke ruangan berikutnya. Pemain dapat menyimpan *progress* yang sudah dicapai pada setiap *level*, sehingga pemain tidak perlu menyelesaikan semua ruangan pada satu *level* dalam satu sesi.

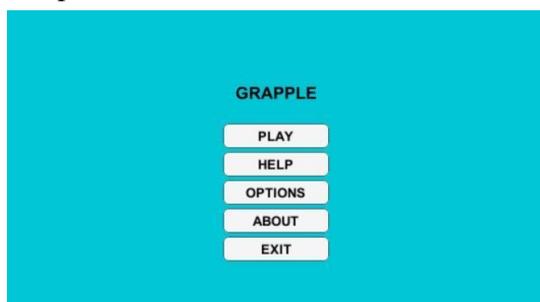
### 3. HASIL DAN PEMBAHASAN

*Game* yang telah selesai melalui tahap pembuatan akan memasuki tahap pengujian. Pengujian dilakukan untuk memastikan bahwa *game* berjalan sesuai dengan rancangannya. Tahap pengujian *game* terdiri dari *black box testing*.

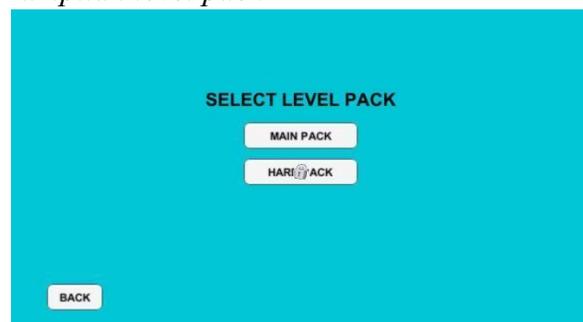
*Black box testing* dilakukan untuk memeriksa modul yang terdapat pada *game* ini. Berikut adalah modul yang diuji:

(a) Pengujian modul menu utama

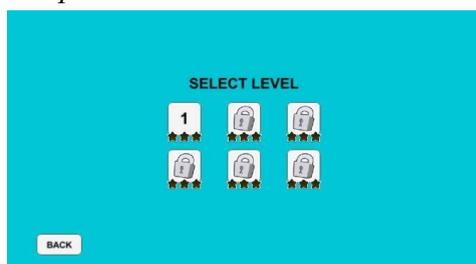
Modul ini merupakan modul awal permainan. Modul ini terdapat 5 tombol, yaitu *play*, *help*, *options*, *about*, dan *exit*. Tombol *play* berfungsi untuk membawa pemain ke modul *level pack*. Tombol *Help* berfungsi untuk menampilkan *window help* yang berisikan cara dasar memainkan *game* ini. Tombol *options* berfungsi untuk menampilkan *window options* yang digunakan untuk mengatur ukuran layar dan suara *game*. Tombol *about* berfungsi untuk menampilkan *window about* yang memiliki informasi dan dosen pembimbing pembuat *game*. Tombol *exit* berfungsi untuk menutup *game*. Tampilan modul ini dapat dilihat pada gambar 1.

**Gambar 1***Tampilan menu utama***(b) Pengujian modul *level pack***

Pada *window level pack*, pemain dapat memilih antara 2 *level pack* yang tersedia, yaitu *main pack* dan *hard pack*. *Hard pack* tidak dapat diakses oleh pemain sampai pemain menyelesaikan level 6 dan mengumpulkan minimal 15 bintang di *main pack*. Tombol *main pack* berfungsi untuk menampilkan *window main level select*. Tombol *hard pack* berfungsi untuk menampilkan *window hard level select*. Pengujian *window level pack* dapat dilihat pada gambar 2.

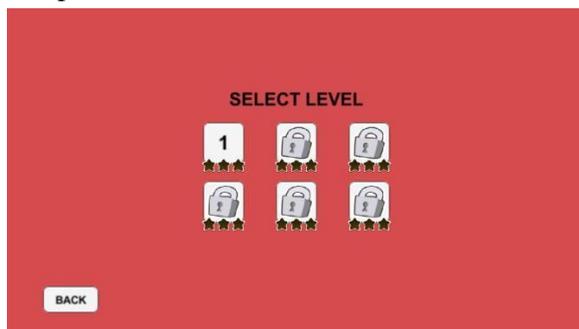
**Gambar 2***Tampilan level pack***(c) Pengujian modul *level select***

Modul *level select* terdiri dari 2 jenis, yaitu *main level select* dan *hard level select*. Masing-masing modul *level select* memiliki 6 tombol level dan tombol *back* di bawah kiri. Pada saat awal memainkan *game* ini, hanya tombol level 1 yang dapat ditekan dan sisa tombol level tidak dapat ditekan sampai level sebelumnya diselesaikan pemain. Menekan salah satu tombol level yang tersedia akan menampilkan *window level information*. Tombol *back* berfungsi untuk kembali ke *window level pack*. Pengujian modul *Level Select* dapat dilihat pada gambar 3 dan gambar 4.

**Gambar 3***Tampilan modul main level select*

## Gambar 4

*Tampilan modul hard level select*



(d) *Tampilan window level information*

Tampilan *level information* terdiri dari tiga bagian, yaitu: (a) *Level information*; (b) *Saved progress information*; dan (c) *Confirm play*. Pada *level information*, *window* yang ditunjukkan berupa informasi level yang akan dimainkan. Informasi yang ditunjukkan terdiri dari level yang akan dimainkan, jumlah ruangan yang harus dilalui, 3 bintang dengan skor yang harus dicapai untuk pada level tersebut untuk mendapatkan bintang tersebut, *highscore* yang merupakan skor tertinggi yang telah dicapai oleh pemain saat menyelesaikan level tersebut, tombol *play*, dan tombol *back*. Tombol *play* berfungsi untuk memainkan level tersebut, sedangkan tombol *back* berfungsi untuk menutup *window level information*. Tampilan *window level information* dapat dilihat pada gambar 5.

## Gambar 5

*Tampilan window level information*



Selanjutnya adalah *saved progress information*. *Window* ini akan muncul di sebelah kanan *window level information* jika pemain sebelumnya telah memainkan level tersebut dan belum menyelesaikannya. *Window* ini berisikan tulisan bahwa pemain ada membuat *progress* pada level ini sebelumnya, waktu yang telah dilalui, ruangan dimana pemain keluar dari level tersebut, jumlah kematian pemain pada level tersebut, dan tombol *continue*. Tombol *continue* berfungsi untuk melanjutkan memainkan level tersebut sesuai dengan informasi yang telah disimpan untuk level tersebut. Tampilan *window saved progress information* dapat dilihat pada gambar 6.

**Gambar 6***Tampilan window saved progress information*

Adapun tahap *confirm play*. *Window* ini akan muncul jika *window saved progress information* muncul dan pada *window level information*, pemain menekan tombol *play*. Pada saat pemain melakukan hal tersebut, *window confirm play* akan muncul untuk mengkonfirmasi pemain apakah pemain yakin ingin mengulang level tersebut dari awal, meskipun pemain ada membuat *progress* sebelumnya. *Window* ini terdiri dari judul menyatakan untuk mengulang level dari awal, catatan menyatakan bahwa *progress* yang sudah dicapai oleh pemain sebelumnya akan dihapus jika pemain memutuskan untuk melakukannya, tombol *play*, dan tombol *back*. Jika pemain menekan tombol *play*, maka pemain akan memainkan level tersebut dari awal dan informasi yang telah disimpan sebelumnya akan dihapus. Jika pemain menekan tombol *back*, maka *window* ini akan ditutup dan memunculkan *window level information* dan *saved progress information* kembali. Tampilan *window confirm play* dapat dilihat pada gambar 7.

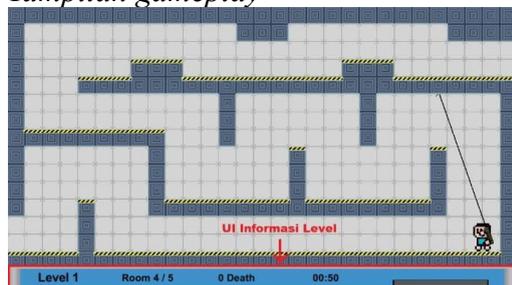
**Gambar 7***Tampilan window confirm play*(e) Tampilan *gameplay*

Tampilan *gameplay* dari *game* “Grapple” terdiri dari *gameplay* dari level yang sedang dimainkan, UI informasi level, dan UI dialog. Pemain dapat menggerakkan karakter dengan tombol W, A, dan D atau tombol panah pada *keyboard* yang terhubung ke PC. Pada saat mendapatkan *grappling gun*, pemain dapat membidik arah tembakan berupa *hook* dengan *mouse* dan menggunakan klik kiri dan klik kanan untuk mengontrol tembakannya. Saat *hook* belum ditembakkan, klik kiri digunakan untuk menembakkan *hook* ke arah sesuai dengan posisi *mouse*. Saat *hook* sudah berhenti bergerak, klik kiri digunakan untuk menarik pemain ke arah *hook*, sedangkan klik kanan digunakan untuk mengembalikan *hook* ke *grappling gun*. Jika pada saat menembak *hook* mendarat pada *metal box*, maka klik kiri akan menarik *metal box* ke arah pemain. Tampilan *gameplay* dapat dilihat pada gambar 8. Berikut merupakan penjelasan dari UI yang terdapat pada tampilan *gameplay*: (1) UI Informasi Level: Terdiri dari level yang dimainkan sekarang,

*progress* ruangan, jumlah kematian pemain, waktu yang telah diluangkan pada level tersebut, dan *progress bar*. *Progress bar* berfungsi untuk menampilkan *timer* untuk 2 *state*, yaitu *reloading* dan *respawning*. *State reloading* ditunjukkan saat pemain menembakkan *hook* dan tali yang terhubung antara *hook* dan *grappling gun* terputus. Pada saat hal ini terjadi, *progress bar* akan mulai terisi secara perlahan-lahan sampai penuh dalam waktu 2 detik dengan teks “Reloading...” di atasnya. Setelah *progress bar* terisi penuh, maka *grappling gun* dapat digunakan lagi seperti biasa. *State respawning* ditunjukkan saat karakter pemain mati. Pada saat hal ini terjadi, *progress bar* akan mulai terisi secara perlahan-lahan sampai penuh dalam waktu 3 detik dengan teks “Respawning...” di atasnya. Setelah *progress bar* terisi penuh, pemain akan mengulang ruangan tersebut dari awal. Tampilan UI informasi level dapat dilihat pada gambar 8.

### Gambar 8

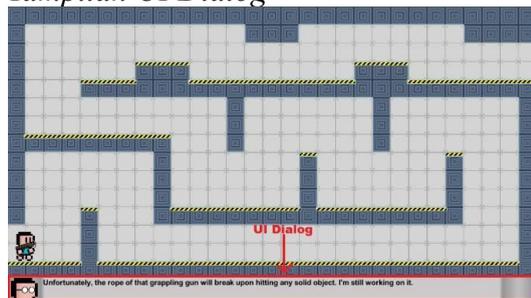
#### Tampilan gameplay



Di dalam UI Dialog, terdiri dari gambar karakter yang mengucapkan kalimatnya dan kalimat yang diucapkan. Pada saat pemain pertama kali masuk ke dalam ruangan dalam sebuah level, UI dialog akan muncul untuk sementara. UI ini berisikan dialog yang diucapkan oleh karakter lain. Setelah selesai mengucapkan dialognya, UI ini akan menghilang 5 detik kemudian. Tampilan UI dialog dapat dilihat pada gambar 9.

### Gambar 9

#### Tampilan UI Dialog



#### (f) Tampilan modul *pause*

Modul ini dapat diakses dengan menekan tombol Esc pada *keyboard* saat dalam tampilan *gameplay*. Tampilan *pause* terdiri dari tiga bagian. Bagian pertama adalah *pause*, merupakan *window* yang pertama muncul saat pemain mengakses *window pause*. Terdapat 3 tombol di bawah tulisan “*paused*”, yaitu tombol *resume*, *restart level*, dan *exit level*. Tombol *resume* berfungsi untuk melanjutkan memainkan *level* yang sedang dimainkan. Tombol *restart level* berfungsi untuk menampilkan *window confirm restart*. Tombol *exit level* berfungsi untuk menampilkan *window confirm exit*. Tampilan *window pause* dapat dilihat pada gambar 10.

## Gambar 10

### Tampilan *window pause*



Bagian kedua adalah *confirm restart*, merupakan *window* yang ditampilkan jika pemain menekan tombol *restart level* pada *window pause*. *Window* ini mengkonfirmasi pemain jika ingin mengulang level yang dimainkan dari awal dan menghapus *progress* yang sudah dicapai pemain pada saat itu. *Window* ini terdiri dari teks konfirmasi untuk mengulang level yang dimainkan dari awal, catatan berisikan *progress* yang sudah dibuat pada level tersebut akan dihapus, tombol *restart*, dan tombol *cancel*. Jika pemain menekan tombol *Restart*, maka pemain akan mengulang level yang sedang dimainkan dari awal dan *progress* yang sudah dibuat sebelumnya akan dihapus. Jika pemain menekan tombol *cancel*, maka pemain akan kembali ke *window Pause*. Tampilan *window confirm restart* dapat dilihat pada gambar 11.

## Gambar 11

### Tampilan *window confirm restart*



Bagian ketiga adalah *confirm exit*, merupakan *window* yang ditampilkan jika pemain menekan tombol *exit level* pada *window pause*. *Window* ini mengkonfirmasi pemain jika ingin keluar dari level yang dimainkan dan memungkinkan menyimpan *progress* yang sudah dicapai pemain pada saat itu berdasarkan kondisi dari level yang dimainkan. *Window* ini terdiri dari teks konfirmasi untuk keluar dari level yang dimainkan, catatan berisikan apa yang akan terjadi pada *progress* yang telah dibuat oleh pemain pada level tersebut, tombol *exit*, dan tombol *cancel*. Jika pemain menekan tombol *exit*, maka pemain akan keluar dari level yang sedang dimainkan dan *progress* yang sudah dibuat sebelumnya akan disimpan atau tidak berdasarkan kondisi dari level yang dimainkan. Terdapat 3 kemungkinan *progress* yang sudah dibuat akan disimpan atau tidak. Kemungkinan pertama adalah pemain masih berada di ruangan pertama. Kemungkinan ini membuat *progress* yang dibuat tidak akan disimpan. Kemungkinan kedua adalah pemain berada di ruangan kedua atau lebih dan belum melakukan pergerakan pada ruangan tersebut. Kemungkinan ini membuat *progress* yang dibuat akan disimpan.

Kemungkinan ketiga adalah pemain berada di ruangan kedua atau lebih dan sudah melakukan pergerakan pada ruangan tersebut. Kemungkinan ini membuat *progress* yang dibuat akan disimpan, tetapi pemain akan mendapatkan penalti berupa penambahan jumlah *death* sebanyak satu untuk level tersebut. Jika pemain menekan tombol *cancel*, maka pemain akan kembali ke *window pause*. Tampilan *window confirm exit* beserta dengan semua kemungkinan *progress* akan disimpan atau tidak dapat dilihat pada gambar 12 sampai gambar 14.

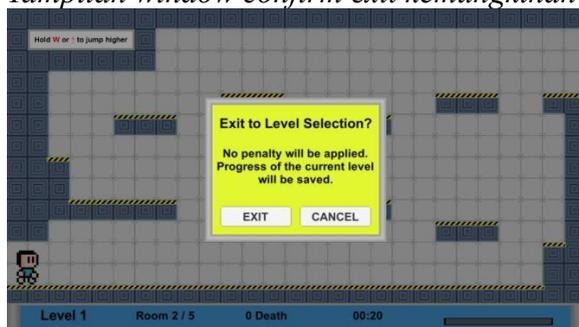
### Gambar 12

*Tampilan window confirm exit kemungkinan pertama*



### Gambar 13

*Tampilan window confirm exit kemungkinan kedua*



### Gambar 14

*Tampilan Window Confirm Exit Kemungkinan Ketiga*



## 4. KESIMPULAN DAN SARAN

Setelah selesai melakukan pengujian pada game “Grapple” dengan metode *blackbox testing*, maka dapat ditarik kesimpulan yaitu: (a) game sudah selesai dan dapat berjalan dengan baik; (b) fitur yang direncanakan sudah diimplementasi dan berfungsi dengan baik; (c) berikutnya game akan memasuki tahap *alpha & beta testing* untuk mengetahui *feedback* dari *user* secara langsung.

**REFERENSI**

- Ar-Raqib, A., & Roche E. M. (2010). *Virtual Worlds Real Terrorism*. Barraclough Ltd.
- Adams, E. (2014). *Fundamental of Action and Arcade Game Design*. Pearson Education.
- iD Tech. (2022, November 18). *10 Types of Platforms in Platform Video Games*. iD Tech. <https://www.idtech.com/blog/10-types-of-platforms-in-platform-video-games>
- Schell, J. (2019). *The Art of Game Design: A Book of Lenses, Third Edition*. CRC Press.
- Subetha, Z., Pragantha, J., Haris, D. A. (2021). *Pembuatan Game Platformer “Interval” dengan Fitur Auto Jumping*. *Jurnal Ilmu Komputer dan Sistem Informatika*, 9(1), 237-242.
- Vince. (2022, November 18). *The Many Different Types of Video Games & Their Subgenres*. iD Tech. <https://www.idtech.com/blog/different-%20types-of-video-game-genres>