

Penerapan *Multiplayer* Pada Gim Tower Defense Menggunakan Photon Unity

Shena Star Sarwodi¹, Wibisono Sukmo Wardhono², Muhammad Aminul Akbar³

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya
Email: ¹senanomorsatu@gmail.com, ²wibiwardhono@ub.ac.id, ³muhammad.aminul@ub.ac.id

Abstrak

Gim *multiplayer online* adalah gim yang diharuskan untuk memiliki koneksi internet, yang mempunyai kelebihan dapat dimainkan dimanapun dengan jarak yang jauh sekalipun. *Tower defense* adalah subgenre dari genre strategi yang bertujuan untuk mempertahankan wilayah atau harta benda pemain dengan menghalangi penyerang musuh, biasanya dicapai dengan menempatkan struktur pertahanan di sepanjang jalur serangan mereka. *Framework* yang dapat digunakan untuk pengembangan gim *multiplayer* adalah Photon Unity. Photon Unity adalah kerangka pengembangan permainan *multiplayer* real-time yang memiliki layanan server dan internet. Tujuan penelitian ini adalah untuk meningkatkan interaktivitas pada gim bergenre *tower defense* dengan menerapkan sistem *multiplayer* menggunakan photon unity. Berdasarkan pengujian *Game Experience Questioner* (GEQ) yang telah dilakukan untuk mengetahui dampak penerapan fitur *multiplayer* ini, didapatkan bahwa pemain dengan jenis kelamin perempuan lebih memiliki perasaan empati dan negatif terhadap lawan bermain. Sedangkan pemain dengan jenis kelamin laki-laki cenderung melibatkan perilaku terhadap lawan mainnya. Pemain yang belum pernah memainkan gim bergenre *real-time strategy* memiliki rasa empati lebih tinggi terhadap lawan bermain. Sedangkan pemain yang pernah memainkan gim bergenre *real-time strategy* memiliki perasaan negatif dan keterlibatan perilaku yang lebih terhadap lawan mainnya.

Kata kunci: *Gim, Android, Multiplayer, Tower Defense, Photon Unity Networking*

Abstract

Online multiplayer game is a game that necessarily need an internet connection and ease to be played anywhere remotely, known as an advantage of this game. Tower defense is a subgenre of strategy which purposely used to defend player's territory or property by blocking enemies as an attacker through defense structures by placing along their roadway of attack. The framework that can be used for multiplayer game development is Photon Unity. Photon Unity is a real-time multiplayer framework that has server and internet services. The objective of this study is to increase interactivity in tower defense genre games by implementing a multiplayer system using photon unity. Based on the Game Experience Questioner (GEQ) researched which used to determine the impact of implementation of this multiplayer feature, it was found that female players had more empathy and negative feelings towards their opponents. Meanwhile, male players have higher behavioral involvement. Players who have never played a real-time strategy genre have a higher sense of empathy to their opponent. Whereas, players who have played a real time strategy genre game have negative feelings and has higher behavioral involvement towards their opponents.

Keywords: *Gim, Android, Multiplayer, Tower Defense, Photon Unity Networking*

1. PENDAHULUAN

Game mobile adalah sebuah gim yang didesain dan dimainkan pada perangkat bergerak, seperti PDA (*Personal Digital*

Assistant), smartphone, tablet PC (*Personal Computer*), dan *portable media player*. Keuntungan tersendiri memainkan *game mobile* adalah portabilitas, yaitu pemain dapat memainkan gim dimana saja selama mereka

mempunyai perangkat yang mampu menjalankan *mobile* gim (Rahadian, Agustri, & Suhandi, 2015). Salah satu platform *mobile* yang banyak digunakan dalam video gim adalah android. Android merupakan sistem operasi yang sudah dikembangkan sejak 2008 dan digemari oleh semua kalangan mulai dari anak-anak hingga dewasa karena fiturnya yang mudah dan efisien (Prabowo & Fatta, 2016). Saat ini gim yang cukup populer dimainkan oleh kebanyakan kalangan adalah gim dengan jumlah pemain yang lebih dari satu untuk memainkannya atau bisa disebut juga dengan gim *multiplayer*.

Sistem *multiplayer* pada sebuah gim membuat gim tersebut menjadi lebih interaktif dan menarik untuk dimainkan. Dalam sebuah gim jika pemain memilih untuk *singleplayer* maka pemain tersebut akan berhadapan dengan lawan NPC (*Non Playable Character*) sedangkan jika *multiplayer* maka pemain tersebut akan berhadapan dengan pemain lain. Gim *multiplayer* juga terbagi menjadi *multiplayer offline* dan *multiplayer online*. Pada *game multiplayer offline* pemain dapat berinteraksi dengan pemain lain tanpa harus menggunakan koneksi internet, contohnya game "Two Guys & Zombies" dengan memanfaatkan jaringan Bluetooth dan game "Ludo King" yang dimainkan berdua dalam satu perangkat yang sama (Ismail, 2019). Sedangkan game *multiplayer online* diharuskan untuk memiliki koneksi internet, kelebihannya dari *multiplayer offline* adalah *multiplayer online* dapat dimainkan dimanapun dengan jarak yang jauh sekalipun.

Selain jumlah pemain dan platform yang digunakan gim dapat dibedakan menurut genre untuk mengelompokkan gim tersebut. Dalam setiap gim terdapat genre yang bermacam-macam, salah satunya adalah *strategy*. Gim *strategy* adalah genre dalam sebuah gim yang berfokus pada pemikiran dan perencanaan yang terampil untuk mencapai sebuah kemenangan (Rollings & Adams, 2003). Dalam genre *strategy* terdapat sub genre yang cukup populer yaitu *Tower defense*, *tower defense* sendiri adalah subgenre dari video game strategi yang bertujuan untuk mempertahankan wilayah atau harta benda pemain dengan menghalangi penyerang musuh, biasanya dicapai dengan menempatkan struktur pertahanan di sepanjang jalur serangan mereka (Damon Reece, 2015). *Tower defense* juga dapat dikategorikan sebagai subgenre dari genre *real-time strategy*, karena

asal-usulnya yang *real-time*, meskipun banyak game bergenre *tower defense* modern memasukkannya ke dalam genre *turn-based strategy*.

Untuk mengembangkan sebuah gim bergenre tower defense, developer gim dapat mempelajari cara mengembangkannya melewati berbagai macam media elektronik seperti buku elektronik, video tutorial dan website pembelajaran. Salah satu website pembelajaran yang ada di Indonesia adalah dicoding, dicoding merupakan salah satu contoh media pembelajaran elektronik yang bertujuan untuk mengembangkan ekosistem developer di Indonesia. Gim tower defense dapat dipelajari melalui kelas "Menjadi Game Developer Expert" pada halaman website Dicoding. Pada kelas tersebut juga terdapat materi tentang bagaimana mengembangkan gim *multiplayer*.

Untuk materi gim *multiplayer* dicoding menggunakan UNet (Unity networking) dalam pembelajarannya, sedangkan pada laman website support unity (Glover, 2020) mengatakan bahwa *service* UNet tidak digunakan lagi untuk masa yang akan datang. Maka dari itu dibutuhkan *framework* lain untuk mengembangkan gim *multiplayer*. Salah satu *framework* yang dapat digunakan untuk pengembangan gim *multiplayer* adalah Photon Unity. Photon Unity adalah kerangka pengembangan permainan *multiplayer* real-time yang memiliki layanan server dan internet (paladinstudios, 2015).

2. LANDASAN PUSTAKA

2.1. Genre Tower Defense

Tower defense adalah subgenre dari genre gim strategi yang bertujuan untuk mempertahankan wilayah atau harta benda pemain dengan menghalangi penyerang musuh, biasanya dicapai dengan menempatkan struktur pertahanan di sepanjang jalur serangan mereka (Damon Reece, 2015). *Tower defense* juga dapat dikategorikan sebagai subgenre dari genre *real-time strategy*, karena asal-usulnya yang *real-time*, meskipun banyak game bergenre *tower defense* modern memasukkannya ke dalam genre *turn-based strategy*.

Tower defense memiliki gameplay dimana pemain akan menghalangi musuh yang akan menyerang benteng si pemain, pemain akan menghalangi musuh menggunakan bidak untuk bertahan dan menggunakan bidang untuk

menyerang benteng lawan (Teguh Pribadi, Cahyawan Wiranatha, & Buana, 2017). Contoh gim bergenre tower defense yang populer adalah Clash Royale yang dapat dilihat pada Gambar 1



Gambar 1. Gameplay Clash Royale

2.2. Multiplayer

Multiplayer adalah fitur dari sebuah gim yang memungkinkan pemain bisa saling berinteraksi dengan pemain lainnya. Fitur *multiplayer* membuat sebuah gim menjadi lebih interaktif dan menarik untuk dimainkan. Gim *multiplayer* juga terbagi menjadi *multiplayer offline* dan *multiplayer online*. Pada gim *multiplayer offline* pemain dapat berinteraksi dengan pemain lain tanpa harus menggunakan koneksi internet, contohnya game “Two Guys & Zombies” dengan memanfaatkan jaringan Bluetooth yang dapat dilihat pada Gambar 2 dan game “Ludo King” yang dimainkan berdua dalam satu perangkat yang sama dapat dilihat pada Gambar 3 (Ismail, 2019). Sedangkan gim *multiplayer online* diharuskan untuk memiliki koneksi internet, kelebihanannya dari *multiplayer offline* adalah *multiplayer online* dapat dimainkan dimanapun dengan jarak yang jauh sekalipun.



Gambar 2. Gameplay Two Guys & Zombies



Gambar 3. Fitur Ludo King

2.3. Unity

Unity merupakan salah satu *game engine* paling populer saat ini. Unity adalah sebuah *software development* yang terintegrasi untuk menciptakan *video game* atau konten lainnya seperti visualisasi arsitektur atau *real-time animasi* baik yang bernuansa 2D maupun 3D. Unity juga merupakan perangkat lunak yang digunakan untuk mengembangkan *game multiplatform* yang didesain secara *user friendly* (Iman, 2017). Keunggulan Unity adalah Unity dapat dengan mudah mengontrol objek-objek dalam gim atau aplikasi. Unity terdapat 2 jenis lisensi yaitu *personal edition* yang dapat diakses secara gratis dan *professional edition* yang diharuskan untuk membayar perbulan untuk mengaksesnya dengan beberapa fitur tambahan yang tidak terdapat di *personal edition* (Bimantara & Asri, 2020).

2.4. Photon Unity Networking

Photon Unity Networking (PUN) adalah sebuah *framework* untuk pengembangan gim *multiplayer* secara *online* pada Unity. PUN tersedia dalam dua versi yaitu gratis dan berbayar, keunggulan PUN versi berbayar yaitu terdapat fitur tambahan yang tidak terdapat pada versi yang gratisnya. Terdapat dua konsep utama dalam PUN yaitu *lobby* dan *room*. Menyambungkan ke *lobby* merupakan proses untuk menyambungkan perangkat pada server photon. Sedangkan menyambungkan ke *room* merupakan proses untuk menyambungkan ke *room* yang tersedia untuk berinteraksi dengan pemain lain. Dalam proses *multiplayer online*, PUN menyinkronkan data dari satu klien ke klien lainnya. Komunikasi yang digunakan untuk mekanisme sinkronisasi dapat melalui sebagai berikut: *Object Synchronization*, *Remote*

Procedure Call (RPC), dan *Custom Properties* (D. Polančec and I. Mekterović, 2017).

Object Synchronization digunakan untuk menyinkronkan posisi, rotasi dan nilai lainnya dengan duplikat jarak jauh menggunakan komponen PhotonView. PhotonView harus diatur untuk mengamati komponen seperti Transform pada *script*. Untuk menyinkronkan suatu objek, *script* yang digunakan harus mengimplementasikan

`OnPhotonSerializeView()` agar dapat diamati komponen PhotonView. Pada `OnPhotonSerializeView()`, posisi dan nilai-nilai lain ditulis ke Stream dan membacanya. Untuk menggunakan fungsi ini *script* harus mengimplementasikan *interface* `IPunObservable` (Photon).

Remote Procedure Call (RPC) digunakan ketika membutuhkan method yang harus dipanggil oleh semua klien yang ada dalam satu *room*. Dengan menggunakan atribut `[PunRPC]` pada suatu method, ketika method itu akan dipanggil maka seluruh klien yang ada dalam satu *room* akan memanggil method yang sama pada *script* masing-masing. Cara untuk memanggil method dengan atribut RPC adalah dengan menggunakan `photonView.RPC("Nama_method", RpcTarget.All)`, panggilan ini akan menargetkan PhotonView tertentu pada `GameObject` (Photon).

Custom Properties terdiri dari nilai-nilai kunci Hashtable yang dapat diisi sesuai permintaan. Nilai-nilai disinkronkan dan *cache* pada klien, sehingga klien tidak harus mengambilnya sebelum digunakan. Perubahan akan terjadi pada klien lain dengan menggunakan `SetCustomProperties()`. *Custom Properties* digunakan ketika *room* dan pemain memiliki beberapa atribut yang tidak terkait dengan `GameObject` (Photon).

2.4. Frame Per Second

Frame per second (FPS) adalah seberapa banyak frame yang bisa dihasilkan oleh sebuah perangkat pada sebuah gim dalam hitungan per detik. Dalam sebuah gim, semakin tinggi jumlah FPS yang dihasilkan maka gerakan yang terasa pada gim tersebut akan semakin halus dan lancar. Tinggi rendahnya sebuah FPS dipengaruhi oleh kemampuan atau kapasitas dari perangkat yang digunakan. Pada perangkat *smartphone* rata-rata memiliki layar dengan ukuran FPS 60Hz, yang berarti bahwa perangkat

tersebut sanggup menjalankan gim dengan FPS hingga 60Hz. Dalam sebuah gim jika terjadi FPS *drop* maka tampilan gim akan menjadi patah-patah seperti terjadi gangguan koneksi, namun FPS sama sekali tidak ada hubungannya dengan permasalahan koneksi (Subari, n.d.).

2.5. Pengujian Delay

Delay atau latency adalah waktu yang dibutuhkan untuk sebuah paket yang dikirimkan dari sumber awal menuju tujuan akhir pada sebuah jaringan. Dalam sebuah proses transmisi paket pada jaringan komputer, delay disebabkan karena adanya antrian yang panjang. Delay diperoleh dari selisih waktu kirim antara satu paket TCP dengan paket lainnya dalam satuan seconds. Untuk menghitung delay pada paket yang ditransmisikan adalah dengan membagi antara panjang paket (dalam satuan bit) dibagi dengan link bandwidth (dalam satuan bit/s) (Fahmi, 2018).

Untuk mengontrol jaringan dan simulasi lag dari client, Photon menyediakan sebuah komponen GUI sederhana yaitu Photon Lag Simulation Gui. Dalam komponen ini terdapat lag dan loss yang bias ditambahkan untuk mencoba seberapa baik mekanisme gim dalam kondisi jaringan yang lebih buruk (Photon, Photon Lag Simulation Gui, 2020). Kontrol yang terdapat pada komponen simulasi ini dapat dilihat pada Tabel 1

Tabel 1. Kontrol pada komponen Photon Lag Simulation GUI

Kontrol	Keterangan
RTT	Waktu pulang pergi hingga pesan diterima oleh server
Sim	Mengaktifkan dan menonaktifkan simulasi
Lag	Menambahkan penundaan tetap untuk semua pesan keluar dan masuk (satuan milidetik)
Jit	Menambahkan penundaan acak hingga n milidetik per pesan
Loss	Menjatuhkan persentase pesan yang ditetapkan

2.6. Pengujian Game Experience Questioner Game Experience Questioner (GEQ)

adalah alat ukur dalam bentuk kuesioner untuk mengukur bagaimana pengalaman pengguna dalam bermain gim. GEQ memiliki tiga struktur modular yaitu *The core questionnaire*, *The Social Presence Module* dan *The Post-game module*. Ketiga modul tersebut dimaksudkan untuk dikelola segera setelah sesi permainan selesai. Bagian *The core questionnaire* dan *The Social Presence Module* menyelidiki perasaan dan pikiran para pemain saat bermain gim, sedangkan *The Post-game module* menilai perasaan pemain setelah mereka berhenti bermain (Poels, IJsselsteijn, & Kort, 2013). Pada pengujian ini peneliti menggunakan modul *Social Presence* dikarenakan pada sistem *multiplayer* terdapat keterlibatan pemain dengan pemain lainnya.

The Social Presence Module merupakan modul pada GEQ yang menyelidiki keterlibatan psikologis dan perilaku pemain dengan entitas sosial lainnya, baik virtual (karakter dalam gim), dimediasi (pemain lain saat bermain *online*), atau ditempatkan bersama (Poels, IJsselsteijn, & Kort, 2013). Pada modul ini terdapat 17 pertanyaan yang dapat dilihat pada Gambar 4.

4. GEQ - Social Presence Module

Please indicate how you felt while playing the game for each of the items, on the following scale:

not at all	slightly	moderately	fairly	extremely
0	1	2	3	4
< >	< >	< >	< >	< >

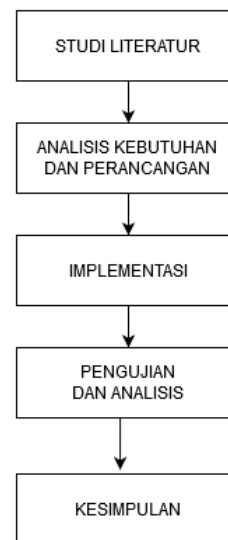
- 1 I empathized with the other(s)
- 2 My actions depended on the other(s) actions
- 3 The other's actions were dependent on my actions
- 4 I felt connected to the other(s)
- 5 The other(s) paid close attention to me
- 6 I paid close attention to the other(s)
- 7 I felt jealous about the other(s)
- 8 I found it enjoyable to be with the other(s)
- 9 When I was happy, the other(s) was(were) happy
- 10 When the other(s) was(were) happy, I was happy
- 11 I influenced the mood of the other(s)
- 12 I was influenced by the other(s) moods
- 13 I admired the other(s)
- 14 What the other(s) did affected what I did
- 15 What I did affected what the other(s) did
- 16 I felt revengeful
- 17 I felt schadenfreude (malicious delight)

Gambar 4. Pertanyaan Modul *Social Presence* pada GEQ

3. METODOLOGI

Pada bagian ini akan dijelaskan tahap-tahap yang akan digunakan dalam melakukan penelitian, termasuk diantaranya studi literatur

serta proses pengembangan game itu sendiri. Alur diagram dari metodologi penelitian dapat dilihat pada Gambar 5.



Gambar 5. Diagram Alur Metodologi Penelitian

4. HASIL DAN PEMBAHASAN

4.1. Analisis Kebutuhan

4.1.1. Kebutuhan Fungsional

Kebutuhan fungsional adalah komponen-komponen atau fungsi yang harus ada dalam sebuah system. Untuk mempermudah proses identifikasi kebutuhan maka diberi suatu identitas dalam penulisannya. Pada kebutuhan suatu system hal ini diperlukan untuk menjaga kekonsistensiannya. Kebutuhan fungsional dapat dilihat pada Tabel 2.

Tabel 2. Kebutuhan fungsional sistem *multiplayer*

No	Kode Kebutuhan Sistem	Deskripsi Kebutuhan
1	PM-001	Sistem harus dapat menghubungkan gim ke server photon
2	PM-002	Sistem harus dapat menyediakan mekanisme untuk membuat room baru
3	PM-003	Sistem harus dapat menyediakan mekanisme untuk bergabung room yang sudah ada
4	PM-004	Sistem harus dapat melihat list room yang ada
5	PM-005	Sistem harus dapat membuat objek pasukan
6	PM-006	Sistem harus dapat menyinkronkan posisi objek pasukan
7	PM-007	Sistem harus dapat

menyinkronkan state objek pasukan

4.1.2. Kebutuhan Non-Fungsional

Kebutuhan non-fungsional adalah kebutuhan yang tidak berada pada kebutuhan fungsionalitas. Kebutuhan non-fungsionalitas sistem ini dapat dilihat pada Tabel 3.

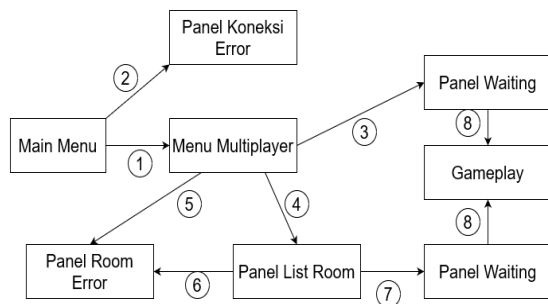
Tabel 3. Kebutuhan Non-fungsional sistem

Parameter Kebutuhan	Deskripsi Kebutuhan
Expeirience	Pemain dapat menambah pengalaman bermain dengan sistem <i>multiplayer</i>
FPS	Tidak terdapat perbedaan FPS antara <i>singleplayer</i> dan <i>multiplayer</i>

4.2. Perancangan

4.2.1. Perancangan Antarmuka

Untuk antarmuka yang dirancang adalah antarmuka yang akan ditampilkan pada pemain untuk sistem *multiplayer*. Terdapat enam antarmuka yaitu antarmuka menu, antarmuka menu *multiplayer*, antarmuka panel list, antarmuka waiting, antarmuka koneksi error dan antarmuka room error. Untuk alur kerja antarmuka dapat dilihat pada Gambar 6.

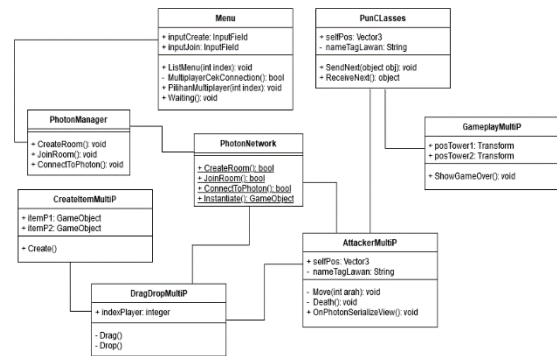


Gambar 6. Alur Kerja Antarmuka

4.2.2. Perancangan Class Diagram

Berikut merupakan kelas diagram pada sistem multiplayer yang akan diterapkan pada gim. Class PhotonNetwork dan PunClasses merupakan class yang ada pada library photon. Class PhotonManager merupakan class yang menampung fungsi untuk menyambungkan ke server, membuat room dan bergabung pada room yang ada. Class Menu berfungsi sebagai antarmuka pada main menu, semua aksi yang ada pada main menu ditampung pada class Menu. Class CreateItemMultiP dan DragDropMultiP merupakan class yang

berfungsi untuk membuat pasukan pada gameplay multiplayer. Class AttackMultiP merupakan class yang mengatur pergerakan pada pasukan. Class GameplayMultiP merupakan class yang mengatur gameplay multipler untuk mengetahui menang atau kalahnya diakhir gim. Class diagram dapat dilihat pada Gambar 7.



Gambar 7. Class Diagram sistem Multiplayer

4.2.3. Perancangan Algoritme

1. ConnectToServer

Menyambungkan ke server

Pass In: tidak ada

IF tersambung internet

Sambungkan ke server

ELSE

Menampilkan pemberitahuan dibutuhkan sambungan internet

ENDIF

Pass Out: tidak ada

2. CreateRoom

Membuat Room Baru

FUNCTION membuat room baru

Pass In: nama room

Membuat room dengan nama room sesuai inputan

Menampilkan tampilan menunggu pemain lain

Pass Out: tidak ada

3. JoinRoom

Bergabung pada Room yang ada

FUNCTION bergabung pada room yang ada

Pass In: nama room

Bergabung pada room dengan nama room sesuai inputan

IF berhasil bergabung room

Menampilkan gameplay gim

ELSE

Menampilkan pemberitahuan tidak terdapat room

ENDIF

Pass Out: tidak ada

4. ListRoom

Menampilkan Daftar Room

FUNCTION menampilkan daftar room yang tersedia

Pass In: List room

FOR sebanyak room yang telah ada

```
Hapus room
ENDFOR
FOR sebanyak room yang terdapat pada List room
  Menginisialisasi room sesuai List
ENDFOR
Pass Out: tidak ada
```

5. CreateItem

```
Membuat Pasukan
FUNCTION membuat pasukan pemain
Pass In: Transform posisi, area
  IF area valid
    Buat pasukan
  ELSE
    Pasukan tidak dibuat
  ENDIF
Pass Out: tidak ada
```

6. Menampilkan pasukan

```
Menampilkan Pasukan
FUNCTION menampilkan pasukan lawan
Pass In: pemilik pasukan
  IF pemilik pasukan adalah saya
    Posisi pasukan sama dengan posisi sekarang
    ditambah arah lawan
    Kirim posisi sekarang ke server
  ELSE
    Posisi pasukan sama dengan posisi yang ada pada
    server
  ENDIF
Pass Out: tidak ada
```

7. Menghancurkan pasukan

```
Menghancurkan Pasukan
FUNCTION menghancurkan pasukan
Pass In: tidak ada
  IF defense <= 0
    IF pemilik pasukan adalah saya
      Hancurkan objek pasukan dari server
    ENDIF
  ENDIF
Pass Out: tidak ada
```

8. Kondisi menang atau kalah

```
Menampilkan Kondisi Menang atau kalah
FUNCTION menampilkan kondisi menang atau kalah
Pass In: tidak ada
  IF Tower hancur
    Gambar kalah
  ELSE
    Gambar menang
  ENDIF
  Hitung score dan tampilkan panel score
Pass Out: tidak ada
```

4.2.4. Perancangan Karakter

Perancangan karakter merupakan rancangan terhadap karakter dan komponen apa saja yang dibutuhkan untuk penerapan *multiplayer*. Dalam gim ini terdapat tiga karakter yaitu kastil, pasukan menyerang dan pasukan bertahan.

1. Kastil

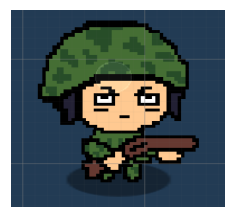
Kastil merupakan objek yang sangat penting, dikarenakan hancurnya kastil dapat menentukan menang atau kalahnya dalam gim ini. Terdapat dua kastil yaitu kastil player1 (bagian kiri) dan kastil player2 (bagian kanan) yang dapat dilihat pada Gambar 8. Elemen yang terdapat pada karakter ini adalah defense sebesar 5000.



Gambar 8. Kastil Player1 dan Player2

2. Tentara

Karakter ini berfungsi untuk menyerang kastil lawan atau menyerang pasukan lawan yang akan menyerang kastil pemain. Pada karakter ini terdapat data yang perlu disinkronkan yaitu posisi dan kondisi dari karakter. Gambar dari karakter tentara dapat dilihat pada Gambar 9. Elemen yang terdapat pada karakter ini adalah attack bernilai 100, defense bernilai 200 dan cost bernilai 300.



Gambar 9. Karakter Tentara

3. Barikade

Karakter ini berfungsi untuk menahan pasukan musuh untuk mencegah musuh menyerang kastil pemain. Tampilan dari karakter barikade dapat dilihat pada Gambar 10. Elemen yang terdapat pada karakter ini adalah defense bernilai 500 dan cost bernilai 100.



Gambar 10. Karakter Barikade

4.2.5. Perancangan Level

Perancangan level merupakan rancangan terhadap *rule* dari *gameplay multiplayer* pada gim ini. *Rule* dari *gameplay multiplayer* dapat dilihat pada Tabel 4.

Tabel 4. *Rule Gameplay Multiplayer*

No	Rule
1	Pemain menunggu jumlah koin terisi hingga bisa membeli pasukan
2	Pemain membeli pasukan menyerang atau bertahan
3	Pemain membuat pasukan untuk menghancurkan kastil lawan
4	Pemain akan mendapatkan score ketika mengalahkan pasukan lawan
5	Pemain akan kalah jika kastil berhasil dihancurkan pemain lain
6	Pemain akan menang jika berhasil menghancurkan kastil pemain lain

4.2.6. Sinkronisasi Data dan State

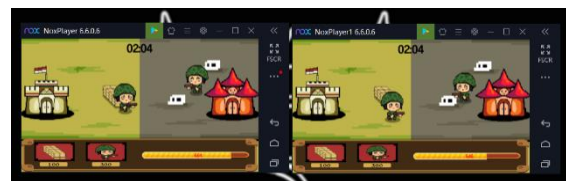
Sinkronisasi data dan state merupakan rancangan terhadap data dan state apa saja yang perlu disinkronkan untuk penerapan sistem *multiplayer* pada gim. Data dan state yang perlu disinkronkan terdiri dari tiga kategori yaitu: *Frequent Updates*, *Infrequent Updates*, dan *Rare Updates And State* yang dapat dilihat pada Tabel 5.

Tabel 5. Sinkronisasi Data dan State

No	Kategori	Data atau State
1	<i>Frequent Updates</i>	Posisi dan kondisi karakter tentara
2	<i>Infrequent Updates</i>	Ketika pemain membuat objek pasukan
3	<i>Rare Updates And State</i>	Kondisi menang atau kalah dan kondisi putus koneksi

4.3. Implementasi

Tahap ini akan menerapkan hasil rancangan sebelumnya menggunakan gim engine Unity3D, untuk rancangan antarmuka akan langsung diterapkan pada tampilan menu di gim tersebut dan untuk kode program akan diimplementasikan menggunakan Bahasa pemrograman C# menggunakan Visual Studio 2017. Berikut merupakan tampilan *Gameplay* dari sistem *multiplayer* yang telah diterapkan. Tampilan *Gameplay multiplayer* dapat dilihat pada Gambar 11.



Gambar 11. Tampilan *Gameplay Multiplayer*

4.4. Pengujian

Pada penelitian ini dilakukan 5 pengujian yaitu pengujian *blackbox*, *whitebox*, *fps*, *delay* dan *Game Experience Questioner (GEQ)*.

4.4.1. Pengujian Blackbox

Berdasarkan hasil dari pengujian black box yang telah dilakukan terhadap sistem yang diterapkan pada gim, didapatkan hasil yang sesuai dengan apa yang diharapkan, antara lain:

1. Sistem dapat menghubungkan gim ke server photon
2. Sistem dapat menyediakan mekanisme untuk membuar room baru
3. Sistem dapat menyediakan mekanisme untuk bergabung room yang sudah ada
4. Sistem dapat melihat list room yang ada
5. Sistem dapat membuat objek pasukan
6. Sistem dapat menyinkronkan posisi objek pasukan
7. Sistem dapat menyinkronkan state objek pasukan

4.4.2. Pengujian Whitebox

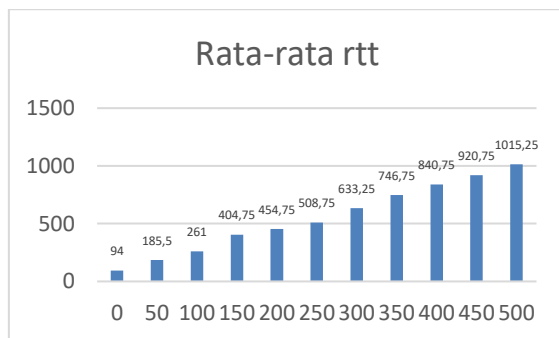
Dari pengujian yang telah dilakukan tidak ditemukan terjadinya error pada kode program. Dari hasil pengujian white box ini menunjukkan bahwa aplikasi sudah sesuai dengan harapan dari algoritme yang telah dirancang.

4.4.3. Pengujian FPS

Berikut merupakan hasil analisis dari pengujian FPS pada penerapan sistem multiplayer pada game tower defense, pengujian FPS membandingkan antara mode singleplayer dan mode multiplayer. Saat dimainkan FPS pada singleplayer berkisar antara 58 sampai dengan 60 FPS dan FPS pada multiplayer berkisar antara 58 sampai dengan 60 FPS. Dari hal tersebut dapat disimpulkan bahwa tidak terdapat perbedaan FPS pada saat singleplayer dan multiplayer.

4.4.4. Pengujian Delay

Dari pengujian delay menggunakan lag simulasi dari photon didapatkan ukuran paket yang dikirimkan ke server sebesar 70 bytes dan menerima dari server sebesar 67 bytes untuk satu karakter. Untuk mengukur lama perjalanan pulang pergi ke server dapat dilihat dari RTTnya. Berikut pada Gambar 12 menunjukkan rata-rata RTT dari Lag simulasi dengan rentan dari 0 sampai 500. Pada saat nilai lag 150 terdapat delay saat pasukan musuh telah dikalahkan dan gerakan pasukan pemain terlihat patah-patah saat berjalan. Semakin tinggi nilai lag maka delay pasukan musuh menghilang akan semakin lama dan pergerakan musuh akan lebih patah-patah lagi.



Gambar 12. Tampilan *Gameplay Multiplayer*

4.4.5. Pengujian GEQ

Pengujian geq dilakukan dengan cara meminta 10 orang untuk memainkan gim tower defense yang telah menerapkan multiplayer. Pemain diminta untuk memainkan gim pada menu pvp bersama dengan pemain lainnya. Setelah selesai memainkan gim, pemain diminta untuk mengisi kuesioner yang berisi pertanyaan dari modul Social Presence.

Hasil analisis dari pengujian GEQ menggunakan penilaian yang ada pada modul Social Presence. Pada hasil analisis ini kan

diambil penilaian berdasarkan kategori umur, jenis kelamin dan genre yang pernah dimainkan. Penilaian pada modul Social Presence terdiri dari tiga aspek yaitu Empathy, Negative Feelings, dan Behavioural Involvement. Skor komponen dihitung sebagai nilai rata-rata itemnya.

1. Jenis kelamin

Pada kategori jenis kelamin terdapat 6 responden dengan jenis kelamin laki-laki dan 4 responden dengan jenis kelamin perempuan. Tabel 6 menunjukkan bahwa pemain dengan jenis kelamin perempuan lebih memiliki perasaan empati dan negatif terhadap lawan bermain. Sedangkan pemain dengan jenis laki-laki cenderung menerlibatkan perilaku terhadap lawan mainnya.

Tabel 6. Hasil analisis GEQ berdasarkan jenis kelamin

Aspek	Laki-laki	Perempuan
<i>Empathy</i>	2.69	2.79
<i>Negative Feelings</i>	2.23	2.45
<i>Behavioural Involvement</i>	2.97	2.67

2. Genre yang pernah dimainkan

Kategori genre yang pernah dimainkan adalah responden yang pernah memainkan genre *Real-time strategy* dan akan dibandingkan dengan yang belum pernah memainkan genre tersebut. Tabel 7 menunjukkan bahwa pemain yang belum pernah memainkan gim bergenre real-time strategy memiliki rasa empati lebih tinggi terhadap lawan bermain. Sedangkan pemain yang pernah memainkan gim bergenre real-time strategy memiliki perasaan negatif dan keterlibatan perilaku yang lebih terhadap lawan mainnya.

Tabel 7. Hasil analisis GEQ berdasarkan genre

Aspek	Pernah	Belum pernah
<i>Empathy</i>	2.33	3.13
<i>Negative Feelings</i>	2.36	2.28
<i>Behavioural Involvement</i>	2.90	2.80

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Penelitian penerapan multiplayer pada gim tower defense menggunakan Photon Unity berhasil diterapkan berdasarkan studi literature, analisis kebutuhan, perancangan, implementasi dan pengujian yang telah dibahas pada bab sebelumnya. Maka dapat diambil kesimpulan sebagai berikut:

1. Dengan menerapkan Photon Unity Networking pada gim tower defense maka dapat diimplementasikan sebuah fitur yang dapat meningkatkan interaktivitas dan ketertarikan pemain pada gim yaitu fitur multiplayer. Berdasarkan hasil dari pengujian Blackbox, fitur multiplayer yang dibuat sudah sesuai dengan kebutuhan fungsional yang telah dibuat dan berdasarkan hasil dari pengujian Whitebox, tidak terdapat logical error pada kode program yang telah diimplementasikan.
2. Berdasarkan pengujian *Frame Per Second* (FPS) yang telah dilakukan, didapatkan bahwa saat dimainkan FPS pada *singleplayer* berkisar antara 58 sampai dengan 60 FPS dan FPS pada *multiplayer* berkisar antara 58 sampai dengan 60 FPS. Dengan demikian dapat disimpulkan bahwa tidak terdapat perbedaan FPS pada saat *singleplayer* dan *multiplayer*.
3. Berdasarkan pengujian *Game Experience Questioner* (GEQ) yang telah dilakukan untuk mengetahui dampak penerapan fitur *multiplayer* pada gim tower defense terhadap pengalaman bermain, didapatkan bahwa pemain dengan jenis kelamin perempuan lebih memiliki perasaan empati dan negatif terhadap lawan bermain. Sedangkan pemain dengan jenis laki-laki cenderung melibatkan perilaku terhadap lawan mainnya. Pemain yang belum pernah memainkan gim bergenre *real-time strategy* memiliki rasa empati lebih tinggi terhadap lawan bermain. Sedangkan pemain yang pernah memainkan gim bergenre *real-time strategy* memiliki perasaan negatif dan keterlibatan perilaku yang lebih terhadap lawan mainnya.

5.2. Saran

Berdasarkan penelitian yang telah dilakukan, maka saran untuk penelitian selanjutnya adalah sebagai berikut:

1. Penerapan *multiplayer* dapat menggunakan *framework* lain selain photon unity networking.
2. Menerapkan *multiplayer* pada genre gim yang lain selain genre tower defense.

6. DAFTAR PUSTAKA

- Bimantara, T., & Asri, S. D. (2020). Pembuatan Game Puzzle Online Dengan Prinsip Kooperatif Berbasis Android.
- D. Polančec and I. Mekterović. (2017). Developing MOBA games using the Unity game engine. *2017 40th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, 1510-1515.
- Damon Reece. (2015, April 27). *Best Tower Defense Games of All Time*. Retrieved from [gameranx.com](https://gameranx.com/features/id/13529/article/best-tower-defense-games/): <https://gameranx.com/features/id/13529/article/best-tower-defense-games/>
- Fahmi, H. (2018). Analisis Qos (Quality Of Service) Pengukuran Delay, Jitter, Packet Lost Dan Throughput Untuk Mendapatkan Kualitas Kerja Radio Streaming Yang Baik.
- Glover, D. (2020, February 27). *UNet Deprecation FAQ*. Retrieved from [unity3D.com](https://support.unity3d.com/hc/en-us/articles/360001252086-UNet-Deprecation-FAQ?_ga=2.159492216.432905744.1583688530-1676824110.1580575272): https://support.unity3d.com/hc/en-us/articles/360001252086-UNet-Deprecation-FAQ?_ga=2.159492216.432905744.1583688530-1676824110.1580575272
- Iman, A. N. (2017). 3D Labirin Adventure Game Untuk Pengenalan Pelajaran PKPBA Menggunakan Algoritma Menge Sebagai Crowd Simulation Pada NPC.
- Ismail, N. (2019, November 5). *"20 Game Multiplayer Offline Android Terbaik*

- di 2019 / Tanpa Internet Seru Buat *Mabar!*". Retrieved from jalantikus: <https://jalantikus.com/tips/game-multiplayer-offline/>
- paladinstudios. (2015, April 1). *Photon Unity Network, How to Create an Online Multiplayer Game with Photon Unity*. Retrieved from paladinstudios: <http://www.paladinstudios.com>
- Photon. (2020). *Photon Lag Simulation Gui*. Retrieved from photonengine: <https://doc.photonengine.com/en-us/pun/current/troubleshooting/phot-on-lag-simulation-gui>
- Photon. (n.d.). *Synchronization and State*. Retrieved from Photonengine: <https://doc.photonengine.com/en-us/pun/v2/gameplay/synchronization-and-state#main>
- Poels, K., IJsselsteijn, W., & Kort, Y. d. (2013). The game experience questionnaire. *Eindhoven: Technische Universiteit Eindhoven*, 3-9.
- Prabowo, A. D., & Fatta, H. A. (2016). "Perancangan Dan Pembuatan Game Arcade 'Phandawa : Werkudara Sang Penyelamat'".
- Rahadian, M., Agustri, S., & Suhandi, N. (2015). "Pembangunan Game Ayo Tarik Berbasis Android".
- Rollings, A., & Adams, E. (2003). Andrew Rollings and Ernest Adams on Game Design.
- Subari, H. R. (n.d.). *Frame Per Second (FPS): Fitur di Dalam Game yang Sering Bikin Salah Kaprah!* Retrieved from Dunia Games: <https://duniagames.co.id/discover/article/frame-per-second-fps-fitur-di-dalam-game-yang-sering-bikin-salah-kaprah>
- Teguh Pribadi, I. G., Cahyawan Wiranatha, A. A., & Buana, P. W. (2017). Rancang Bangun Game Tower Defense 'Defense of Dewata Island' Berbasis Android.