

Rancang Bangun *Game Side Scrolling War Online Multiplayer 2D*

Dio Wandana¹, Husaini^{2*}, Indrawati³

^{1,2,3} Jurusan Teknologi Informasi dan Komputer Politeknik Negeri Lhokseumawe
Jln. B.Aceh Medan Km.280 Buketrata 24301 INDONESIA

¹diowandana16@gmail.com

² husaini@pnl.ac.id (penulis korespondensi)

³indrawati@pnl.ac.id

Abstrak— Dalam beberapa tahun terakhir, *game* menjadi salah satu hiburan yang paling diminati salah satunya *game side scrolling 2D*, namun, pada umumnya *game side scrolling 2D* hanya dapat dimainkan secara *offline*. *Game "Side-scrolling war"* merupakan salah satu *game online multiplayer 2D* yang dibangun sebagai salah satu media hiburan. Penelitian ini bertujuan untuk merancang dan membangun *game "Side-scrolling war"* dengan menggunakan jaringan internet sebagai media penghubung agar pemain dapat bermain secara *online multiplayer* dan berinteraksi secara *real-time*, dengan menggunakan jaringan internet pemain dapat mengakses *game* dimanapun dan kapanpun selama perangkat mereka terhubung dengan internet, jaringan internet juga memungkinkan *game* untuk menawarkan mode *multiplayer* yang luas, termasuk permainan kooperatif dan kompetitif. *Game* ini menggunakan *Photon Unity Networking* yang merupakan sebuah *toolkit* atau *plugin* untuk *unity* yang dikembangkan oleh *photon*. Berdasarkan hasil dari pengujian ping menunjukkan respon yang cukup baik dari *photon* dengan menunjukkan nilai rata-rata sebesar 41 ms. Berdasarkan pengujian kelayakan fungsional *game* menggunakan teknik pengujian *blackbox* diperoleh nilai persentase dari hasil pengujian tersebut sebesar 100%, yang didapat dari 6 pengujian yaitu pengujian tombol *credit*, tombol info, tombol buat *room*, tombol *join room*, tombol keluar *room* dan tombol keluar *game*.

Kata kunci— *game, multiplayer, side scrolling, photon unity networking*

Abstract— In recent years, games have become one of the most popular forms of entertainment, one of which is 2D side scrolling games, however, in general 2D side scrolling games can only be played offline. The game "Side-scrolling war" is a 2D multiplayer online game built as an entertainment medium. This research aims to design and build a "Side-scrolling war" game using the internet network as a connecting medium so that players can play online multiplayer and interact in real-time, by using the internet network players can access the game anywhere and anytime as long as their devices are connected with the internet, internet networks also allow games to offer extensive multiplayer modes, including cooperative and competitive play. This game uses Photon Unity Networking which is a toolkit or plugin for unity developed by Photon. Based on the results of the ping test, it shows a fairly good response from photon by showing an average value of 41 ms. Based on testing the functional feasibility of the game using the black box testing technique, the percentage value obtained from the test results was 100%, which was obtained from 6 tests, namely testing the credit button, info button, create room button, join room button, exit room button and exit game button.

Keyword— *game, multiplayer, side scrolling, photon unity networking*

I. PENDAHULUAN

Dalam beberapa tahun terakhir kemajuan teknologi komputer sangat pesat seiring berjalannya waktu. Permainan atau game pada komputer juga banyak bermunculan seiring dengan perkembangan teknologi komputer tersebut. Game juga menjadi salah satu opsi untuk menghilangkan kejenuhan dan menjadi hiburan untuk menghilangkan rasa penat para pemainnya. Game tidak hanya dapat dimainkan dengan satu device saja, namun game multiplayer yang dapat dimainkan menggunakan dua device yang saling terkoneksi juga dapat dibuat untuk dimainkan bersama. Untuk membuat suatu game multiplayer yang dapat dimainkan oleh lebih dari satu pemain, maka dibutuhkan suatu media penghubung yang berfungsi untuk menghubungkan dua atau lebih komputer yang bermain bersama. media penghubung yang dapat digunakan adalah sebuah jaringan public. Jaringan public adalah jaringan komputer yang terdiri dari perangkat dan infrastruktur yang dapat diakses oleh publik secara umum.

Jaringan public ini dirancang untuk menghubungkan berbagai perangkat, sistem, dan layanan di seluruh dunia agar dapat saling berkomunikasi dan bertukar informasi[1].

Permasalahan pada game side-scrolling 2D ini pada umumnya hanya bisa dimainkan secara offline dengan hanya menggunakan satu device saja dan jumlah pemain yang terbatas. Berdasarkan permasalahan tersebut dibangunnya game side-scrolling war online multiplayer 2D menggunakan sebuah jaringan public sebagai media penghubung. Permainan ini menyediakan satu latar tempat dimana setiap pemain saling berperang dan harus mengalahkan pemain lain[1].

Berdasarkan uraian diatas, maka pada skripsi ini dilakukan penelitian tentang rancang bangun game side-scrolling war online multiplayer 2D menggunakan photon unity networking agar dapat dimainkan menggunakan jaringan public. Nopi Ramsari dan Gilang Ramadhan (2018) melakukan penelitian sejenis yaitu membuat game side-scrolling 2D The Naila's survival. Perbedaan penelitian

skripsi ini dengan penelitian skripsi Nopi Ramsari dan Gilang Ramadhan adalah pada penelitian Nopi Ramsari dan Gilang Ramadhan membuat game side-scrolling 2D The Naila's survival yang hanya dapat dimainkan oleh satu pemain atau single player dan dimainkan tanpa menggunakan jaringan dan terbatas pada satu device, sedangkan pada penelitian skripsi ini akan dibangun game side-scrolling war online multiplayer 2D yang dimainkan dengan jaringan yang menghubungkan lebih dari satu device dan juga multiplayer[2].

A. *Blackbox Testing*

Blackbox testing adalah metode pengujian perangkat lunak yang berfokus pada menguji fungsionalitas eksternal suatu sistem tanpa memperhatikan struktur internal atau kode program yang mendasarinya. Dalam metode ini, pengujian dilakukan dari sudut pandang pengguna akhir atau pengguna luar yang tidak memiliki pengetahuan rinci tentang bagaimana perangkat lunak diimplementasikan. *Blackbox testing* bertujuan untuk mengidentifikasi kesalahan atau cacat dalam perangkat lunak berdasarkan masukan dan keluaran yang diharapkan dari sistem tersebut. Ini mirip dengan menguji produk dengan cara yang sama seperti pengguna sebenarnya akan menggunakannya, tanpa harus mengetahui bagaimana program di dalamnya dirancang atau diimplementasikan[3].

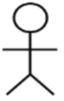
B. UML(Unified Modeling Language)

UML merupakan singkatan dari "Unified Modeling Language" yang dalam bahasa Indonesia dapat diartikan sebagai "Bahasa Pemodelan Bersatu." UML adalah suatu bahasa standar yang digunakan dalam rekayasa perangkat lunak untuk menggambarkan dan merancang berbagai aspek dari sistem perangkat lunak yang akan dikembangkan. UML digunakan untuk membantu pengembang perangkat lunak dalam menggambarkan konsep, desain, dan struktur sistem secara visual, sehingga memudahkan pemahaman dalam pengembangan perangkat lunak. UML melibatkan penggunaan diagram-diagram yang menggambarkan berbagai aspek sistem[4]. Diagram-diagram UML antara lain adalah:

1. Diagram Use Case

Diagram *use case* merupakan diagram yang menggambarkan interaksi antara aktor (pengguna eksternal) dan sistem. Tabel I menjelaskan simbol-simbol diagram *use case*

TABEL I
SIMBOL-SIMBOL DIAGRAM USE CASE

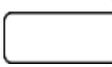
No	Simbol	Nama	Keterangan
1		<i>Actor</i>	<i>Actor</i> mewakili peran atau entitas dari luar sistem yang berinteraksi dengan sistem untuk mencapai tujuan tertentu.
2		<i>Generalization</i>	Mengindikasikan bila actor berinteraksi secara pasif dengan sistem.
3		<i>Include</i>	<i>Include</i> adalah salah satu relasi yang digunakan untuk menggambarkan hubungan antara dua skenario <i>use case</i> yang terkait.
4		<i>Extend</i>	<i>Extend</i> mengindikasikan bahwa suatu aksi atau

			perilaku opsional dapat terjadi dalam konteks dari <i>use case base</i> apabila kondisi tertentu terpenuhi.
5		<i>Association</i>	<i>Association</i> dalam <i>use case</i> diagram menunjukkan bahwa suatu <i>actor</i> terlibat dalam suatu skenario penggunaan tertentu.
6		<i>Use Case</i>	<i>Use Case</i> mewakili fungsi-fungsi atau aktivitas yang dapat dilakukan oleh <i>actor</i> dengan sistem perangkat lunak.

2. Diagram Activity

Diagram *activity* merupakan diagram yang menunjukkan alur kerja atau proses dalam bentuk alur langkah-langkah. Tabel II menjelaskan simbol-simbol diagram *activity*.

TABEL II
SIMBOL-SIMBOL DIAGRAM ACTIVITY

No	Simbol	Nama	Keterangan
1		<i>Start point</i>	<i>start point</i> adalah simbol yang menunjukkan titik awal dari aliran aktivitas atau proses.
2		<i>End point</i>	<i>End point</i> adalah simbol dalam diagram <i>activity</i> yang menggambarkan titik akhir dari aliran aktivitas.
3		<i>Activities</i>	<i>Activities</i> atau aktivitas menggambarkan suatu proses atau tindakan.
4		<i>Decision point</i>	<i>decision point</i> adalah suatu elemen yang digunakan untuk menggambarkan titik dalam alur aktivitas yang mengharuskan sistem membuat keputusan berdasarkan kondisi tertentu.

C. C#

C# (dibaca sebagai "C sharp") adalah bahasa pemrograman yang dikembangkan oleh Microsoft. C# merupakan salah satu bahasa pemrograman yang termasuk dalam keluarga bahasa pemrograman C, dan dirancang untuk digunakan dalam pengembangan aplikasi berbasis .NET Framework. .NET Framework adalah sebuah platform pengembangan perangkat lunak yang dikembangkan oleh Microsoft untuk membangun berbagai jenis aplikasi, termasuk aplikasi desktop, aplikasi web, dan aplikasi seluler. C# dikembangkan oleh Microsoft pada tahun 2000 dan telah menjadi salah satu bahasa pemrograman yang populer di industri perangkat lunak. Bahasa ini sangat kuat, aman, dan mudah dipelajari, dan memiliki dukungan yang luas dari komunitas pengembang[5].

D. Unity

Unity adalah sebuah platform pengembangan perangkat lunak yang digunakan untuk membuat permainan komputer, permainan video, aplikasi berbasis realitas virtual (VR), simulasi, dan berbagai jenis aplikasi interaktif lainnya. Platform ini merupakan sebuah game engine yang kuat dan

populer di kalangan pengembang permainan dan aplikasi. Dengan menggunakan Unity, para pengembang dapat membuat permainan dan aplikasi yang dapat berjalan di berbagai platform, termasuk Windows, macOS, Linux, iOS, Android, PlayStation, Xbox, Nintendo, dan lain-lain. Unity menyediakan berbagai fitur dan alat yang memungkinkan para pengembang untuk menciptakan grafik 2D dan 3D, fisika, animasi, skriping, manajemen asset, dan lain-lain[6].

E. Game

Game adalah bentuk hiburan interaktif yang melibatkan pemain untuk berpartisipasi secara aktif dalam simulasi atau tantangan yang dibuat oleh pembuat *game*. *Game* seringkali memiliki aturan dan tujuan tertentu yang harus dicapai oleh pemain untuk mencapai kemenangan atau keberhasilan dalam permainan. *Game* dapat dimainkan secara tunggal atau multi-pemain dan dapat dijalankan pada berbagai platform, seperti komputer, konsol *game*, atau perangkat seluler. *Game* sendiri memiliki beragam *genre* seperti *action*, *adventure*, *strategy*, *sports*, dan lain-lain[7].

Menurut Dawang Muctar (2005), *game* atau permainan adalah sesuatu yang dapat dimainkan dengan aturan tertentu sehingga ada yang menang dan ada yang kalah, biasanya dalam konteks tidak serius atau dengan tujuan *refreshing*[1].

F. Game 2 Dimensi

Game 2 dimensi adalah game yang berbasis pada dua dimensi atau bidang datar, yang artinya game ini hanya memiliki kedalaman visual dalam game yang terbatas. Game 2D seringkali menggunakan sprite atau karakter 2D yang dipasang pada background atau lingkungan 2 dimensi. Game 2 dimensi ini biasanya memiliki gameplay yang lebih sederhana dan kurang kompleks jika dibandingkan dengan game 3 dimensi[2].

G. PVP Multiplayer Online

PvP *Multiplayer Online* adalah singkatan dari *Player versus Player Multiplayer Online*. Ini mengacu pada jenis permainan atau mode permainan dalam video *game* di mana pemain bermain melawan pemain lainnya dalam lingkungan daring atau *online*. Dalam *PvP Multiplayer Online*, pemain saling berhadapan dan bertanding untuk mencapai tujuan tertentu, seperti mengalahkan tim lawan, mencapai skor tertinggi, menguasai wilayah, atau mencapai prestasi tertentu. Pemain memiliki kesempatan untuk menunjukkan keterampilan dan strategi dalam melawan pemain lain secara langsung. *PvP Multiplayer Online* menawarkan pengalaman yang intens dan kompetitif, di mana pemain dapat berinteraksi dengan pemain lain dari seluruh dunia. Para pemain harus beradaptasi dengan gaya bermain lawan, merencanakan strategi, dan berkoordinasi dengan anggota tim untuk mencapai kemenangan. Hal ini menciptakan lingkungan permainan yang dinamis dan menantang, yang membuat banyak pemain tertarik untuk terus bermain dan memperbaiki keterampilan mereka[8].

H. Photon Cloud

Photon cloud adalah bagian dari layanan yang disediakan oleh *Photon Network*, yang merupakan *platform* pengembangan perangkat lunak untuk permainan *real-time* dan aplikasi multipemain. *Photon cloud* adalah solusi *hosting cloud* yang disediakan oleh *Photon* untuk memfasilitasi permainan multipemain yang responsif, andal, dan mudah

diakses. *Photon cloud* menyediakan infrastruktur dan *server* yang terletak di *cloud* (awan) untuk mengelola koneksi dan komunikasi antara pemain dalam permainan multipemain. Ini berarti bahwa data permainan disinkronisasi secara *real-time* melalui *server cloud*, memastikan pemain di seluruh dunia dapat bermain bersama tanpa masalah *latency* atau keterlambatan yang signifikan[8].

I. PUN (Photon Unity Networking)

PUN adalah sebuah *toolkit* atau *plugin* untuk *unity* yang dikembangkan oleh *photon*. Ini adalah salah satu bagian dari *photon network* yang dirancang khusus untuk memudahkan integrasi dan pengembangan permainan multipemain berbasis *unity*. PUN memungkinkan para pengembang menggunakan *photon cloud* atau *server* pribadi mereka sendiri untuk meng-host permainan multipemain secara *real-time*. Dengan PUN, pengembang dapat dengan mudah menyinkronkan data permainan antara para pemain, mengelola koneksi, mendeteksi benturan antar objek, dan mengimplementasikan fitur-fitur multipemain lainnya dengan cepat dan efisien[8].

J. Side Scrolling

Menurut Ramsari dan Ramadhan (2018), *Side Scrolling* merupakan sudut pandang pemain yang terlihat dari samping dan kemungkinan karakter utama untuk bergerak dari kiri ke kanan serta kemungkinan *background* pada objek dan sifatnya dalam game. Komponen juga disebut pecahan-pecahan fungsional dari setiap *Game Object*[9].

II. METODOLOGI PENELITIAN

A. Metode MDLC(Multimedia Development Life Cycle)

Dalam perancangan dan pengembangannya penelitian ini menggunakan metode MDLC(Multimedia Development Life Cycle). Metode ini terdiri dari enam tahap, yaitu:

1. Concept

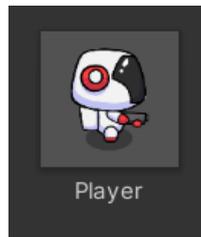
Game side scrolling war merupakan *game* tembak menembak *Player Versus Player* (PVP) yang berarti pemain akan melawan pemain lain dan saling menembak satu sama lain. *Game* ini memiliki aturan bermain dimana pemain harus saling menembak dan membunuh untuk mendapat *point kill* jika salah satu pemain telah mencapai 5 *point kill* maka pemain tersebut dinyatakan sebagai pemenang dan *game* pun berakhir.

2. Design

Perancangan desain *game* ini terdiri dari perancangan map dan karakter.

3. Material Collecting

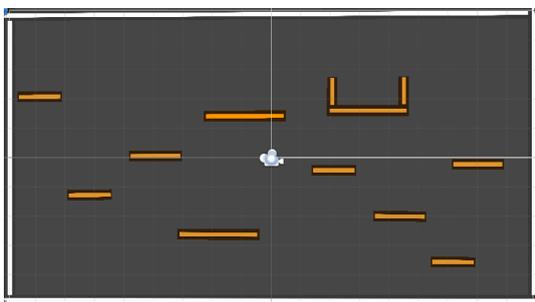
Pada tahap ini peneliti mengumpulkan material yang dibutuhkan, material yang diperoleh untuk membangun *game* yaitu asset, semua asset yang digunakan merupakan asset yang diperoleh dari internet. Berikut beberapa asset yang digunakan dalam pembuatan *game*. Gambar 1, gambar 2, gambar 3 dan gambar 4 menunjukkan *asset* yang digunakan.



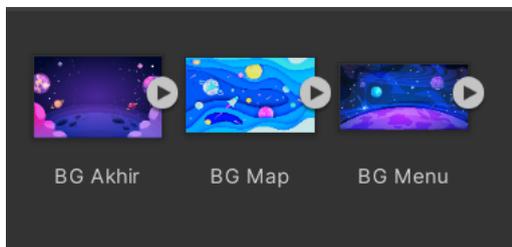
Gambar 1. Character Collecting



Gambar 2. Bullet Collecting



Gambar 3. Block collecting



Gambar 4. Background Collecting

4. Coding

Setelah mengumpulkan semua material yang dibutuhkan, tahap selanjutnya adalah melakukan *coding*, ini dibutuhkan untuk membuat rancangan berfungsi seperti membuat karakter dapat berjalan, melompat serta membuat komponen seperti tombol dapat berfungsi dengan semestinya.

5. Testing

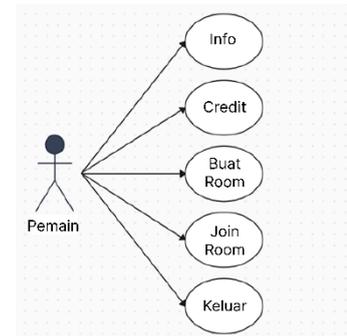
Setelah melakukan tahap *coding*, langkah selanjutnya adalah melakukan tahap *testing*, pada tahap ini melakukan uji coba terhadap game yang telah dibangun dengan cara memainkan game tersebut. Tahap ini dilakukan oleh kerabat atau orang yang ada di sekitar.

6. Distribution

Setelah melakukan tahap *testing* dan *game* telah berjalan seperti yang diharapkan, selanjutnya dilakukan tahap *distribution* tahap ini *game* akan di-*compile* menjadi (exe).

B. Rancangan Use Case Diagram

Diagram *Use Case* adalah salah satu jenis diagram yang digunakan dalam pemodelan perangkat lunak untuk menggambarkan cara pengguna (atau aktor) berinteraksi dengan sistem perangkat lunak. Gambar 5 menunjukkan rancangan *use case* diagram.



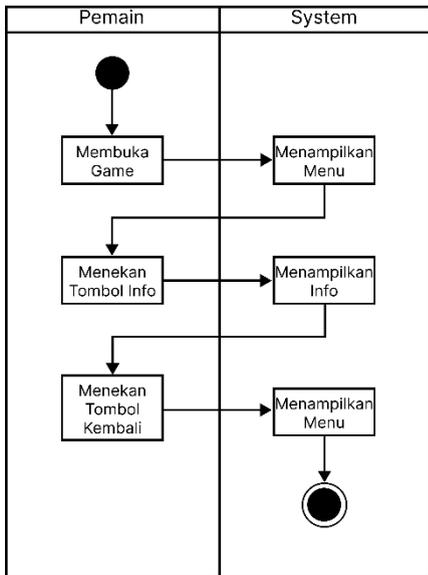
Gambar 5. Rancangan Use Case Diagram

Gambar 5 menjelaskan tentang *use case diagram* dimana terdapat satu aktor dan lima *use case* yaitu *info*, *credit*, *buat room*, *join room* dan *keluar*. Pada *use case info* pemain dapat melihat informasi mengenai cara bermain meliputi tombol-tombol yang digunakan untuk mengontrol karakter dan mekanisme permainan seperti bagaimana cara untuk memenangkan permainan, kemudian pemain dapat kembali ke menu utama ketika menekan tombol kembali. Pada *use case credit* pemain dapat melihat nama dari pembuat *game*, kemudian pemain dapat Kembali ke menu utama ketika menekan tombol kembali. Pada *use case buat room* pemain harus memasukkan nama *room* yang ingin dibuat, ini bertujuan agar nantinya pemain yang ingin *join room* dapat mengetahui *room* yang ingin dituju. Pada *use case join room* pemain juga harus mengisi nama *room* yang ingin mereka tuju agar dapat bergabung dengan pemain lain yang telah membuat *room*. Pada *use case keluar* ini akan membuat pemain keluar dari *game*.

C. Rancangan Activity Diagram

1. Activity Diagram Info

Activity diagram *info* merupakan aktivitas yang akan menampilkan layar *info* ketika pemain menekan tombol *info*. Gambar 6 merupakan *activity* diagram *info*.

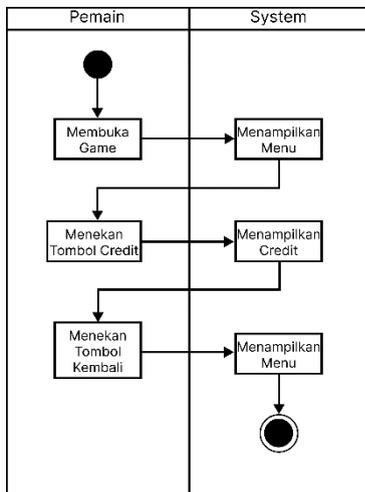


Gambar 6. Activity Diagram Info

Gambar 6 menjelaskan *activity diagram info*, diawali dengan pemain membuka *game*, kemudian *system* akan menampilkan menu utama, selanjutnya pemain menekan tombol info dan *system* pun menampilkan info yang menyangkut cara bermain dan tombol-tombol kontrol, kemudian pemain menekan tombol kembali dan *system* akan kembali menampilkan menu utama.

2. Activity Diagram Credit

Activity diagram credit merupakan aktivitas yang akan menampilkan layar *credit* ketika pemain menekan tombol *credit*. Gambar 7 merupakan *activity diagram credit*.



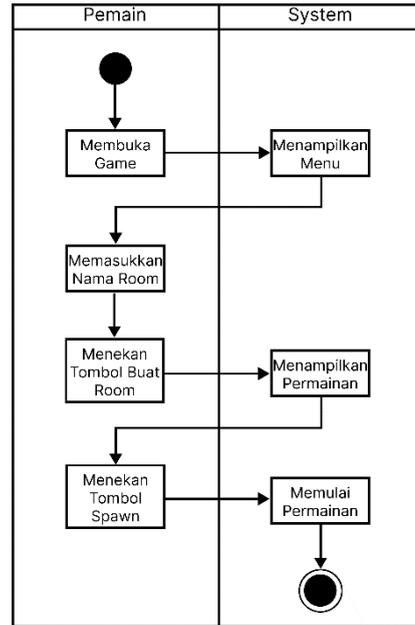
Gambar 7. Activity Diagram Credit

Gambar 7 menjelaskan *activity diagram credit*, diawali dengan pemain membuka *game*, kemudian *system* akan menampilkan menu utama, selanjutnya pemain menekan tombol *credit* dan *system* pun menampilkan *credit* yang berisi nama dari pembuat *game*, kemudian pemain menekan tombol kembali dan *system* akan kembali menampilkan menu utama.

3. Activity Diagram Buat Room

Activity diagram buat room merupakan aktivitas yang akan menampilkan permainan ketika pemain menekan

tombol buat *room* dan berhasil membuat *room*. Gambar 8 merupakan *activity diagram buat room*.

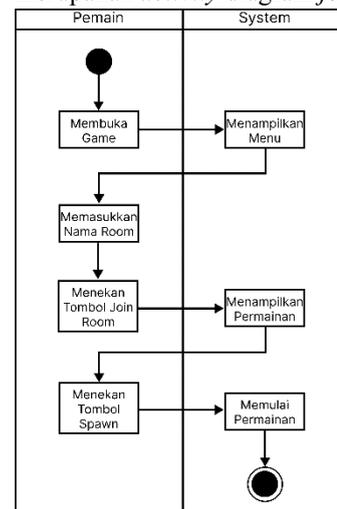


Gambar 8. Activity Diagram Buat Room

Gambar 8 menjelaskan *activity diagram buat room*, diawali dengan pemain membuka *game*, kemudian *system* akan menampilkan menu utama, selanjutnya pemain memasukkan nama *room* dan menekan tombol buat *room* yang selanjutnya *system* akan menampilkan *map* permainan, kemudian pemain menekan tombol *spawn* maka *system* akan memulai permainan.

4. Activity Diagram Join Room

Activity diagram join room merupakan aktivitas yang akan menampilkan permainan ketika pemain menekan tombol *join room* dan berhasil bergabung ke dalam *room*. Gambar 9 merupakan *activity diagram join room*.

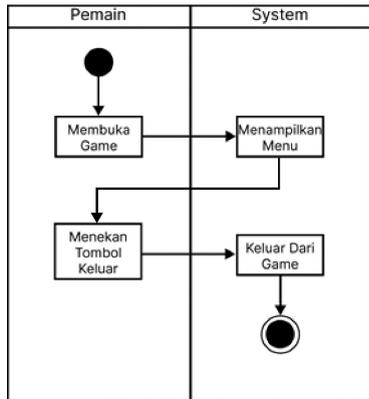


Gambar 9. Activity Diagram Join Room

Gambar 9 menjelaskan *activity diagram join room*, diawali dengan pemain membuka *game*, kemudian *system* akan menampilkan menu utama, selanjutnya pemain memasukkan nama *room* dan menekan tombol *join room* yang selanjutnya *system* akan menampilkan *map* permainan dari *room* yang dituju, kemudian pemain menekan tombol *spawn* maka *system* akan memulai permainan

5. *Activity Diagram Keluar*

Activity diagram keluar merupakan aktivitas yang akan mengeluarkan pemain dari aplikasi permainan ketika pemain menekan tombol keluar. Gambar 10 merupakan *activity diagram* keluar.

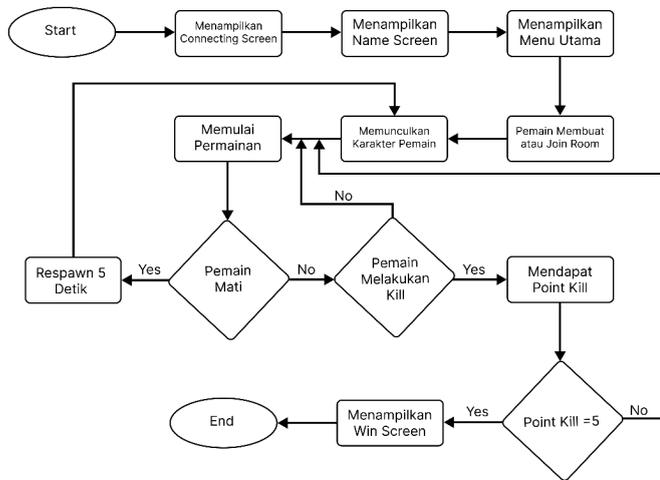


Gambar 10. *Activity Diagram Keluar*

Gambar 10 menjelaskan *activity diagram* keluar, diawali dengan pemain membuka *game*, kemudian *system* akan menampilkan menu utama, selanjutnya pemain menekan tombol keluar yang mana *system* akan langsung keluar dari *game*.

D. Rancangan *Blok Diagram Alur Game*

Pada tahap ini akan menjelaskan bagaimana alur dari *game side scrolling war* melalui rancangan *blok diagram*. Gambar 6 merupakan *blok diagram* alur *game*.



Gambar 11. *Blok Diagram Alur Game*

Gambar 11 menjelaskan bagaimana sistem *game* berjalan. Diawali dengan pemain masuk ke dalam *game*, kemudian sistem akan menampilkan *connecting screen* setelah berhasil tersambung ke *server* sistem akan menampilkan *name screen* agar pemain mengisi *username* yang ingin digunakan, setelah pemain mengisi *username* sistem akan menampilkan menu utama, kemudian dari menu utama pemain akan membuat atau melakukan *join room*, setelah berhasil membuat atau *join room* sistem akan memunculkan karakter untuk masing-masing pemain. Pemain akan mulai berperang satu sama lain jika salah satu pemain mati maka sistem akan melakukan *respawn* terhadap pemain tersebut dan memberikan *point kill* untuk pemain yang melakukan *kill*. Jika salah satu pemain telah mengumpulkan 5 *point kill*, maka

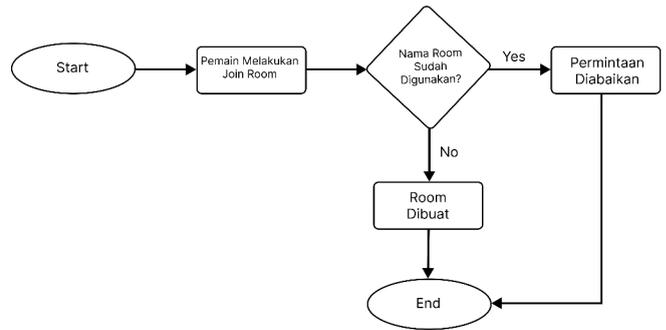
sistem akan menampilkan *win screen* dan permainan pun berakhir.

E. Rancangan *Blok Diagram Alur Create dan join room PUN*

Pada tahap ini akan menjelaskan alur kerja dari *create dan join room* yang dilakukan pada *photon uniry networking* ketika pemain membuat atau bergabung dengan *room*.

1. *Create Room*

Pada tahap ini menjelaskan bagaimana alur dari *create room* ketika pemain membuat *room*. Gambar 12 merupakan alur kerja *create room*.

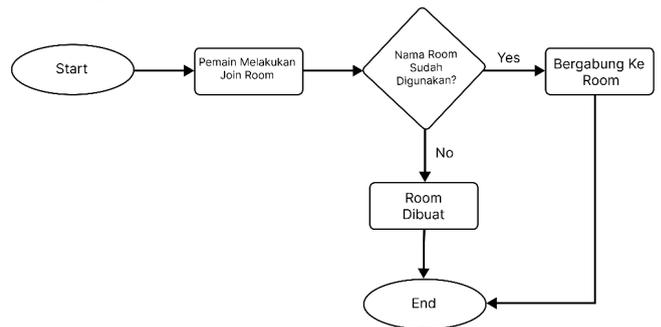


Gambar 12. *Blok Diagram Alur Create Room*

Gambar 12 menjelaskan alur kerja *create room*. Diawali saat *game* dimulai kemudian ketika pemain membuat *room*, maka *photon unity networking* akan melakukan *check* terhadap nama *room* yang di *input* oleh pemain, jika nama *room* tersebut telah digunakan oleh pemain lain atau *room* dengan nama tersebut telah ada, maka permintaan pemain untuk membuat *room* dengan nama tersebut akan diabaikan, namun, jika nama tersebut belum digunakan maka *photon unity networking* akan membuat *room* untuk pemain dengan nama tersebut.

2. *Join Room*

Pada tahap ini menjelaskan bagaimana alur dari *join room* ketika pemain bergabung ke dalam *room*. Gambar 13 merupakan alur kerja *join room*.



Gambar 13. *Blok Diagram Alur Join Room*

Gambar 13 menjelaskan alur kerja *join room*. Diawali saat *game* dimulai kemudian ketika pemain melakukan *join room*, maka *photon unity networking* akan melakukan *check* terhadap nama *room* yang di *input* oleh pemain, jika nama *room* tersebut telah digunakan oleh pemain lain atau *room* dengan nama tersebut telah ada, maka pemain akan bergabung ke dalam *room* tersebut, namun, jika nama tersebut belum digunakan maka *photon unity networking* akan membuat *room* untuk pemain dengan nama tersebut.

III. HASIL DAN PEMBAHASAN

A. Hasil

Tahap ini merupakan tahap yang berisi tentang penjelasan bagaimana *game* ini dapat bekerja sesuai dengan yang diharapkan dan dapat berjalan dengan baik. Tahapan ini meliputi perancangan perangkat lunak, bagian program yang penting dan implementasi sehingga dapat dipahami dengan baik dan mengetahui cara menggunakannya.

1. Tampilan *Connecting to Server*

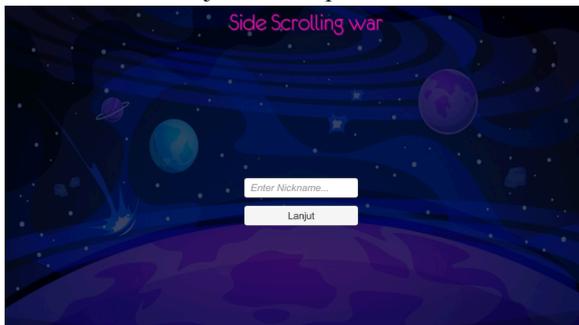
Tampilan *connecting to server* adalah tampilan awal saat pertama kali pemain membuka *game*, pada tampilan ini akan menampilkan tulisan “menghubungkan ke server” dan ketika *game* berhasil terhubung maka sistem akan otomatis menampilkan tampilan *name screen*. Gambar 14 menunjukkan tampilan *connecting to server*.



Gambar 14. Tampilan *Connecting to Server*

2. Tampilan *Name Screen*

Tampilan *name screen* adalah tampilan yang akan ditampilkan ketika *game* telah berhasil terhubung ke server. Tampilan ini akan menampilkan kolom untuk pemain menginput nama dan ketika pemain telah menginput nama maka akan muncul tombol “lanjut”. Gambar 15 menunjukkan tampilan *name screen*.



Gambar 15. Tampilan *Name Screen*

3. Tampilan Info

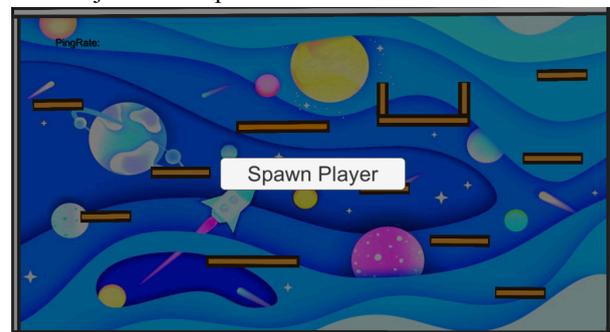
Pada tampilan ini akan menampilkan info atau cara bermain dan mekanisme dari *game side scrolling war* serta tombol “kembali”. Gambar 16 menunjukkan tampilan info.



Gambar 16. Tampilan Info

4. Tampilan *Room*

Pada tampilan ini akan menampilkan background berupa map dari *game side scrolling war* serta tombol “spawn player” ketika tombol ditekan maka akan memuat karakter pemain dan permainan pun dimulai. Gambar 17 menunjukkan tampilan *room*.



Gambar 17. Tampilan *Room*

5. Tampilan *Gameplay*

Pada tampilan ini akan memuat map dari *game side scrolling war*, karakter beserta nama dan *healthbar* dari pemain serta akan memuat peluru jika karakter melakukan aksi menembak. Gambar 18 menampilkan tampilan *gameplay*.



Gambar 18. Tampilan *Gameplay*

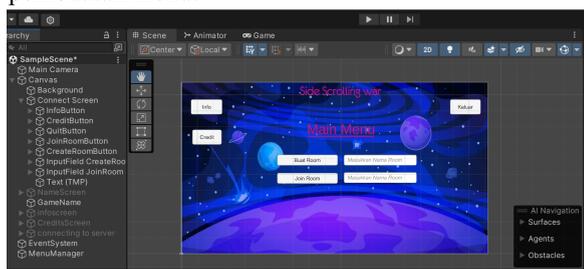
B. Pembahasan

Tahap ini merupakan tahap yang menguraikan bagaimana permainan ini beroperasi sesuai dengan yang diharapkan dan menjalankan desain yang telah dirancang. Pada tahap ini, mencakup proses pembuatan *game*, map *game* dan pemrograman.

1. Pembuatan Menu

Pada tahap ini melakukan pembuatan menu yang nantinya menu ini akan berinteraksi langsung dengan pemain. Menu pada *game side scrolling war* sendiri terdiri dari *background*, tombol “credit”, tombol “info”, tombol

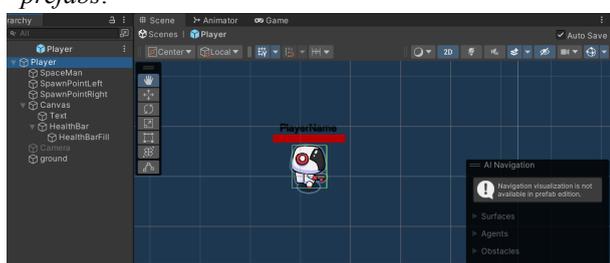
“buat room”, tombol “join room”, tombol “keluar” dan 2 kolom pengisian nama room. Gambar 19 menunjukkan pembuatan menu.



Gambar 19. Pembuatan Menu

2. Pembuatan Player Prefabs

Pada tahap pembuatan *player prefabs* merupakan tahap yang sangat penting, tahap ini diperlukan untuk membuat karakter yang nantinya akan dikontrol oleh pemain saat bermain. Gambar 20 menunjukkan pembuatan *player prefabs*.



Gambar 20. Pembuatan Player Prefabs

3. Pembuatan Animasi Diam

Pembuatan animasi diam berfungsi untuk membuat tampilan karakter bergerak seolah-olah seperti sedang bernapas ketika karakter tidak melakukan aksi apapun. Gambar 21 menunjukkan pembuatan animasi diam.



Gambar 21. Pembuatan Animasi Diam

4. Pembuatan Animasi Lari

Animasi berlari berfungsi untuk membuat tampilan karakter bergerak khususnya kaki karakter akan bergerak seperti sedang berlari ketika karakter melakukan aksi ini. Gambar 22 menunjukkan pembuatan animasi lari.



Gambar 22. Pembuatan Animasi Lari

5. Pembuatan Healthbar

Setiap karakter nantinya memiliki *healthbar* yang berisi penuh dan ketika *healthbar* ini menurun kurang dari nol maka karakter akan mati dan karakter akan di hilangkan sesaat dari permainan. Gambar 23 merupakan desain dari *healthbar*.



Gambar 23. Pembuatan Healthbar

6. Pembuatan Peluru

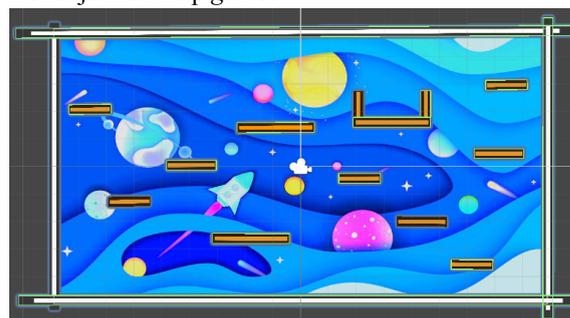
Peluru merupakan objek yang nantinya akan memberikan *damage* kepada karakter dan ketika karakter menerima *damage* dari peluru maka *healthbar* akan menurun. Gambar 24 merupakan desain dari peluru.



Gambar 24. Pembuatan Peluru

7. Pembuatan Map Game

Tahap ini merupakan tahap pembuatan map yang akan digunakan saat bermain. Map yang akan dibuat pada *game side scrolling war* terdiri dari beberapa bahan yaitu *background*, dan beberapa blok sebagai *ground* atau pijakan bagi karakter di dalam *game*. Gambar 25 menunjukkan map *game*.



Gambar 25. Pembuatan Map Game

C. Hasil Pengujian Black-Box

Pada tahap ini akan menampilkan hasil dari pengujian *blackbox* yang telah dilakukan oleh dua pemain, pengujian ini bertujuan untuk menguji fungsional yang meliputi tombol-tombol dan komponen yang ada di dalam *game side scrolling war*.

1. Tombol Credit

Pada tahap ini melakukan pengujian *blackbox* terhadap tombol *credit*. Tabel III menunjukkan hasil pengujian *blackbox* terhadap tombol *credit*.

TABEL III
TOMBOL CREDIT

No	Aktivitas Penguji	Hasil yang Diharapkan	pemain	Keterangan
1	Tombol <i>credit</i>	Menampilkan <i>credit game</i> .	1	Berhasil
			2	Berhasil
			3	Berhasil
			4	Berhasil

2. Tombol Info

Pada tahap ini melakukan pengujian *blackbox* terhadap tombol info. Tabel IV menunjukkan hasil pengujian *blackbox* terhadap tombol info.

TABEL IV
TOMBOL INFO

No	Aktivitas Penguji	Hasil yang Diharapkan	pemain	Keterangan
1	Tombol info	Menampilkan info <i>game</i> .	1	Berhasil
			2	Berhasil
			3	Berhasil
			4	Berhasil

3. Tombol Buat Room

Pada tahap ini melakukan pengujian *blackbox* terhadap tombol buat room. Tabel V menunjukkan hasil pengujian *blackbox* terhadap tombol buat room.

TABEL V
TOMBOL BUAT ROOM

No	Aktivitas Penguji	Hasil yang Diharapkan	pemain	Keterangan
1	Tombol buat room	Menampilkan map <i>game</i> dan tombol "Spawn Player".	1	Berhasil
			2	Berhasil
			3	Berhasil
			4	Berhasil

4. Tombol Join Room

Pada tahap ini melakukan pengujian *blackbox* terhadap tombol join room. Tabel VI menunjukkan hasil pengujian *blackbox* terhadap tombol join room.

TABEL VI
TOMBOL JOIN ROOM

No	Aktivitas Penguji	Hasil yang Diharapkan	pemain	Keterangan
1	Tombol join room	Menampilkan map <i>game</i> dan tombol "Spawn Player".	1	Berhasil
			2	Berhasil
			3	Berhasil
			4	Berhasil

5. Tombol Keluar Room

Pada tahap ini melakukan pengujian *blackbox* terhadap tombol keluar room. Tabel VII menunjukkan hasil pengujian *blackbox* terhadap tombol keluar room.

TABEL VII
TOMBOL KELUAR ROOM

No	Aktivitas Penguji	Hasil yang Diharapkan	pemain	Keterangan
1	Tombol keluar room	Menampilkan halaman awal.	1	Berhasil
			2	Berhasil
			3	Berhasil
			4	Berhasil

6. Tombol Keluar Game

Pada tahap ini melakukan pengujian *blackbox* terhadap tombol keluar game. Tabel VIII menunjukkan hasil pengujian *blackbox* terhadap tombol keluar game.

TABEL VIII
TOMBOL KELUAR GAME

No	Aktivitas Penguji	Hasil yang Diharapkan	pemain	Keterangan
1	Tombol keluar game	Keluar dari aplikasi permainan.	1	Berhasil
			2	Berhasil
			3	Berhasil
			4	Berhasil

Hasil perhitungan persentase dari pengujian *blackbox* dapat dilihat pada tabel IX.

TABEL IX
PERHITUNGAN PENGUJIAN BLACK-BOX

No	Pengguna Berhasil	Pengguna Tidak Berhasil	Persentase	Hasil
A01	4	0	4/4×100 %	100%

B01	4	0	4/4×100 %	100%
C01	4	0	4/4×100 %	100%
D01	4	0	4/4×100 %	100%
E01	4	0	4/4×100 %	100%
F01	4	0	4/4×100 %	100%
Rata-Rata persentase				100%

D. Hasil Pengujian Ping

Pada tahap ini dilakukan pengujian sebanyak seribu ping, ini dilakukan dengan mengirimkan paket data ke server dan mengukur berapa lama waktu yang diperlukan untuk menerima respon. Hasilnya diukur dalam *milisecond* (MS). Pengujian ini dilakukan dengan cara menambahkan *code* perintah untuk memuat hasil ping pada *console* unity. Tabel X menunjukkan hasil dari 4 percobaan dengan *loop* sebanyak 100 ping.

TABEL X
HASIL PENGUJIAN PING

No	Percobaan	Min	Max	Average
1	Percobaan 1	33ms	44ms	35ms
2	Percobaan 2	40ms	47ms	45ms
3	Percobaan 3	38ms	46ms	41ms
4	Percobaan 4	36ms	48ms	42ms
Rata-rata Ping				41ms

E. Hasil Pengujian Multiplayer Online

Pada pengujian *multiplayer online* ini melibatkan 2 pemain untuk melakukan pengujian secara *online* dengan menggunakan jaringan yang berbeda.

1. Pengujian Jika Pemain Offline

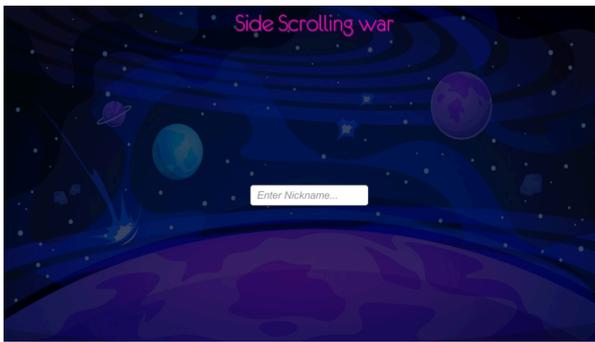
Pada pengujian ini akan menguji apakah *game* dapat berjalan jika pemain tidak terhubung ke jaringan internet, hasil dari pengujian ini adalah *game* akan menampilkan teks "Gagal menghubungkan ke server", hasilnya dapat dilihat pada gambar 26.



Gambar 26. Pengujian Jika Pemain Offline

2. Pengujian Jika Pemain Online

Pada pengujian ini akan menguji apakah *game* dapat berjalan jika pemain tidak terhubung ke jaringan, hasil dari pengujian ini adalah *game* akan menampilkan teks "Menghubungkan ke server" beberapa detik kemudian akan menampilkan *name screen*, hasilnya dapat dilihat pada gambar 27.



Gambar 27. Pengujian Jika Pemain Online

3. Pengujian Buat Room

Pada pengujian ini melibatkan pemain satu. Pemain satu akan masuk ke dalam *game* dan membuat *room* dengan nama "Room Satu" seperti yang ditampilkan pada gambar 28.



Gambar 28. Pengujian Buat Room

4. Pengujian Berhasil Buat Room

Setelah sebelumnya pemain satu membuat *room* dengan nama "Room Satu" kemudian hasilnya adalah pemain satu berhasil membuat *room* dan masuk ke permainan seperti yang ditampilkan pada gambar 29.



Gambar 29. Pengujian Berhasil Buat Room

5. Pengujian Join Room

Pada pengujian ini melibatkan pemain dua. Pemain dua akan masuk ke dalam *game* dan akan melakukan *join room* dengan nama "Room Satu" seperti yang ditampilkan pada gambar 30.



Gambar 30. Pengujian Join Room

6. Pengujian Gameplay

Setelah pemain satu dan pemain dua berhasil terhubung selanjutnya pada tahap ini akan melakukan pengujian *gameplay* dimana kedua pemain akan saling menembak dan mengumpulkan *point kill*. Gambar 31 menunjukkan pengujian *gameplay*.



Gambar 31. Pengujian Gameplay

IV. SIMPULAN

Berdasarkan penelitian yang telah dilakukan oleh penulis, dapat diambil kesimpulan bahwa "Rancang Bangun Game "Side Scrolling War" Online Multiplayer 2D" berhasil dilakukan dengan kesimpulan pada pengujian kelayakan fungsional *game* menggunakan teknik pengujian *blackbox* diperoleh nilai persentase dari hasil pengujian tersebut sebesar 100%, yang didapat dari 6 pengujian yaitu yaitu pengujian tombol *credit*, tombol *info*, tombol *buat room*, tombol *join room*, tombol *keluar room* dan tombol *keluar game*, yang artinya seluruh pemain yang terlibat dalam pengujian tersebut berhasil dalam pengujian ini dan dapat disimpulkan bahwa *game side scrolling war* layak sebagai sarana hiburan. Hasil dari pengujian ping menunjukkan respon yang sangat baik dari photon dengan menunjukkan nilai rata-rata sebesar 41 ms yang berarti nilai tersebut sangat baik untuk *game* yang memerlukan aksi *real-time* atau komunikasi cepat antar pemain. Dari hasil pengujian *multiplayer online* dapat disimpulkan bahwa photon unity *networking* berhasil menyambungkan dua pemain ke dalam satu *room* dan dapat bermain bersama, yang berarti pengujian *multiplayer online* telah berhasil dan *game* dapat dimainkan secara *multiplayer online*.

REFERENSI

- [1] M. Fikri, "Rancangan Bangun Aplikasi Game Battleship Multiplayer pada Jaringan LAN," p. 95, 2013.
- [2] N. Ramsari and G. Ramadhan, "Pembuatan Game Side Scrolling 2D the Naila'S Survival Berbasis Android," *Jurnal FIKI*, vol. VIII, no. 2, pp. 2087–2372, 2018, [Online]. Available: <http://jurnal.unnur.ac.id/index.php/jurnalfiki>
- [3] F. C. Ningrum, D. Suherman, S. Aryanti, H. A. Prasetya and A. Saifudin, "Pengujian Black Box pada Aplikasi Sistem Seleksi Sales

- Terbaik Menggunakan Teknik Equivalence Partitions,” vol. 4, no. 4, 2019, [Online]. Available: <http://openjournal.unpam.ac.id/index.php/informatika>
- [4] E. Hutabri, P. A. Dasa “Perancangan Media Pembelajaran Interaktif Berbasis Android pada Mata Pelajaran Ilmu Pengetahuan Sosial untuk Anak Sekolah Dasar,” *Jurnal Sustainable: Jurnal Hasil Penelitian dan Industri Terapan*, vol. 08, no. 02, pp. 57–64, 2019.
- [5] L. Stefano Mongi, A. S. M Lumenta and A. M. Sambul, “Rancang Bangun Game Adventure of Unsrat Menggunakan Game Engine Unity,” *Journal Teknik Informatika*, vol. 14, no. 1, 2018.
- [6] L. Stefano Mongi, A. S. M Lumenta, and A. M. Sambul, “Rancang Bangun Game Adventure of Unsrat Menggunakan Game Engine Unity,” *Journal Teknik Informatika*, vol. 14, no. 1, 2018.
- [7] I. I. Purnomo, “Aplikasi Game Edukasi Lingkungan Agen P Vs Sampah Berbasis Android Menggunakan Construct 2,” *Technologia: Jurnal Ilmiah*, vol. 11, no. 2, p. 86, 2020, doi: 10.31602/tji.v11i2.2784.
- [8] A. Muhammad, I. Kuswardayan and R. Rahman, “Design and Implementation Synchronous Pvp Multiplayer Online in Rangers Companion of Social Game Using Unity and Photon Unity Networking Framework on Android Device,” 2015.
- [9] D. P. Sari, I. Arfyanti and S. E. Yunita, “Membangun Game Side Scroller Cerita Rakyat “Kisah Timun Mas” Untuk Lingkungan SDN 007 Samarinda Berbasis Android,” 2023.