

PERANCANGAN GAME BERBASIS ANDROID “KNIGHT THE DEVIL SLAYER” MENGUNAKAN METODE PATHFINDING

Fiqih Diouf Ghiffary, Hani Zulfia Zahro', Febriana Santi Wahyuni

Program Studi Teknik Informatika S1, Fakultas Teknologi Industri
Institut Teknologi Nasional Malang, Jalan Raya Karanglo km 2 Malang, Indonesia
2018011@scholar.itn.ac.id

ABSTRAK

Perkembangan teknologi smartphone yang pesat telah membuat game Android menjadi hiburan populer. Seiring meningkatnya kompleksitas game, terutama yang melibatkan eksplorasi atau pencarian jalan (Pathfinding), muncul tantangan baru dalam pengembangannya. Tantangan utama adalah menciptakan AI yang mampu menemukan jalur terbaik dalam lingkungan dinamis. Penelitian ini bertujuan untuk merancang dan menerapkan algoritma Pathfinding A* dalam Game 2D di platform Android menggunakan Godot Game Engine. Algoritma A* dipilih karena menggabungkan pencarian berdasarkan kedekatan dan pencarian biaya terkecil, memungkinkan pencarian jalur terpendek secara efisien. Godot Game Engine menyediakan fitur dan alat yang mempermudah pengembangan game, termasuk dukungan untuk algoritma pathfinding seperti A*. Penelitian ini membahas penerapan metode Pathfinding A* dalam game "Knight The Devil Slayer". Hasil pengujian menunjukkan bahwa mayoritas pemain memberikan penilaian positif, dengan 99.62% responden menyatakan game ini bagus dan 0.38% menyatakan buruk. Hasil ini menunjukkan bahwa penggunaan metode Pathfinding A* dapat menghasilkan game yang dinamis dan menarik, serta diterima dengan baik oleh para pemain. Penelitian ini diharapkan dapat membantu perkembangan teknologi game di Indonesia dan memberikan wawasan baru bagi pengembang game tentang metode Pathfinding A* yang efektif dan efisien, serta meningkatkan kualitas pengalaman bermain game bagi pengguna. Dengan implementasi yang tepat, game dapat menjadi lebih menarik dan memuaskan bagi para pemain.

Kata kunci : Knight, Game, Pathfinding, Android.

1. PENDAHULUAN

Saat ini, pengembangan game Android sedang berkembang dengan cepat. Perangkat lunak dan teknologi canggih saat ini memungkinkan generasi muda untuk membuat game mereka sendiri. Pembuat game dapat dengan mudah membuat game berbasis dua dimensi atau tiga dimensi untuk platform desktop dan Android dengan Godot [1].

Pathfinding mengacu pada konsep menemukan jalur optimal dari simpul sumber ke simpul tujuan dalam waktu yang paling singkat. Beberapa Algoritma telah dirancang untuk jalur terpendek dari sumber ke tujuan dengan mencoba menghindari semua rintangan di jalan. Mereka juga dapat menggunakan AI untuk pathfinding. Komponen karakter khusus yang digunakan dalam permainan disebut NPC atau karakter non-pemain [2].

Salah satu algoritma Pathfinding yang paling populer dan efektif adalah A* (A-star). Algoritma ini menggabungkan dua pendekatan utama: pencarian berdasarkan kedekatan (greedy search) dan pencarian biaya terkecil (uniform-cost search). A* menggunakan fungsi heuristik untuk memperkirakan biaya terendah dari titik awal ke titik tujuan, memungkinkan algoritma ini untuk menemukan jalur terpendek secara efisien. A* menjadi pilihan utama dalam banyak aplikasi karena keseimbangan yang baik antara optimalitas dan efisiensi waktu pencarian. Algoritma ini tidak hanya mampu menemukan solusi optimal tetapi juga melakukannya dalam waktu yang relatif

singkat, menjadikannya sangat cocok untuk aplikasi real-time seperti game [3].

Oleh karena itu, penggunaan metode Pathfinding A* dalam pengembangan game 2D di platform Android, dengan bantuan tools seperti Godot Game Engine, dapat menghasilkan permainan yang lebih dinamis dan menarik. Algoritma A* memungkinkan karakter dalam game untuk menemukan jalur optimal dengan efisien, menghindari rintangan, dan beradaptasi dengan perubahan lingkungan. Dengan memanfaatkan kemampuan dan fitur dari Godot Game Engine, proses pengembangan menjadi lebih mudah dan cepat, serta menghasilkan game dengan performa yang optimal.

Jadi penelitian ini akan membahas perancangan dan penerapan menemukan jalur terbaik dalam game Knight The Devil Slayer dengan menggunakan metode Pathfinding A*. Diharapkan penelitian ini akan membantu perkembangan teknologi game di Indonesia dan memberi para pengembang game wawasan baru tentang cara menggunakan metode Pathfinding A* yang efektif.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terdahulu

Pada penelitian [4] dengan judul “Real-Time Pathfinding for Non-Player Characters in Open-World Games”. teknik pathfinding real-time untuk karakter non-pemain (NPC) dalam game dunia terbuka dengan menggunakan pendekatan berbasis graf yang dioptimalkan. Penelitian ini mengungkapkan bahwa

metode yang diusulkan memungkinkan NPC untuk menavigasi lingkungan yang sangat besar dan kompleks dengan efisiensi yang tinggi. Algoritma ini mengatasi tantangan dalam penanganan jalur di lingkungan yang luas dan dinamis, dengan meningkatkan kinerja NPC dalam merespons perubahan lingkungan dan berinteraksi secara lebih alami dengan pemain. Penerapan metode ini berdampak positif pada pengalaman bermain game dunia terbuka yang luas.

Pada penelitian [5] dengan judul “Penerapan Metode Pathfinding Pada Pengembangan Game The Book of Aksara Pada Perangkat Bergerak”. Jika Pemain berada di dalam area deteksi musuh, karakter Karakter Non-Pemain musuh memiliki kemampuan untuk bergerak dan bereaksi saat mengejar Pemain. Titik area deteksi karakter Karakter Non-Pemain musuh memungkinkan mereka untuk mengikuti titik terakhir posisi musuh saat mengejar Pemain, yang membuat Karakter Non-Pemain musuh lebih aktif dan responsif. Menurut hasil uji coba, metode Pathfinding dikombinasikan dengan metode Finite State Machine berhasil diterapkan dan menghasilkan peningkatan efisiensi sebesar 8,3 persen.

Pada penelitian [6] dengan judul “Penyelesaian Pathfinding dalam Game Maze menggunakan metode Backtracking”. Metode backtracking dapat menjadi alat yang berguna dalam mengatasi masalah permainan labirin dalam menentukan jalur untuk mencapai titik akhir. Metode backtracking dapat dengan mudah diterapkan, dan efeknya terlihat jelas pada kemampuan pemain untuk menyelesaikan tugas yang ada dan melarikan diri dari labirin. Untuk meningkatkan skalabilitas solusi permainan labirin, terdapat ruang untuk perbaikan, seperti penambahan algoritma kecerdasan buatan dan penyelidikan skenario dalam permainan lainnya.

Pada penelitian [7] dengan judul “Pathfinding Dinamis untuk pengikut Karakter Non-Pemain di Game”. Berdasarkan penelitian Dynamic Pathfinding diperoleh kesimpulan bahwa Metode yang diusulkan dalam penelitian ini dapat mengurangi waktu komputasi pada permasalahan pencarian jalur dinamis. Pengujian dalam lingkungan simulasi menunjukkan hasil yang lebih cepat dalam pencarian jalur dinamis dibandingkan algoritma Astar dan DPC. Bagi penelitian selanjutnya, penerapan fungsi medan potensial atraktif dan perilaku berkelompok benda dapat dikembangkan tidak hanya pada lingkungan dua dimensi saja tetapi juga lingkungan tiga dimensi. Oleh karena itu, ini dapat berguna tidak hanya untuk pengikut Karakter Non-Pemain dalam karakter berjalan tetapi juga untuk karakter terbang sebagai pesawat udara dalam game. Algoritma yang diusulkan dapat dikembangkan untuk sekumpulan pengikut Karakter Non-Pemain sehingga tidak hanya ada satu pengikut Karakter Non-Pemain yang melakukan pencarian jalur dinamis.

Pada penelitian [8] dengan judul "Implementasi Metode Pathfinding dengan Penerapan Algoritma A-

Star untuk Mencari Jalur Terpendek pada Game Jumrah Launch Story", hasilnya menunjukkan bahwa penggunaan Algoritma A-Star berhasil. Karakter Non-Pemain yang tidak memiliki algoritma A-Star hanya dapat bergerak sesuai dengan program awal mereka, tetapi Karakter Non-Pemain dengan algoritma ini dapat mengejar dan membunuh Pemain saat mereka tidak berhati-hati. Mereka juga dapat mengejar walaupun ada penghalang atau Pemain bersembunyi.

2.2. Game

Game merupakan salah satu media hiburan yang banyak diminati oleh kalangan masyarakat, dari anak usia dini hingga orang dewasa. Secara umum game dibagi menjadi 7 (tujuh) jenis yang diterapkan yaitu, RTS (Real Time Strategy), FPS (First Person Shooter), RPG (Role Playing Game), Construction and Management Simulation Games, Vehicle Simulation, Action, Adventure dan Fighting. [9].

2.3. Godot

Godot adalah mesin permainan sumber terbuka yang menyediakan pengembangan permainan lintas platform dalam format 2D dan 3D. Mesin ini memiliki kemampuan penulisan skrip yang luas dalam berbagai bahasa, pengoptimalan kinerja, fitur jaringan, dan sistem adegan berbasis node untuk membuat level dan lingkungan. Mesin permainan Godot menyediakan pengoptimalan dan dukungan yang luas untuk menyebarkan permainan di berbagai platform, termasuk web. UI Godot menyediakan akses langsung ke pembangun adegan, sistem file, dan inspektur node untuk membangun lingkungan permainan dengan ketelitian tinggi dengan mudah. [10].

Godot adalah mesin game sumber terbuka yang menyediakan pengembangan game lintas platform dalam 2D dan 3D. Godot memiliki kemampuan skrip yang luas dalam berbagai bahasa, pengoptimalan kinerja, fitur jaringan, dan sistem adegan berbasis node untuk membuat level dan lingkungan. Mesin permainan Godot memberikan optimalisasi dan dukungan ekstensif untuk menerapkan permainan di berbagai platform, termasuk web.

Mesin permainan Godot menyediakan banyak fitur dan kemampuan yang ideal untuk pengembangan game tingkat lanjut dan alat untuk membuat game 2D dan 3D. Selain itu, Godot menyediakan fitur yang sesuai untuk membangun lingkungan pembelajaran mesin yang lebih kompleks.

- a. Dukungan multi-platform: Godot mendukung berbagai platform termasuk Windows, MacOS, Linux, Android, dan HTML. Hal ini memungkinkan pengembang untuk menciptakan lingkungan pelatihan untuk sistem operasi.
- b. Sumber Terbuka: Godot sepenuhnya bersumber terbuka, dan pengembang merilisnya di bawah lisensi MIT. Hal ini memungkinkan pengembang dan peneliti untuk membuat aplikasi tanpa batasan hukum. Godot dapat dikustomisasi karena kode

- c. Scripting: Godot dikembangkan dalam berbagai bahasa. Ini memiliki bahasa skrip bawaan yang disebut GDScript, mirip dengan Python. Selain itu, ini mendukung bahasa pengembangan lainnya, seperti C#, C++, dan Visual Script. Ini menyediakan berbagai bahasa pemrograman untuk tujuan pekerjaan dan pengembangan.
- d. Mesin Fisika: Godot mencakup mesin fisika 2D dan 3D. Secara default, Godot menggunakan mesin fisika Bullet dengan ketelitian tinggi. Selain itu, berisi mesin fisika fidelitas rendah yang disebut GodotPhysics.
- e. Jaringan: Mesin Godot memberikan dukungan ekstensif untuk jaringan klien-server dan peer-to-peer untuk membuat game multipemain.
- f. Sistem adegan berbasis node: Mesin Godot memungkinkan konstruksi lingkungan atau adegan menggunakan sistem berbasis node yang membuat hierarki antar node.

Fitur-fitur di atas memberikan titik awal yang sangat baik untuk menciptakan lingkungan virtual untuk pembelajaran DRL. Kemampuan untuk memilih mesin fisika yang berbeda, fitur jaringan, dan dukungan multi-platform menyediakan aksesibilitas untuk mengendalikan simulasi.

2.4. Pathfinding Algoritma A*

Algoritma A-star (A*) termasuk dalam kategori metode pencarian yang mempunyai informasi. Algoritma ini sangat bagus untuk menyelesaikan proses pencarian jalur. Algoritma ini menghitung jarak tercepat yang dapat ditempuh oleh titik awal menuju objek tujuan [10].

Algoritma A* memerlukan dua model antrian, yaitu terbuka dan tertutup. Algoritma A* memiliki kemampuan untuk melakukan prediksi pada setiap node yang dibuat dengan mengubah fungsi heuristiknya. Langkah ini dilakukan untuk menyederhanakan algoritma yang akan digunakan untuk menentukan langkah selanjutnya. Metode ini dikenal sebagai pendekatan fungsi $f(n)$, di mana fungsi $f(n)$ adalah fungsi evaluasi sebenarnya terhadap node (n) . Rumus fungsi matematika A* dapat ditulis sebagai persamaan 1 [11].

$$f(n) = G(n) + H(n) \dots \dots \dots (1)$$

Informasi:

$F(n)$: nilai simpul

$G(n)$: jarak antar node (n)

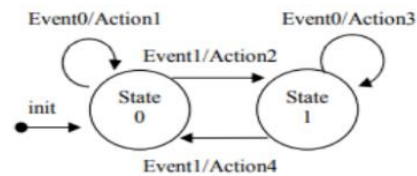
$H(n)$: perkiraan waktu yang diperlukan untuk mencapai tujuan.

Sedangkan untuk memperoleh jalur tercepat pada robot ditentukan berdasarkan putaran roda yang tertanam pada encoder. Nilai encoder ini kemudian dijumlahkan sehingga diperoleh nilai terkecil dari penjumlahannya. Jika putaran encoder tidak diketahui maka pembacaan nilai node dapat diperoleh dari persamaan 2 [10].

$$node = \frac{\text{heuristic value}}{\text{speed}} \dots \dots \dots (2)$$

2.5. FSM (Finite State Machine)

Finite State Machine Merupakan sebuah metodologi perancangan sistem kontrol yang terdiri dari gambaran tingkah laku atau prinsip sistem dengan menggunakan tiga hal yaitu state (keadaan), event (kejadian), dan action (aksi). Metode pengembangan sistem yang digunakan.[12].



Gambar 1. Diagram finite state machine

Pada Gambar 1. menunjukkan mesin finite state yang memiliki empat tindakan keluaran, dua keadaan (state), dua masukan, dan dua tindakan keluaran yang berbeda. Seperti yang ditunjukkan pada gambar, ini akan muncul ketika sistem dimulai. Dalam situasi ini, sistem menampilkan action 1 ketika event 0 dimasukkan; namun, ketika event 1 dimasukkan, action 2 selesai. Setelah itu, sistem kembali ke keadaan 1 dan seterusnya.

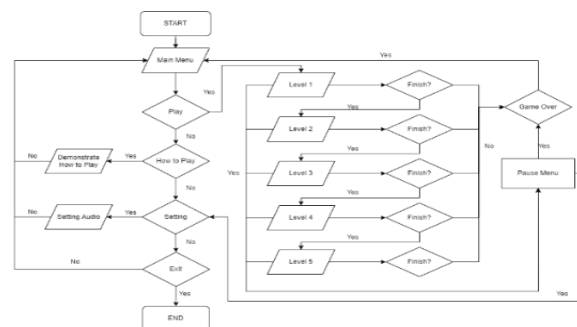
3. METODE PENELITIAN

3.1. Flowchart Game

Setelah program dimulai, pengguna akan dihadapkan dengan menu utama yang memiliki beberapa pilihan, yaitu New Game, Cara Bermain, Pengaturan, dan Exit. Jika pengguna memilih "Cara Bermain", maka akan ditampilkan cara bermain. Namun, jika pengguna memilih "Pengaturan", maka akan muncul pilihan untuk mengatur audio. Setelah itu, pengguna akan kembali ke menu utama.

Dalam menu utama, jika pengguna memilih New Game, maka program akan memulai permainan dengan tingkat kekompleksan yang akan bertambah dari level 1 sampai level 5. Jika pengguna gagal mengatasi stage, maka permainan akan berakhir. Namun, jika pengguna berhasil, maka akan lanjut ke level selanjutnya. Jika permainan belum selesai, maka pengguna akan kembali ke menu jeda dan dapat memilih untuk melanjutkan permainan atau kembali ke main menu.

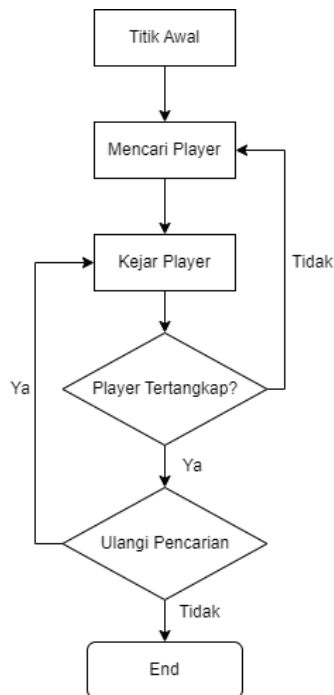
Jika pengguna memilih Exit di menu utama, maka akan secara langsung keluar dari game.



Gambar 2. Flowchart game

3.2. Alur Pathfinding

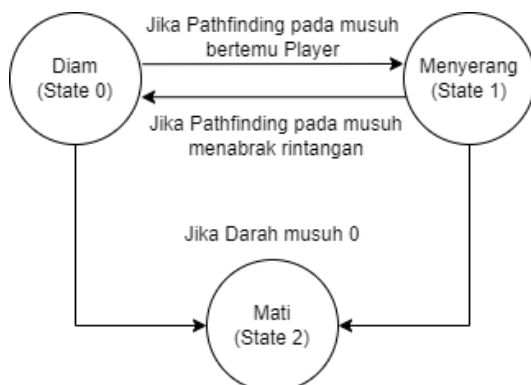
Berikut adalah perancangan alur metode Pathfinding untuk musuh dalam game “Knight The Devil Slayer”:



Gambar 3 Diagram Pathfinding pada musuh

Gambar 3. menunjukkan titik awal di mana musuh memulai operasinya. Musuh memulai dengan mencari keberadaan pemain (Mencari Pemain). Setelah ditemukan, musuh akan mulai mengejar pemain (Kejar Pemain). Tahap pengejaran ini menentukan apakah pemain telah ditangkap. Proses berakhir jika pemain tertangkap. Namun, jika pemain masih belum tertangkap, musuh akan mengulangi pencarian (Ulangi Pencarian) dan kembali ke langkah mencari pemain. Proses ini akan berlanjut sampai pemain tertangkap.

3.3. Alur Finite State Machine



Gambar 4. Diagram FSM musuh

Game ini memiliki tiga negara utama: negara diam sebagai negara 0, negara menyerang sebagai

negara 1, dan negara mati sebagai negara 2. Ketika pemain berada di map, mereka dapat menerapkan aliran FSM pada musuh. Jika Pathfinding pada musuh bertemu Pemain, musuh akan memasuki state menyerang. Jika Pathfinding pada musuh menabrak rintangan, musuh akan memasuki state diam dan Pathfinding akan mencari kembali Pemain. Jika darah musuh 0 maka musuh akan mati.

3.4. Perancangan Karakter

Karakter pemain adalah karakter yang digunakan pemain didalam game Knight The Devil Slayer. Desain karakter pemain dapat dilihat pada Gambar dibawah ini.



Gambar 5. Desain Karakter Knight

Pada gambar 5. merupakan Karakter tersebut merupakan karakter utama yang akan digunakan Pemain dalam game. Karakter memiliki damage 20 dan health point 100. Karakter ini dibuat dengan Pixilart.

3.5. Perancangan Karakter Enemy

Musuh adalah karakter yang akan digunakan menghalangi pemain didalam game Knight The Devil Slayer. Musuh terbagi menjadi empat yaitu Slime, Skeleton, Golem dan Devil. Desain Musuh dapat dilihat pada Gambar dibawah ini.



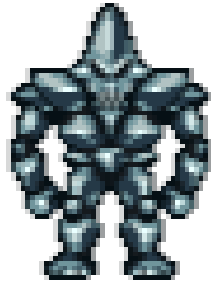
Gambar 6. Desain musuh Slime

Pada gambar 6. merupakan Desain musuh Slime yang akan menghalangi dan menyerang pemain pada level 2. Slime memiliki damage 5 dan health point 25. Desain Slime dibuat dengan Pixilart.



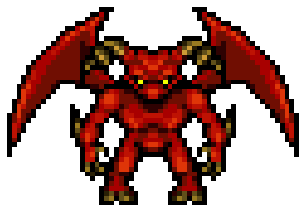
Gambar 7. Desain musuh Skeleton

Pada gambar 7. merupakan Desain musuh Skeleton yang akan menghalangi dan menyerang pemain pada level 3 dan 4. Skeleton memiliki damage 10 dan health point 50. Desain Skeleton dibuat dengan Pixilart.



Gambar 8. Desain musuh Golem

Pada gambar 8. merupakan Desain musuh Golem yang akan menghalangi dan menyerang pemain pada level 4. Golem memiliki damage 20 dan health point 100. Desain Golem dibuat dengan Pixilart



Gambar 9. Desain musuh Devil

Pada gambar 9. merupakan Desain musuh Devil yang akan menghalangi dan menyerang pemain pada level 5. Devil memiliki damage 25 dan health point 150. Desain Devil dibuat dengan Pixilart

4. HASIL DAN PEMBAHASAN

Hasil dan pembahasan pada penelitian ini menjelaskan mengenai tampilan dari game Knight The Devil Slayer yang sudah melewati hasil pengujian metode, fungsional serta responden.

4.1. Menu Utama

Menu utama adalah scene awal ketika game Knight The Devil Slayer dijalankan. Pada menu utama terdapat pilihan menu Play Game, Pengaturan, Cara Bermain, dan Exit Game. Desain pada Menu Utama ini dibuat dengan



Gambar 10. Tampilan Menu utama

4.2. Tampilan Level 1

Tampilan level 1 adalah tampilan awal level dalam permainan “Knight The Devil Slayer”. Pada level 1 berlatar di desa kecil. Desain pada level 1 ini dibuat dengan menambahkan tileset ke dalam Godot Game Engine.



Gambar 11. Tampilan level 1

4.3. Tampilan Level 2

Tampilan level 2 adalah tampilan level kedua dalam permainan “Knight The Devil Slayer”. Pada level 2 berlatar di area padang rumput. Desain pada level 2 ini dibuat dengan menambahkan tileset ke dalam Godot Game Engine.



Gambar 12. Tampilan level 2

4.4. Tampilan Level 3

Tampilan level 3 adalah tampilan level ketiga dalam permainan “Knight The Devil Slayer”. Pada level 3 berlatar di area hutan. Desain pada level 3 ini dibuat dengan menambahkan tileset ke dalam Godot Game Engine.



Gambar 13. Tampilan level 3

4.5. Tampilan Level 4

Tampilan level 4 adalah tampilan level keempat dalam permainan “Knight The Devil Slayer”. Pada level 4 berlatar di area lembah. Desain pada level 4 ini dibuat dengan menambahkan tileset ke dalam Godot Game Engine.



Gambar 14. Tampilan level 3

4.6. Tampilan Level 5

Tampilan level 5 adalah tampilan level kelima dalam permainan “Knight The Devil Slayer”. Pada level 5 berlatar di area dungeon. Desain pada level 5 ini dibuat dengan menambahkan tileset ke dalam Godot Game Engine.



Gambar 15. Tampilan level 3

4.7. Pengujian Beta

Pada titik ini, permainan yang telah dibuat akan diuji untuk memastikan bahwa itu beroperasi dengan baik dan bahwa hasilnya sesuai dengan konsep dasar yang dibuat.

Tabel 1. Pengujian Main Menu

Kasus dan Hasil Uji			
Kasus	Yang diharapkan	Ya	Tidak
Tekan <i>Button Play Game</i>	Sistem melakukan proses untuk menampilkan level 1	✓	-
Tekan <i>Button Pengaturan</i>	Sistem melakukan proses untuk menampilkan Pengaturan game	✓	-
Tekan <i>Button Cara Bermain</i>	Sistem melakukan proses untuk menampilkan cara bermain	✓	-
Tekan <i>Button Exit Game</i>	Sistem melakukan proses untuk menutup game	✓	-

Tabel 2. Pengujian Pengaturan

Kasus dan Hasil Uji			
Kasus	Yang diharapkan	Ya	Tidak
<i>Master Music</i>	Jika volume music ,di geser ke kiri maka suara keseluruhan music berkurang	✓	-
<i>Audio Music</i>	Jika volume music,di geser ke kiri maka <i>background sound</i> berkurang	✓	-
<i>SFX Music</i>	Jika volume music,di geser ke kiri maka <i>sound effect</i> berkurang	✓	-

Tabel 3. Pengujian Pause Menu

Kasus dan Hasil Uji			
Kasus	Yang diharapkan	Ya	Tidak
<i>Button Resume</i>	Jika ditekan akan melanjutkan game	✓	-
<i>Button Pengaturan</i>	Sistem melakukan proses untuk menampilkan Pengaturan game	✓	-
<i>Button Main Menu</i>	Sistem melakukan proses untuk kembali ke main menu	✓	-
<i>Button Exit</i>	Sistem melakukan proses untuk menutup game	✓	-

Pada tabel 1, 2 dan 3 merupakan pengujian main menu, pengaturan dan pause menu menunjukkan bahwa semua tombol pada main menu berfungsi sesuai yang diharapkan pada setiap kasus berhasil tercapai. Pada pengujian pengaturan memperlihatkan bahwa pengaturan volume berfungsi dengan baik saat volume digeser ke kiri, yang menunjukkan penurunan suara secara keseluruhan, background sound, dan sound effect. Pada pengujian Pause Menu menunjukkan bahwa semua tombol, termasuk Resume, Pengaturan, Main Menu, dan Exit, berfungsi dengan baik saat ditekan selama permainan.

Tabel 4. Pengujian Pathfinding

Kasus dan Hasil Uji		
Kasus	Yang diharapkan	Hasil
Musuh pada <i>Level 2</i>	Musuh level 2 dapat menemukan jalur optimal	Berhasil, Musuh pada level 2 dapat menemukan jalur optimal untuk menemukan Pemain
Musuh pada <i>Level 3</i>	Musuh level 3 dapat menemukan jalur optimal	Berhasil, Musuh pada level 3 dapat menemukan jalur optimal untuk menemukan Pemain
Musuh pada <i>Level 4</i>	Musuh level 4 dapat menemukan jalur optimal	Berhasil, Musuh pada level 4 dapat menemukan jalur optimal untuk menemukan Pemain
Musuh pada <i>Level 5</i>	Musuh level 5 dapat menemukan jalur optimal	Berhasil, Musuh pada level 5 dapat menemukan jalur optimal untuk menemukan Pemain

Pada Tabel 4. merupakan Pengujian *Pathfinding*, *Pathfinding* menunjukkan bahwa musuh di berbagai level (2, 3, 4, dan 5) berhasil menemukan jalur optimal untuk menemukan pemain, sesuai dengan ekspektasi yang diharapkan.

Tabel 5. Pengujian Finite State Machine

Kasus dan Hasil Uji		
Kasus	Yang diharapkan	Hasil
<i>Finite State Machine</i> antara Musuh dengan Pemain	Musuh dapat memberikan damage, pada saat menyerang pemain.	Berhasil, Hp atau darah Pemain dapat berkurang pada saat musuh menyerang pemain.
<i>Finite State Machine</i> antara Pemain dengan Musuh	Pemain dapat memberikan damage, pada saat menyerang musuh.	Berhasil, Hp atau darah musuh dapat berkurang pada saat pemain menyerang musuh.

Pada Tabel 5. merupakan Pengujian *Finite State Machine*, pengujian *Finite State Machine* memperlihatkan bahwa mekanisme interaksi antara pemain dan musuh berfungsi dengan baik, di mana

pemain dapat memberikan damage pada musuh dan sebaliknya, musuh dapat memberikan damage pada pemain.

Tabel 6. Pengujian Perangkat

Merk HP	Prosesor	RAM	Hasil
Xiaomi	Helio G99	4 GB	Berhasil, game dapat berjalan dengan normal.
Realme	Snapdragon 695	8 GB	Berhasil, game dapat berjalan dengan normal.
Infinix	Snapdragon 732G	8 GB	Berhasil, game dapat berjalan dengan normal.

Pada Tabel 6. merupakan Pengujian terhadap 3 perangkat, Hasil pengujian perangkat pada Tabel 6. menunjukkan bahwa game dapat berjalan dengan normal pada tiga perangkat berbeda (Xiaomi, Realme, dan Infinix) dengan spesifikasi prosesor dan RAM yang beragam, menunjukkan kompatibilitas yang baik dari game tersebut.

Tabel 7. Pengujian Perangkat pada Game

No.	Pertanyaan	Penilaian	
		Ya	Tidak
1.	Apakah tombol Play Game di Menu utama berfungsi?	9	0
2.	Apakah tombol Pengaturan di Menu utama berfungsi?	9	0
3.	Apakah di Pengaturan Volume Bar dapat digeser ke kanan dan kiri?	9	0
4.	Apakah tombol Cara Bermain di Menu utama berfungsi?	9	0
5.	Apakah tombol Exit Game di Menu utama berfungsi?	9	0
6.	Apakah Karakter dapat digerakkan dengan Joystick?	9	0
7.	Apakah tombol Attack berfungsi dengan normal?	9	0
8.	Apakah <i>Pause Menu</i> berfungsi dengan normal pada level 1?	9	0
9.	Apakah pemain dapat berpindah ke level selanjutnya?	9	0
10.	Apakah Slime dapat mengejar dan memberikan damage ke pemain?	9	0
11.	Apakah Pemain dapat memberikan damage ketujuh musuh di level ini?	8	0
12.	Apakah Checkpoint Respawn berfungsi dengan normal?	9	0
13.	Apakah <i>Pause Menu</i> berfungsi dengan normal pada level 2?	9	0
14.	Apakah Pemain dapat menuju ke level selanjutnya?	9	0
15.	Apakah Skeleton dapat mengejar dan memberikan damage ke pemain?	9	0
16.	Apakah Pemain dapat memberikan damage ketujuh musuh di level ini?	9	0
17.	Apakah Checkpoint Respawn berfungsi dengan normal?	9	0
18.	Apakah <i>Pause Menu</i> berfungsi dengan normal pada level 3?	9	0
19.	Apakah Pemain dapat menuju ke level selanjutnya?	9	0
20.	Apakah Skeleton dapat mengejar dan memberikan damage ke pemain?	9	0
21.	Apakah Golem dapat mengejar dan memberikan damage ke pemain?	9	0
22.	Apakah Pemain dapat memberikan damage keenam musuh di level ini?	9	0
23.	Apakah Pemain dapat memberikan damage keenam musuh di level ini?	9	0
24.	Apakah <i>Pause Menu</i> Berfungsi dengan normal pada level 4?	9	0
25.	Apakah Pemain dapat menuju ke level selanjutnya?	9	0
26.	Apakah Devil dapat mengejar dan memberikan damage ke pemain?	9	0
27.	Apakah pemain dapat memberikan damage ke musuh di level ini?	9	0
28.	Apakah Checkpoint Respawn berfungsi dengan normal?	9	0
29.	Apakah <i>Pause Menu</i> Berfungsi dengan normal pada level 5?	9	0
Total		260	1
Presentase		99.62%	0.38%

Berdasarkan Tabel 7. merupakan Pengujian Responden, Dari 9 responder yang memainkan game Knight The Devil Slayer menunjukkan bahwa 99,62% dari responden menjawab Ya dan sebanyak 0,38% menjawab Tidak, dari 29 pertanyaan tentang fitur fitur yang berfungsi di game Knight The Devil Slayer.

5. KESIMPULAN DAN SARAN

Hasil pengujian yang dilakukan pada game Knight The Devil Slayer menunjukkan fitur-fitur pada game berjalan dengan sangat baik secara keseluruhan. Pengujian Pathfinding menunjukkan bahwa musuh di berbagai level berhasil menemukan jalur optimal untuk menemukan pemain, sesuai dengan ekspektasi yang diharapkan bahwa Karakter Non-Pemain memiliki tingkat keberhasilan yang sama sesuai dengan desain Pathfinding. Pengujian Finite State Machine memperlihatkan bahwa mekanisme interaksi antara pemain dan musuh berfungsi dengan baik, di mana pemain dapat memberikan damage pada musuh dan sebaliknya, musuh dapat memberikan damage pada pemain. Dari 9 responder yang memainkan game Knight The Devil Slayer menunjukkan bahwa 99,62% dari responden menjawab Ya dan sebanyak 0,38% menjawab Tidak, dari 29 pertanyaan tentang fitur fitur yang berfungsi di game Knight The Devil Slayer. Saran untuk penelitian selanjutnya adalah dapat dikembangkan dengan Theta* untuk menghasilkan jalur yang lebih optimal dengan lebih sedikit belokan. Serta dapat dikembangkan untuk perangkat mobile lain selain seperti IOS.

DAFTAR PUSTAKA

- [1] Wicaksono, B. A., Mahmudi, A., & Auliasari, K. (2023). Perancangan Game Alien Warfare 2D Menggunakan Metode Finite State Machine (FSM) Berbasis Android. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 7(5), 2945-2951
- [2] Rafiq, A.; Kadir, T.A.; Ihsan, S.N. Pathfinding Algorithms in game development. In *IOP Conference Series: Materials Science and Engineering*; IOP Publishing: Pahang, Malaysia, 2020; Volume 769, p. 012021
- [3] Russell, S. J., & Norvig, P. (2021). *Artificial Intelligence: A Modern Approach*. Pearson.
- [4] Nguyen, T., and Lee, J. "Real-Time Pathfinding for Non-Player Characters in Open-World Games." *Journal of Computer Graphics and Game Development*, vol. 15, no. 1, 2022, pp. 34-50.
- [5] San, E. R. C., & Handriyanti, E. (2022). Penerapan Metode Pathfinding Pada Pengembangan Game "The Book of Aksara" Pada Perangkat Bergerak. *Prosiding SISFOTEK*, 6(1), 81-85
- [6] Prasetyo, T. A., Chandra, R., Simamora, B., Christian, M. J., Silaban, A. R., & Siregar, M. V. (2023). Pathfinding Solving in Maze Game using Backtracking Algorithms. *Jurnal CoreIT*, 9(1)
- [7] Harsadi, P., Asmiatun, S., & Putri, A. N. (2021). Dynamic Pathfinding for Non-Pemain Character Follower on Game. *Jurnal Teknik Informatika CIT Medicom*, 13(2), 55-63
- [8] Mutaqin, G., Fadilah, J. N., & Nugroho, F. (2021). Implementasi Metode Path Finding dengan Penerapan Algoritma A-Star untuk Mencari Jalur Terpendek pada Game "Jumrah Launch Story". *Walisongo Journal of Information Technology*, 3(1), 43-48
- [9] HPratama, R. R., & Surahman, A. (2020). Perancangan aplikasi game fighting 2 dimensi dengan tema karakter Nusantara berbasis Android menggunakan Construct 2. *Jurnal Informatika dan Rekayasa Perangkat Lunak*, 6(2), 45-52
- [10] Ranaweera, Mahesh, & Qusay H. Mahmoud. (2024). "Deep Reinforcement Learning with Godot Game Engine" *Electronics* 13, no. 5: 985. <https://doi.org/10.3390/electronics13050985>
- [11] Romadhon, A. S., Widyaningrum, V. T., & Dafid, A. (2023). Implementasi Algoritma A* (A-Star) dalam Pergerakan Objek Robot. *Technium*, 16(1), 280-285
- [12] Fernando, A., Guntoro, G., Costaner, L., Devega, M., & Lisnawita, L. (2023). Penerapan Metode Finite State Machine Pada Game Pembelajaran Matematika. *Jurnal Karya Ilmiah Multidisiplin (JURKIM)*, 3(1), 60-68.