



Penerapan Algoritma A-Star Dalam Mencari Jalur Tercepat dan Pergerakan NonPlayer Character Pada Game Petualangan Labirin Tech-Edu

Bagus Tegar Dwi Irianto*, Septi Andryana, Aris Gunaryati

Fakultas Teknologi Komunikasi dan Informatika, Prodi Informatika, Universitas Nasional, Jakarta Selatan, Indonesia

Email: ¹bagus1698@gmail.com, ²septi.andryana@civitas.unas.ac.id, ³arisgunaryati@civitas.unas.ac.id

Email Penulis Korespondensi: bagus1698@gmail.com

Abstrak—Permainan merupakan sarana hiburan yang berada dalam media elektronik seperti smartphone yang dibuat semenarik mungkin sehingga pemain mendapat kepuasan batin. Perkembangan pada industri permainan berkembang cukup cepat dari mobile maupun desktop. Kategori pada permainan cukup banyak seperti aksi, strategi, olahraga, shooter, petualangan dan simulasi. Game petualangan merupakan salah satu kategori permainan yang pemainnya dilatih untuk berfikir agar bisa menyelesaikan permainan tersebut. Banyak hal yang perlu diperhatikan seperti visual pada peta ataupun interaksi pada NPC (*NonPlayer Character*), dengan memanfaatkan teknologi sekarang pembelajaran dapat disampaikan melalui sebuah permainan. Karakteristik pada permainan yang menghibur dan kegiatan visual yang menarik akan membuat pembelajaran semakin diminati. Pembuatan permainan tentunya akan menggunakan algoritma sebagai fitur ataupun sistem dalam pembuatan permainan. Algoritma a-star salah satu yang banyak digunakan dalam pembuatan permainan labirin. Algoritma a-star dapat diterapkan pada bantuan jalur tercepat ataupun pergerakan pada NPC. Tujuan dari penelitian membuat permainan edukasi labirin dengan menggunakan algoritma a-star sebagai pencarian jalur tercepat dan pergerakan NPC. Hasil pengujian pada permainan pada tujuh perangkat mendapatkan hasil berhasil dijalankan, dalam penggunaan CPU pada perangkat mobile minimal 7% dan maksimal 57%. Penggunaan CPU pada perangkat desktop minimal sebesar 2% dan maksimal 33%.

Kata Kunci: Algoritma A-Star; Jalur Tercepat; Labirin; NonPlayer Character; Permainan Petualangan

Abstract—Games are a means of entertainment that are in electronic media such as smartphones that are made as attractive as possible so that players get inner satisfaction. The development of the game industry is growing quite fast from mobile to desktop. There are many categories of games such as action, strategy, sports, shooter, adventure and simulation. Adventure games are one category of games where players are trained to think in order to complete the game. There are many things that need to be considered, such as visuals on maps or interactions with NPCs (*NonPlayer Character*), by utilizing technology now learning can be delivered through a game. The characteristics of entertaining games and interesting visual activities will make learning more attractive. Making games will of course use algorithms as features or systems in game making. The a-star algorithm is one that is widely used in making maze games. The a-star algorithm can be applied to the help of the fastest path or movement of the NPC. The purpose of the research is to make a maze educational game using the a-star algorithm as the fastest path search and NPC movement. The test results on games on seven devices got the results that they were successfully run, in CPU usage on mobile devices a minimum of 7% and a maximum of 57%. CPU usage on desktop devices is a minimum of 2% and a maximum of 33%.

Keywords: A-Star Algorithm; Adventure Game; Fastest Path; Maze; NonPlayer Character

1. PENDAHULUAN

Permainan atau *game* adalah sarana hiburan yang berada dalam media elektronik seperti smartphone yang dibuat semenarik mungkin sehingga pemain mendapat kepuasan batin [1]. *Game* juga bisa digunakan untuk mengisi waktu luang dan melepas penat [2]. Perkembangan dalam industri game sangat capat pada semua platform seperti mobile, desktop ataupun dalam konsol. Banyak berbagai jenis kategori dalam permainan dari aksi, strategi, olahraga, shooter, petualangan dan simulasi [3]. Seiring dengan perkembangan teknologi saat ini *game* tidak hanya digunakan sebagai media hiburan saja, namun juga bisa digunakan sebagai media pembelajaran. Karakteristik dari permainan yang menghibur dan kegiatan visual yang menarik saat dimainkan membuat pembelajaran yang menarik [4]. *Game* petualangan menggunakan kemampuan dalam berfikir untuk menganalisis visual, urutan peristiwa dan interaksi pada NPC (*NonPlayer Character*) [5]. Salah satu contoh permainan petualangan seperti permainan labirin, dimana pemain harus mencari jalan keluar dalam suatu lokasi agar bisa menyelesaikan permainan tersebut [6]. Pemain harus berfikir seperti prinsip permainan petualangan.

Dalam pembuatan permainan pastinya membutuhkan algoritma untuk digunakan seperti alur permainan, bantuan, fitur dalam permainan ataupun sistem dalam permainan [7]. Algoritma a-star salah satu yang banyak digunakan dalam pembuatan permainan labirin. Permainan ini dapat menentukan jalur tercepat yang bisa digunakan dalam permainan, algoritma a-star akan mencari jalur tercepat dari titik awal ke titik akhir walaupun terdapat penghalang dalam rute tersebut [8].

Penyusunan penelitian ini, penulis mencari referensi melalui jurnal-jurnal yang membahas mengenai penerapan algoritma a-star ataupun edukasi pada dalam sebuah *game*, seperti penelitian yang dilakukan oleh Oktanugraha dan Nudin yang membahas tentang implementasi algoritma a-star dalam menentukan rute terpendek yang dilewati oleh NPC (*NonPlayer Character*) pada game Good Thief. Algoritma a-star diterapkan pada NPC untuk membuat tingkat kesulitan dari yang mudah hingga sulit, sehingga pada hasil pengujian 6 responden mendapatkan hasil yang cukup baik yaitu sebesar 83,22% [1].



Pada penelitian kedua yang dilakukan oleh Ahmad dan Widodo mengenai penerapan algoritma a-star pada game petualangan labirin berbasis android. Pada pembangunan game labirin ini, AI (*Artificial Intelligence*) menggunakan algoritma a-star dengan *euclidean distance*. Sehingga hasilnya saat pemain kesulitan dalam menemukan jalur untuk menuju makanan kelinci, mereka bisa menggunakan tombol bantuan untuk mencari jalur terpendek untuk menuju lokasi makanan [5].

Pada penelitian berikutnya yang dilakukan oleh Nugraha, Nur Budi mengenai perancangan game labirin dengan algoritma a-star berbasis *mobile*. Algoritma a-star dalam penelitian ini digunakan sebagai tombol bantuan dalam pencarian rute terpendek pada *stage 1*, dimana *game* ini memiliki 3 *stage* yang memiliki tingkat kesulitan yang berbeda-beda [9].

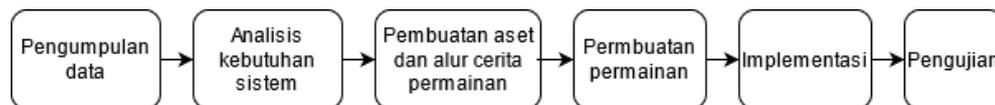
Penelitian selanjutnya dilakukan oleh Athariq dan Putra yang membahas penentuan pergerakan NPC menggunakan algoritma a-star pada *game action role playing game*. Penelitian ini bertujuan melakukan percobaan terhadap implementasi algoritma a-star dalam suatu permainan. Hasilnya NPC mampu berjalan menuju pemain sesuai dengan perhitungan algoritma a-star [10].

Berdasarkan penelitian tersebut penulis membuat permainan labirin dengan menggunakan algoritma a-star. Algoritma tersebut akan diterapkan kedalam permainan nantinya seperti sebagai pergerakan NPC menuju pemain, Pergerakan Pemain menuju NPC dan menentukan jalur tercepat untuk menuju titik akhir. Penulis juga akan menambahkan materi pembelajaran mengenai teknologi yang dapat digunakan dalam bidang pendidikan. Dalam permainan tersebut pemain akan berinteraksi dengan NPC untuk bisa menyelesaikan permainannya.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini menggunakan tahapan-tahapan dalam proses pembuatan permainan. Urutan pada setiap tahapnya dapat dilihat pada gambar 1.



Gambar 1. Alur Penelitian

Pada gambar 1 merupakan urutan yang dilakukan dalam proses pembuatan permainan edukasi, berikut adalah tahapannya:

a. Pengumpulan Data

Pengumpulan data dilakukan dengan mencari jurnal terkait mengenai permainan sebagai sumber penelitian. Jurnal yang dikumpulkan terkait tentang penggunaan algoritma a-star pada permainan yang diterapkan sebagai pencarian jalur terpendek ataupun pergerakan NPC.

b. Analisis Kebutuhan Sistem

Kebutuhan sistem dalam pembuatan permainan edukasi ini yaitu [11]:

1. Perangkat Keras (Hardware)

Laptop7 Lenovo Ideapad L340 dengan spesifikasi :

- Ryzen 5 3500U
- AMD Radeon RX Vega 8 iGPU
- RAM 12GB DDR4
- HDD 1TB+500GB

2. Perangkat Lunak (Software)

Perangkat lunak yang digunakan dalam pembuatan permainan yaitu :

- Windows 10 64 bit
- RPG Maker MV 1.6.1
- Android Studio 3.4

c. Pembuatan Aset dan Alur Cerita Permainan

Tahap ini penulis mempersiapkan aset yang digunakan seperti gambar, suara, efek, dll. Alur cerita juga dibuat pada tahap ini, dengan membuat perencanaan dalam jumlah map, fitur apa saja yang digunakan, tahapan pada permainan dan penggunaan algoritmanya.

d. Pembuatan Permainan

Tahap ini merupakan proses pembuatan permainan dari membuat logika pada permainan, alur permainan, peta pada setiap *stage* dan memasukkan aset yang telah dibuat. Pembuatan permainan dilakukan dengan RPG maker MV dan android studio.

e. Implementasi



Tahap ini mengimplementasikan algoritma yang digunakan pada permainan. Algoritma yang digunakan merupakan algoritma a-star yang dapat mencari jalur tercepat menuju titik tujuan. Algoritma diterapkan pada jalur bantuan pada *stage*, pergerakan NPC, dan pergerakan pemain menuju NPC.

f. Pengujian

Tahapan ini merupakan pengujian permainan pada beberapa perangkat untuk melihat performa permainan yang telah dibuat. Performa permainan dapat dilihat dari penggunaan CPU (*Central Processing Unit*) ataupun penggunaan memori [12].

2.2 Algoritma A-star

Algoritma A-star merupakan algoritma *best first search* dengan menggunakan nilai heuristik untuk menentukan urutannya [13]. Algoritma akan meminimalkan total biaya lintasan, dan pada kondisi yang tertentu akan memberikan solusi yang terbaik dengan waktu yang optimal [14]. Adapun rumus yang digunakan pada Algoritma a-star sebagai berikut:

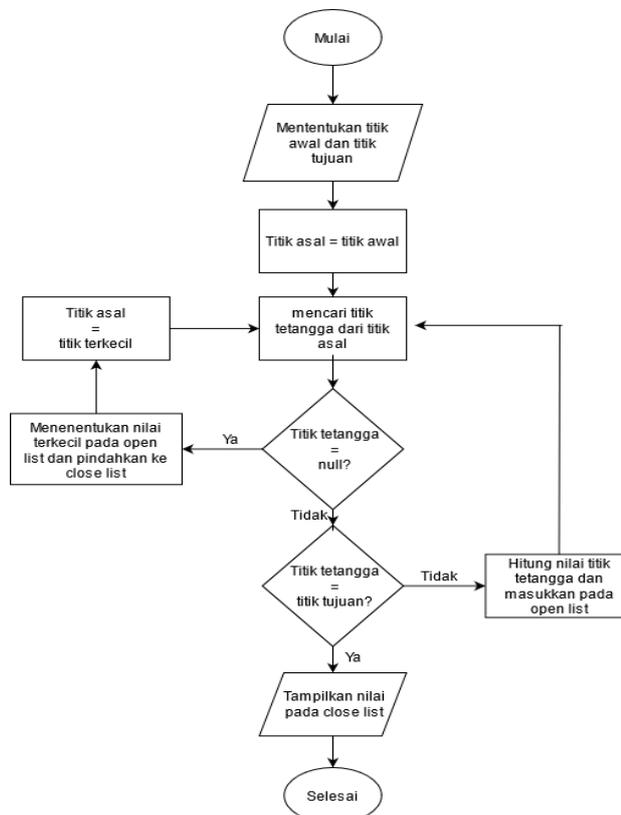
$$H(n) = \sqrt{(x1 - x2)^2 + (y1 - y2)^2} \tag{1}$$

Rumus 1 merupakan cara perhitungan untuk mendapatkan nilai heuristik. Nilai pada koordinat awal dimasukkan dalam nilai X1 dan Y1, lalu nilai pada koordinat tujuan dimasukkan pada nilai X2 dan Y2 [15].

$$F(n) = G(n) + H(n) \tag{2}$$

F(n) merupakan fungsi biaya yang dibutuhkan untuk berpindah dari titik tersebut ke titik tujuan. Biaya yang ditempuh dari node asal dimasukkan kedalam G(n) dan nilai heuristik yang telah dihitung dimasukkan kedalam H(n) [16].

Nilai pada G Cost vertikal horizontal dibedakan dengan node diagonal. Nilai H Cost dihitung sesuai jarak node ke node tujuan dengan penjumlahan vertikal maupun horizontal dan nilai F Cost didapat sesuai dengan rumus, penjumlahan antara G Cost dan H Cost [17].



Gambar 2. Flowchart Algoritma A-star

Pada gambar 2 merupakan flowchart atau alur dari algoritma a-star, berikut tahapan yang dilakukan pada algoritma a-star:

1. Menentukan titik awal dan titik tujuan pada peta.
2. Mengubah titik asal menjadi titik awal.



3. Mencari titik tetangga atau titik yang dapat dilalui dari titik asal.
4. Melakukan cek apakah titik tetangga kosong atau tidak.
5. Jika titik tetangga kosong menentukan nilai terkecil pada *open list* dan memindahkannya pada *close list* lalu mengubah titik asal menjadi titik terkecil.
6. Melakukan cek apakah titik tetangga merupakan titik tujuan atau tidak.
7. Apabila titik tetangga bukan titik tujuan maka akan melakukan hitungan pada titik tetangga tersebut dengan rumus algoritma a-star lalu dimasukkan kedalam *open list*.
8. Jika titik tetangga merupakan tujuan, maka nilai pada *close list* ditampilkan. Nilai pada *close list* merupakan jalur tercepat yang didapatkan [18].

3. HASIL DAN PEMBAHASAN

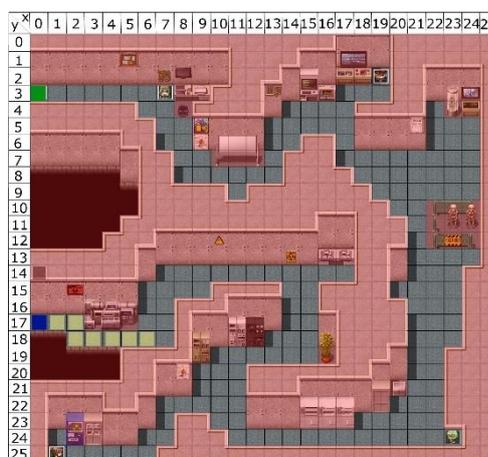
3.1 Implementasi Algoritma a-star

Penerapan algoritma pada permainan ini terdapat pada pergerakan NPC menuju pemain, menuju NPC ketika pemain tersesat dan juga pencarian jalur tercepat untuk menuju akhir *stage*.



Gambar 3. Contoh Map Untuk Perhitungan Algoritma A-star

Pada gambar 3 merupakan contoh map pada *stage* kedua dengan luas map 25x25. Kotak berwarna biru pada titik 0,17 merupakan posisi pemain saat masuk *stage* kedua, kotak berwarna hijau titik 0,3 merupakan posisi akhir untuk menuju *stage* selanjutnya dan kotak berwarna merah merupakan jalur yang tidak bisa dilewati oleh pemain. dengan algoritma a-star penulis dapat menghitung mana jalur tercepat pemain untuk menyelesaikan *stage* ini, namun pemain harus menuju NPC yang ada baru bisa menuju *stage* selanjutnya. Jadi bantuan jalur akan menuju ke NPC satu ke yang lainnya.



Gambar 4. Percabangan pada Titik 6,18

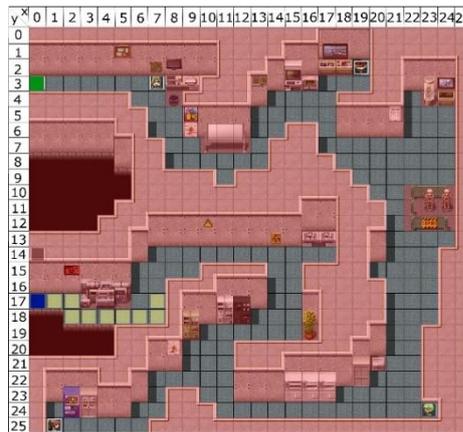
Pertama kita lihat kotak vertikal dan diagonal pada posisi pemain, jika hanya terdapat satu jalur yang di lewati maka itu adalah pilihan jalurnya, contohnya terlihat pada gambar 4. Hingga akhirnya berhenti pada titik 6,18. Posisi tujuan awal kita yaitu pada titik 23,24. Hasil perhitungan dengan algoritma a-star untuk diagonal dan vertikal pada posisi 6,18 dapat dilihat pada tabel 1.



Tabel 1. Perhitungan Algoritma a-star Pada Titik 6,18

node (6,17)	node (7,18)
$F(n) = G(n) + H(n)$	$F(n) = G(n) + H(n)$
$F(6,17) = 1 + \sqrt{(23 - 6)^2 + (24 - 17)^2}$	$F(7,18) = 1 + \sqrt{(23 - 7)^2 + (24 - 18)^2}$
$F(6,17) = 19,3$	$F(7,18) = 18$

Pada tabel 1 terlihat nilai F pada titik yang dapat dilewati dari titik 6,18 yaitu 6,17 dan 7,18. Hasil perhitungan jalur tercepat antara dua titik tersebut dapat dilihat dari nilai F terkecil yaitu pada titik 7,18. Maka titik itulah yang akan dirokemendasikan untuk pemain lewati.



Gambar 5. Percabangan pada Titik 7,17

Hasil dari perhitungan pada tabel 1 dapat dilihat pada gambar 5. Karena pada titik 7,18 hanya ada 1 jalur maka jalur itulah yang dipilih dan berhenti pada titik 7,17 karena memiliki dua percabangan titik yaitu 6,17 dan 7,16. Disini dilakukan lagi perhitungan dengan algoritma a-star.

Tabel 2. Perhitungan Algoritma a-star Pada Titik 7,17

node (6,17)	node (7,16)
$F(n) = G(n) + H(n)$	$F(n) = G(n) + H(n)$
$F(6,17) = 1 + \sqrt{(23 - 6)^2 + (24 - 17)^2}$	$F(7,18) = 1 + \sqrt{(23 - 7)^2 + (24 - 16)^2}$
$F(6,17) = 19,3$	$F(7,18) = 18,8$

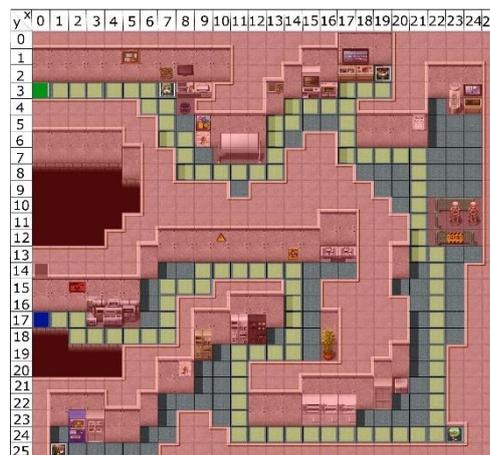
Tabel 3. Perhitungan Algoritma a-star Pada Titik 7,16

node (6,16)	node (7,15)
$F(n) = G(n) + H(n)$	$F(n) = G(n) + H(n)$
$F(6,17) = 1 + \sqrt{(23 - 6)^2 + (24 - 16)^2}$	$F(7,18) = 1 + \sqrt{(23 - 7)^2 + (24 - 15)^2}$
$F(6,17) = 19,7$	$F(7,18) = 19,3$

Tabel 4. Perhitungan Algoritma a-star Pada Titik 7,15

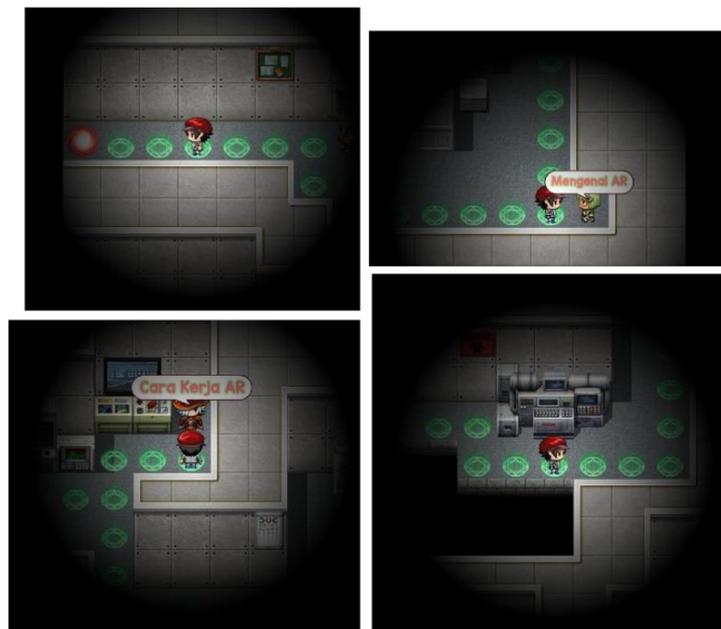
node (6,15)	node (7,14)
$F(n) = G(n) + H(n)$	$F(n) = G(n) + H(n)$
$F(6,17) = 1 + \sqrt{(23 - 6)^2 + (24 - 15)^2}$	$F(7,18) = 1 + \sqrt{(23 - 7)^2 + (24 - 14)^2}$
$F(6,17) = 20,2$	$F(7,18) = 19,8$
node (8,15)	
$F(n) = G(n) + H(n)$	
$F(6,17) = 1 + \sqrt{(23 - 8)^2 + (24 - 15)^2}$	
$F(6,17) = 18,4$	

Hasil dari perhitungan pada titik 7,17 dapat dilihat pada tabel 2. Nilai F terkecil yaitu pada titik 7,16 dan dilanjutkan perhitungan pada 2 titik yang dapat dilalui dari titik tersebut. Pada tabel 3 terlihat nilai F terkecil berada pada titik 7,15. Pada titik 7,15 ditemukan tiga percabangan yang dapat dilalui yaitu titik 6,15, 7,14 dan 8,15. Hasil perhitungan pada ketiga titik tersebut dapat dilihat pada tabel 3, dengan nilai F terkecil 8,15. Dengan perhitungan algoritma a-star maka dapat diketahui jalur tercepat untuk menyelesaikan stage kedua. Proses tersebut akan terus diulang hingga menuju sampai tujuan akhir.



Gambar 6. Jalur Hasil Perhitungan pad Map Stage Kedua

Gambar 6 merupakan hasil jalur tercepat berdasarkan hasil perhitungan algoritma a-star pada *stage* kedua. Kotak kuning menandakan jalur yang dilalui untuk jalur tercepat hingga titik akhir. Jalur tersebut menuju ke NPC yang harus ditemui oleh pemain. terdapat 3 buah NPC dan semuanya dilewati oleh jalur tersebut.



Gambar 7. Hasil Implementasi Jalur Bantuan Dalam Permainan

Pada gambar 8 merupakan hasil jalur bantuan didalam permainan. Hologram berwarna hijau merupakan petunjuk arah menuju NPC lainnya hingga berakhir pada titik akhir. Jarak pandang setiap *stage* dibatasi, hal ini dibuat agar membuat permainan menjadi lebih menarik dah tidak mudah untuk diselesaikan. Hasil dari perhitungan algoritma a-star dengan bantuan jalur tercepat dipastikan sesuai.

Tabel 5. Hasil Pengujian Algoritma a-star Pada Permainan

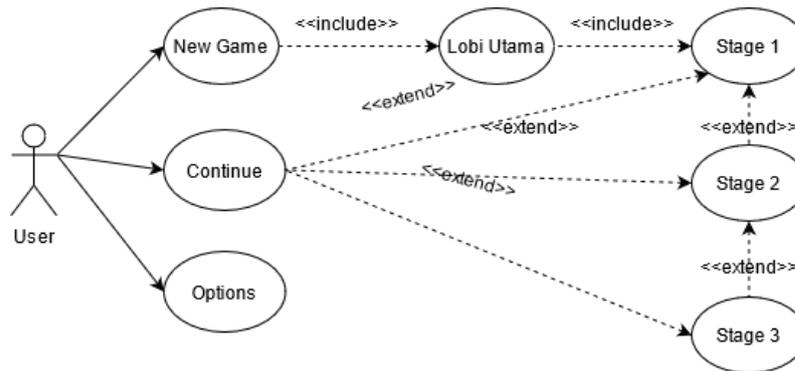
Perhitungan Algoritma A-star	Hasil Pada Permainan
Pergerakan otomatis dari pemain menuju NPC	Hasil sesuai dengan algoritma a-star dan berhasil di implementasikan
Pergerakan NPC bantuan menuju pemain	Hasil sesuai dengan algoritma a-star dan berhasil di implementasikan
Jalur bantuan menuju NPC yang belum ditemui	Hasil sesuai dengan algoritma a-star dan berhasil di implementasikan
Jalur bantuan menuju titik akhir	Hasil sesuai dengan algoritma a-star dan berhasil di implementasikan



Tabel 5 merupakan hasil dari implementasi algoritma a-star pada permainan. Pada setiap perhitungannya sesuai dengan hasil implementasi dalam permainan. Dengan ini algoritma a-star dapat digunakan dalam pencarian jalur ataupun pergerakan NPC dalam permainan.

3.2 Diagram Use Case

Diagram *use case* merupakan perancangan antara pemain dengan sistem pada permainan. Pada gambar 8 menjelaskan bagaimana interaksi antara pemain dengan permainan. Pertama permainan akan menampilkan menu utama yang terdapat pilihan *new game*, *continue* dan *options*. *New game* akan mengarahkan pemain masuk kedalam permainan yang menuju lobi utama, pada menu *continue* pemain dapat melanjutkan permainan yang telah disimpan pada saat bermain. Menu *options* pemain dapat mengatur volue suara dan juga beberapa *gameplay* saat bermain.



Gambar 8. Diagram Use Case pada Permainan

3.3 Tampilan Permainan

Hasil dari penelitian ini, penulis membuat permainan edukasi dengan RPG labirin untuk membuat pembelajaran lebih menarik. Permainan ini menjelaskan tentang teknologi yang dapat digunakan dalam dunia pendidikan seperti AR (*Augmented Reality*), VR (*Virtual Reality*) dan *Blended Learning*.



Gambar 9. Menu utama permainan

Pada gambar 9 merupakan tampilan pada menu awal permainan. Terdapat 3 pilihan pada menu utama ini yaitu *new game*, *continue* dan *options*. Untuk memulai permainan baru atau dari awal, pemain dapat memilih *new game* untuk memulainya.



Gambar 10. Map Lobi Utama dalam Permainan



Gambar 10 *map* lobi utama pada permainan. Pada *map* ini berisi NPC yang akan penjelasan isi dari permainan ini dan petunjuk arah kemana agar bisa memulai permainan dengan menuju ke *stage* pertama. Hal ini untuk memudahkan pemain agar tahu arah tentang permainan ini.



Gambar 11. Map stage pertama

Pada gambar 11 merupakan *map* labirin pada *stage* pertama. Pada *stage* ini pemain akan mempelajari tentang VR. Terdapat beberapa NPC pada *stage* ini yang harus berinteraksi pada pemain agar bisa lanjut ke dalam *stage* selanjutnya. NPC pada *stage* ini yaitu, pengenalan VR yang menjelaskan pengertian VR, cara kerja VR yang menjelaskan tentang bagaimana menggunakan VR dan cara kerjanya, dan juga NPC edukasi dengan VR yang akan menjelaskan bagaimana VR jika digunakan dalam dunia pendidikan.



Gambar 12. Map stage kedua dan ketiga

Map pada *stage* kedua dan ketiga dapat dilihat pada gambar 12. Awal pada saat masuk *map* kedua, pemain akan dapat pertanyaan saat pertama kali masuk pada *stage* kedua ini. Jika pemain dapat menjawab dengan benar maka akan muncul bantuan jalan menuju NPC hingga menuju akhir portal. NPC pada *stage* ini menjelaskan mengenai tentang penjelasan mengenai AR, cara kerja AR, dan contoh pada edukasinya. *Stage* ketiga sama seperti *stage* kedua, dimana pemain akan mendapat bantuan jalur saat berhasil menjawab pertanyaan. *Stage* ketiga ini mempelajari mengenai *blended learning*, seperti penjelasan tentang *blended learning*, kelebihan *blended learning* dan contoh *blended learning* pada dunia pendidikan.

3.4 Hasil Pengujian Permainan

Pengujian yang dilakukan merupakan uji performa permainan dalam beberapa perangkat. Spesifikasi perangkat yang digunakan masing-masing memiliki perbedaan, hal ini dilakukan agar terlihat performa dari perangkat yang memiliki beda spesifikasi. Pengujian pada perangkat *smartphone* akan menunjukkan penggunaan pada CPU, RAM (*Random Access Memory*) dan baterai, sementara pada perangkat desktop terlihat dari penggunaan CPU, GPU (*Graphic Processing Unit*) dan RAM.

Tabel 6. Hasil Pengujian Permainan Pada Smartphone Android

Model perangkat	Spesifikasi	Minimal penggunaan CPU	Maksimal penggunaan CPU	Maksimal Penggunaan RAM	Penggunaan Baterai
Asus Zenfone 3	<ul style="list-style-type: none"> • Snapdragon 625 • RAM 4GB LPDDR3 • Android 8.0 Oreo 	10%	27%	276 MB	Medium



Model perangkat	Spesifikasi	Minimal penggunaan CPU	Maksimal penggunaan CPU	Maksimal Penggunaan RAM	Penggunaan Baterai
Oppo F19 Pro	<ul style="list-style-type: none"> • Baterai 3000 mAh • Snapdragon 720G • RAM 8GB LPDDR4X 	7%	15%	270 MB	Light
	<ul style="list-style-type: none"> • Android 11 Red Velvet Cake • Baterai 4310 mAh 				
Oppo F1S	<ul style="list-style-type: none"> • MediaTek MT6750 • RAM 3GB LPDDR3 • Android 5.1 Lollipop 	15%	57%	181 MB	-
	<ul style="list-style-type: none"> • Baterai 3075 mAh 				
Samsung Galaxy M30	<ul style="list-style-type: none"> • Samsung Exynos 7 octa (7904) • RAM 4GB LPDDR3 • Android 10 Queen Cake 	15%	26%	157 MB	Light
	<ul style="list-style-type: none"> • Baterai 5000 mAh 				
Oppo F7	<ul style="list-style-type: none"> • Mediatek MT6771 Helio P60 • RAM 4GB LPDDR3 • Android 10 Queen Cake 	12%	28%	196 MB	Light
	<ul style="list-style-type: none"> • Baterai 3400 mAh 				

Tabel 7. Hasil Pengujian Permainan Pada Perangkat Desktop

Model perangkat	Spesifikasi	Penggunaan CPU	Penggunaan GPU	Penggunaan RAM
Lenovo L340	<ul style="list-style-type: none"> • CPU AMD Ryzen 5 3500U • GPU Radeon VEGA 8 Graphics 	2% - 10%	30%	87 MB
	<ul style="list-style-type: none"> • RAM 12 GB DDR4 • Windows 10 			
Lenovo G40-45	<ul style="list-style-type: none"> • CPU AMD A8-6410 • GPU AMD Radeon R5 Graphics 	24% - 33%	21%	168 MB
	<ul style="list-style-type: none"> • RAM 8 GB DDR3L • Windows 10 			

Pengujian dilakukan pada 7 perangkat yang terdiri dari 5 perangkat *mobile* dan 2 perangkat desktop. Hasil dari pengujian pada perangkat *mobile* terlihat pada tabel 6. Hasil yang didapat pada tabel 6 memiliki nilai yang berbeda, dari penggunaan CPU, RAM ataupun baterai. Terlihat bahwa pada perangkat Oppo F19 pro memiliki penggunaan CPU yang rendah dibandingkan dengan Oppo F1S, hal ini terjadi karena perbedaan CPU yang dimiliki antara kedua perangkat tersebut. Dimana Oppo F19 Pro memiliki CPU terbaru pada zaman sekarang yang tentunya memiliki performa lebih baik daripada CPU terdahulu. Pada tabel 7 merupakan hasil pengujian pada perangkat desktop, keduanya memiliki hasil yang cukup jauh dimana Lenovo L340 memiliki penggunaan CPU yang kecil walaupun penggunaan GPU yang besar, RAM yang dibutuhkan saat dimainkan pun cukup kecil. Dari semua perangkat yang telah dilakukan pengujian berhasil berjalan dengan baik, performa pada permainan berpengaruh terhadap spesifikasi perangkat yang digunakan. Hasil dari pengujian ini adalah permainan membutuhkan penggunaan CPU paling kecil 7 % pada perangkat mobile dan 2 % pada desktop, dengan maksimal penggunaan CPU pada perangkat mobile 57% dan 33% pada perangkat desktop.

4. KESIMPULAN

Permainan ini merupakan edukasi mengenai teknologi yang dapat digunakan pada bidang pendidikan. Permainan dengan kategori petualangan labirin ini dibuat agar penyampaian tentang materi akan lebih menarik. Pemain harus menjelajahi map dan berinteraksi dengan NPC untuk melanjutkan permainan. Algoritma a-star juga diterapkan dalam permainan ini. Dengan menggunakan rumus pada algoritma a-star sistem dapat membuat jalur menuju target tujuannya seperti dari player ke NPC, NPC menuju player dan juga jalur tercepat menuju titik akhir. Hasil pengujian algoritma a-star ini juga dibuat dengan hasil masing-masing berhasil mencapai tujuan akhirnya.



Pengujian dilakukan pada 7 perangkat yang terdiri dari 5 perangkat *mobile* dan 2 perangkat desktop. Hasil dari pengujian dalam permainan yaitu penggunaan CPU pada perangkat mobile minimal 7 % dan maksimal 57%, pada perangkat desktop penggunaan CPU minimal sebesar 2% dan maksimal 33%.

REFERENCES

- [1] R. F. Oktanugraha and S. R. Nudin, "Implementasi Algoritma A * (A Star) Dalam Penentuan Rute Terpendek Yang Dapat Dilalui Non Player Character Pada Game Good Thief," vol. 02, no. 01, pp. 74–85, 2020.
- [2] H. M. Solihin, A. N. Rachman, and C. M. S. R., "Penerapan Algoritma A * pada Aplikasi Game Space Shooter Berbasis Dekstop dengan Menggunakan Construct 2," *Sci. Artic. Informatics Students*, vol. 3, no. 1, pp. 60–64, 2020.
- [3] M. A. Irfan Faud, "Game Edukasi Memilih Sampah Berbasis Android Menggunakan Algoritma a-Star (a*)," *JIMP - J. Inform. Merdeka Pasuruan*, vol. 1, no. 3, pp. 1–13, 2016, doi: 10.37438/jimp.v1i3.32.
- [4] A. B. Febliama, N. Dian Fitria, and A. N. Handayani, "The Application of a Star (A*) Algorithm on the Android-Based Pacman Adaptation Educational Game as a Learning Media for SMK," *2nd Int. Conf. Vocat. Educ. Train. (ICOVET 2018)*, vol. 242, pp. 200–206, 2019, doi: 10.2991/icovet-18.2019.51.
- [5] I. Ahmad and W. Widodo, "Penerapan Algoritma A Star (A*) pada Game Petualangan Labirin Berbasis Android," *Khazanah Inform. J. Ilmu Komput. dan Inform.*, vol. 3, no. 2, p. 57, 2018, doi: 10.23917/khif.v3i2.5221.
- [6] F. Badri and M. F. Al Habib, "IMPLEMENTASI ALGORITMA A* (A Star) PADA NPC (NON-PLAYABLE CHARACTER) GAME PACMAN MENGGUNAKAN GAME ENGINE UNITY 5 BERBASIS ANDROID," *Tek. Eng. Sains J.*, vol. 4, no. 2, p. 49, 2020, doi: 10.51804/tesj.v4i2.295.49-56.
- [7] R. Wijayanti, W. Nugraha, and K. Kusriani, "Optimalisasi Penyelesaian Permainan pada Game Puzzle 8 dengan Perbandingan Algoritma A* dan Greedy," *Creat. Inf. Technol. J.*, vol. 7, no. 1, p. 10, 2021, doi: 10.24076/citec.2020v7i1.230.
- [8] D. Aditama, N. Fahriani, and P. A. Rakhma Devi, "Pencarian Jalur Terpendek Pada Permainan Pacman Menggunakan Algoritma A*," *J. Comput. Sci. Informatics Eng.*, vol. 2, no. 2, pp. 135–142, 2018, doi: 10.29303/jcosine.v2i2.129.
- [9] N. Budi Nugraha, "Rancang Bangun Game Labirin Menggunakan Algoritma A Star Berbasis Mobile," *J. Unitek*, vol. 11, no. 2, pp. 130–143, 2018.
- [10] Atthariq and D. A. Putra, "Penentuan Pergerakan Non-Player Character Menggunakan Algoritma a* Pada Game Action-Role-Playing Game," *J. Infomedia*, vol. 2, no. 2, pp. 35–40, 2018, doi: 10.30811/v2i2.516.
- [11] A. W. R. Ramadhan and D. Udjulawa, "Perbandingan Algoritma Dijkstra dan Algoritma A Star pada permainan Pac-Man," *J. Algoritm.*, vol. 1, no. 1, pp. 12–20, 2020, doi: 10.35957/algoritme.v1i1.411.
- [12] L. Safira, P. Harsadi, and S. Harjanto, "Penerapan Navmesh Dengan Algoritma A Star Pathfinding Pada Game Edukasi 3d Go Green," vol. 9, no. 1, 2021.
- [13] M. M. Attoyibi, F. E. Fikrisa, and A. N. Handayani, "The Implementation of A Star Algorithm (A*) In the Game Education About Numbers Introduction," *2nd Int. Conf. Vocat. Educ. Train. (ICOVET 2018)*, vol. 242, pp. 234–238, 2019.
- [14] E. Junanto, A. B. Osmond, and A. S. R. Ansori, "Membuat Pergerakan Non-Player Character (Npc) Menggunakan Metode A Star," *e-Proceeding Eng.*, vol. 7, no. 1, pp. 1491–1497, 2020.
- [15] D. S. J. Louis, P. S. Immanuel, and M. A. Angelia, "Game 'Finding Easter Eggs' Berbasis Augmented Reality Menggunakan Algoritma A-Star," *Cogito Smart J.*, vol. 7, no. 1, pp. 15–25, 2021.
- [16] F. Melati, A. P. Sujana, and S. Nurhayati, "Rancang Bangun Game Puzzle The Tabperi Dengan Algoritme A* Berbasis Android," *Komputika J. Sist. Komput.*, vol. 10, no. 1, pp. 11–18, 2021, doi: 10.34010/komputika.v10i1.3989.
- [17] E. F. Riski, A. Triayudi, and E. Mardiani, "Jurnal Mantik Jurnal Mantik," *Implement. A Star Algorithm AndroidBased Alien Shoot. Games*, vol. 4, no. 1, pp. 866–872, 2020.
- [18] L. Hermawan and M. B. Ismiati, "Random Obstacle Avoidance oleh Non Player Character Menggunakan Algoritma A*," *J. Transform.*, vol. 14, no. 2, p. 62, 2017, doi: 10.26623/transformatika.v14i2.438.