

# ***Game Platformer 2d Pembelajaran Matematika Sekolah Dasar Dengan Implementasi Algoritma A-Star Dalam Mencari Jalur Terdekat Pada Pergerakan Npc (Non-Player Character)***

Achmad Sa'ad Hamdi, Alfian Nurlifa, Imron Rosyidi, Zaebi Agustia Hidayatullah  
Program Studi Teknik Informatika, Universitas PGRI Ronggolawe Tuban

Correspondence Author: [lifa.nurlifa13@gmail.com](mailto:lifa.nurlifa13@gmail.com)

## **Abstract**

*Pada dasarnya game dibuat hanya untuk melepas penat dan sebagai sarana hiburan semata, namun seiring dengan perkembangan zaman, game tidak lagi menjadi ajang melepas penat, namun dapat dijadikan sebagai sarana edukatif dalam membantu proses belajar dan mengajar menjadi menyenangkan. Game yang saat ini masih ada dan dikembangkan adalah game aliran platformer, dimana pemain diharuskan untuk melewati berbagai macam rintangan yang ada untuk mencapai skor tertinggi. Tujuan dalam penelitian ini tidak lain yaitu membantu para guru untuk menjadikan game sebagai salah satu alternatif media pembelajaran matematika dan menerapkan Algoritma A\* pada NPC sehingga musuh dapat menggunakan rute terpendek dalam mencari Karakter. Proses pembuatan game ini menggunakan software Construct 2, sehingga output dari game ini adalah berbasis web.*

Keyword: *Game*, A Star, Construct 2, Edukasi, Media Pembelajaran.

## **1. PENDAHULUAN**

*Game* dalam bahasa Indonesia bisa diartikan sebagai permainan. Permainan merupakan aktivitas yang di dalamnya terdapat tata tertib tertentu sehingga mencapai suatu tujuan. Permainan adalah kegiatan yang melibatkan keputusan pemain, berusaha mencapai tujuan dengan dibatasi oleh aturan tertentu. (Clark, 2006). *Game* memiliki banyak jenis aliran, salah satunya adalah *Platformer*. *Platformer* bisa dikategorikan sebagai *genre* aksi di mana pengguna diharuskan menggerakkan *player* dari satu poin ke poin lain, berlari dan melompati platform gantung, lantai, tepian, tangga, atau rintangan lain yang ditampilkan pada layar permainan tunggal atau bergulir baik secara horizontal maupun vertical [2].

*Game* platform pertama dikembangkan pada awal 1980 an sehingga menjadikannya salah satu *genre game* video paling awal yang pernah ada. Banyak Sejarawan dan penggemar *game* menganggap *Space Panic* yang rilis pada tahun 1980 sebagai *game* platform pertama, sementara yang lain menganggap *Nintendo Donkey* yang rilis tahun 1981 sebagai yang pertama. Meskipun masih diperdebatkan *game* mana yang sebenarnya memulai *genre* platform, jelas bahwa *game* klasik awal seperti *Donkey Kong*, *Space Panic*, dan *Mario Bros* sangat berpengaruh dan semua memiliki andil dalam membentuk *genre* [4].

Sebuah *game* dapat terlihat lebih konkret dari segi cara perpindahan karakter musuh, maka diharapkan sebuah algoritma pencarian jalur terdekat yang akan membentuk karakter musuh melakukan perpindahan seperti makhluk hidup berpindah dari satu tempat ke tempat lainnya. Algoritma A\*(A-Star) merupakan salah satu kecerdasan buatan yang dipergunakan sebagai pencari rute terpendek dengan *cost* paling minimum yang tentunya akan menghasilkan NPC lebih cerdas untuk memilih jalurnya. Dengan memanfaatkan teknologi saat ini, penulis akan menerapkan Algoritma A\* untuk membuat karakter musuh yang dapat melakukan pencarian rute terpendek atau terdekat sambil bergerak menuju ke titik tujuan. Algoritma A\* (A-Star) sangat efisien diterapkan pada permainan yang terkait dengan pencari jalur atau *pathfinding*, dikarenakan Algoritma ini memiliki syarat di mana nilai heuristic tidak berlebihan juga tidak kurang dari semestinya [7].

Layaknya *game Super Mario*, peneliti mengajukan penelitian berupa pembuatan *game* bernama *Bejo Adventure* yang di dalamnya terdapat nilai edukasi berupa *Puzzle* dan diharapkan dapat menjadi salah satu media edukatif dan menyenangkan.

## **2. METODE PENELITIAN**

### **2.1 Analisis Sistem**

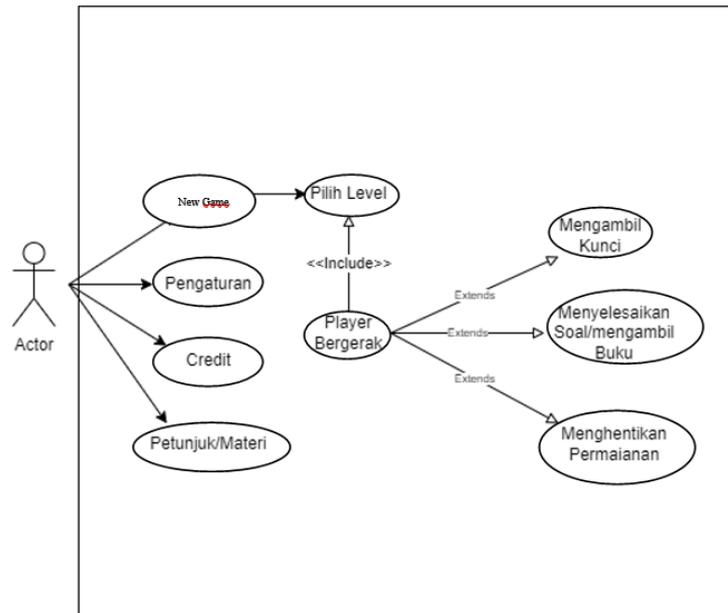
*Game Platformer Bejo Adventure* yang hendak dibentuk ialah permainan yang memiliki grafis 2 dimensi yang ber-*genre Platformer*. *Game* ini mempunyai alur cerita di mana Bejo sebagai Pemain utama dalam *game* akan

menuju ke sekolah dan tentunya di dalam perjalanan ia harus melewati berbagai rintangan dan juga menyelesaikan teka-teki seputar pelajaran sekolah. Sebagian fitur yang ditawarkan di dalam permainan ini [9]:

- Single Player
- Permainan dibangun dengan grafis 2D
- Permainan dibangun dengan tipe Platformer dengan skema sudut pandang orang ketiga
- Penyusunan AI lawan memakai algoritma A\*.

## 2.2 Desain Sistem

Berikut ini adalah *use case diagram* dari game bejo *Adventure* [1]:



Gambar 1. Use Case Diagram Game Bejo Adventure

Berikut di bawah ini adalah table dari definisi *use case* yaitu [1]:

Tabel 1. Definisi Use Case Diagram

| No | Use Case               | Deskripsi                             |
|----|------------------------|---------------------------------------|
| 1  | Mulai                  | Proses memulai permainan              |
| 2  | Pengaturan             | Mengatur aturan dalam game            |
| 3  | Credit                 | Proses menampilkan halaman credits    |
| 4  | Petunjuk atau materi   | Proses menampilkan cara bermain       |
| 5  | Pilih level            | Proses memilih level                  |
| 6  | Player bergerak        | Proses menggerakkan karakter          |
| 8  | Menghentikan Permainan | Proses menghentikan sejenak permainan |
| 9  | Melempar               | Proses player melempar                |
| 10 | Mengambil Kunci        | Proses Player mengambil kunci         |
| 11 | Menyelesaikan Soal     | Proses Player mengerjakan tugas       |

## 2.2 Gameplay

Permainan Bejo *Adventure* ini memiliki 3 stage ataupun tingkat yang sama- sama berkorelasi, dimana stage 1 wajib berakhir saat sebelum beralih ke stage 2 begitupun selanjutnya, ada pula isi dari pemberian tingkat ini bisa di amati di bawah ini [9], [14]:

- Stage 1 (Pedesaan)

- a) Pemain akan berkeliling di pedesaan.
  - b) Hanya terdapat musuh dengan level mudah.
  - c) Pemain bisa membunuh musuh dengan melempar objek yang telah disediakan dan pemain juga bisa tidak membunuh musuh tersebut.
  - d) Pemain harus menjawab soal yang telah diberikan.
  - e) Pemain harus mengumpulkan kunci untuk membuka pintu.
  - f) Misi dinyatakan selesai apabila pemain telah menyelesaikan semua soal yang telah diberikan
2. *Stage 2 (Hutan)*
    - a) Terdapat musuh dengan level mudah dan level sedang
    - b) Pemain bisa membunuh musuh dengan melempar objek yang telah disediakan dan pemain juga bisa tidak membunuh musuh tersebut.
    - c) Pemain harus menjawab soal yang telah diberikan.
    - d) Pemain harus mengumpulkan kunci untuk membuka pintu.
    - e) Misi dinyatakan selesai apabila pemain telah menyelesaikan semua soal yang telah diberikan
  3. *Stage 3 (Kota)*
    - a) Terdapat musuh dengan level mudah, level sedang dan level sulit
    - b) Pemain bisa membunuh musuh dengan melempar objek yang telah disediakan dan pemain juga bisa tidak membunuh musuh tersebut.
    - c) Pemain harus menjawab soal yang telah diberikan.
    - d) Pemain harus mengumpulkan kunci untuk membuka pintu.
    - e) Misi dinyatakan selesai apabila pemain telah menyelesaikan semua soal yang telah diberikan

### 2.3 Rancangan Komponen dalam *Game*

Rancangan komponen dalam *game* bejo *adventure* meliputi Karakter utama, dan Karakter musuh [10].

#### 1. Karakter Utama

Karakter utama *Game Platformer 2D Game Bejo Adventure*,



Gambar 2. Sketsa Karakter Utama *Game Bejo Adventure*

#### 2. Karakter Musuh

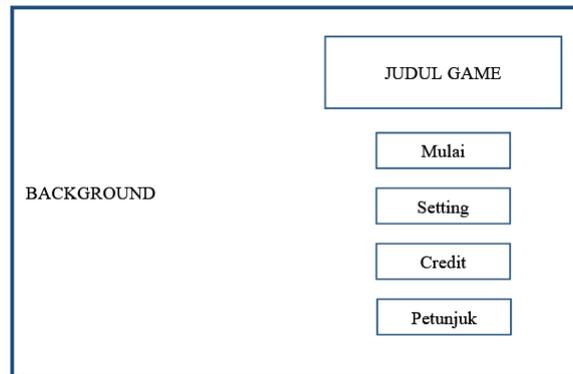
Karakter utama *Game Platformer 2D Game Bejo Adventure*,



Gambar 3. Sketsa Karakter Mush *Game Bejo Adventure*

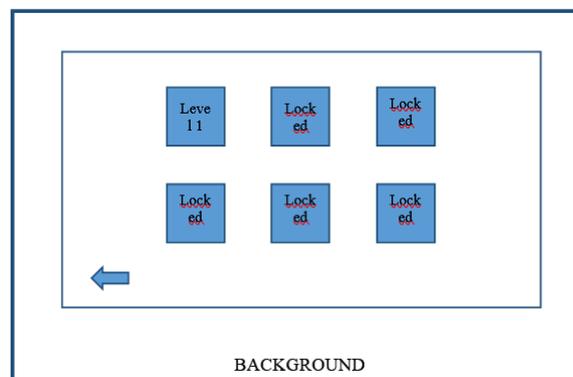
## 2.4 Rancangan Antarmuka Dalam *Game* [9]

### a. Menu Utama



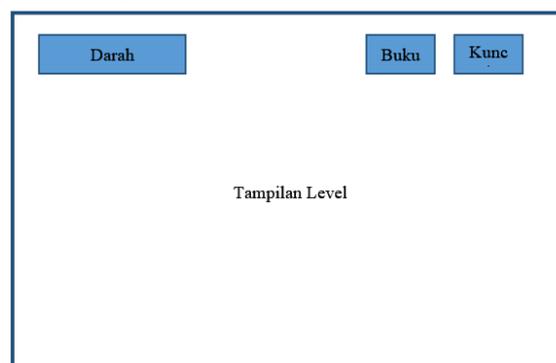
Gambar 4. Antarmuka *Game* Bejo Adventure

### b. Menu Pilih Level



Gambar 5. Antarmuka Pilih Level *Game* Bejo Adventure

### c. Antarmuka Tampilan *GamePlay*



Gambar 6. Antarmuka Permainan *Game* Bejo Adventure

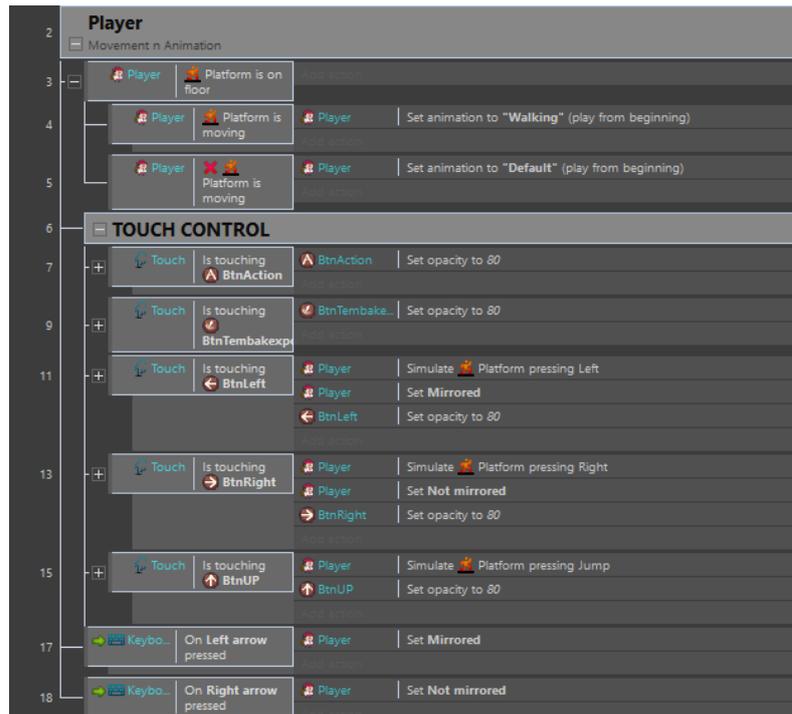
## 3. HASIL DAN ANALISA

pengujian dilakukan dengan dua tahap, yaitu pengujian *alpha* (fungsional) dan pengujian beta. Pengujian *Alpha* menekankan pada sisi fungsionalitas aplikasi dari program. Pengujian Beta dilakukan dengan menerapkan *game* secara langsung oleh pengguna.

### 3.1 Pengujian *Artificial Intelligence*

#### a. Navigasi Karakter (*Event Game*) [13], [3]

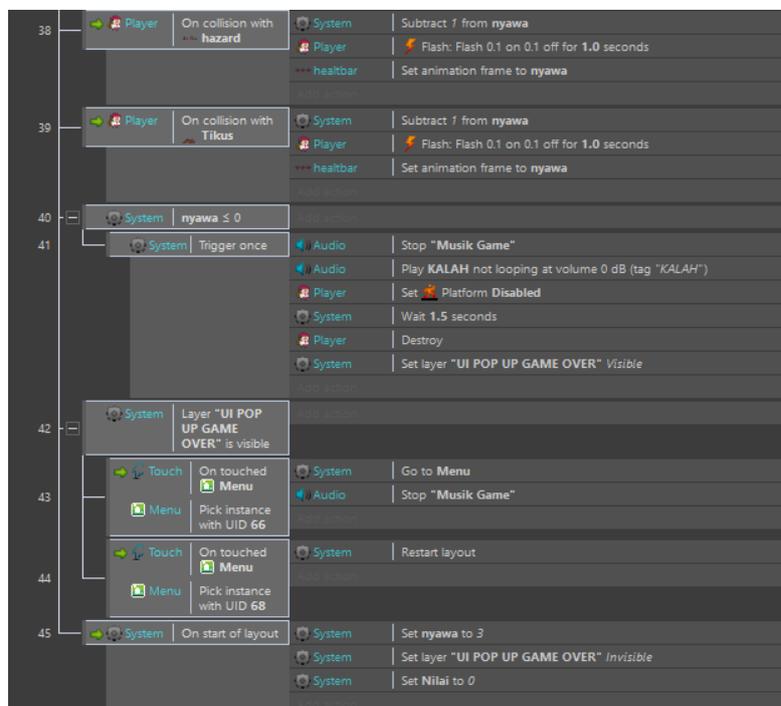
Merupakan dasar navigasi dari karakter seperti berjalan, melompat, menembak, menggunakan kunci



Gambar 7. Event Sistem Pergerakan Karakter

1. Pada baris ke-3 menunjukkan bahwa jika karakter berada pada tanah maka akan mengaktifkan fungsi berjalan.
2. Pada baris ke-4 menunjukkan bahwa jika karakter berjalan, maka akan mengaktifkan fungsi animasi berjalan, sedangkan pada baris ke-5, jika karakter tidak berjalan, maka akan mengaktifkan fungsi animasi diam.

**b. Mengurangi Darah Karakter (*Event Game*) [13], [3]**



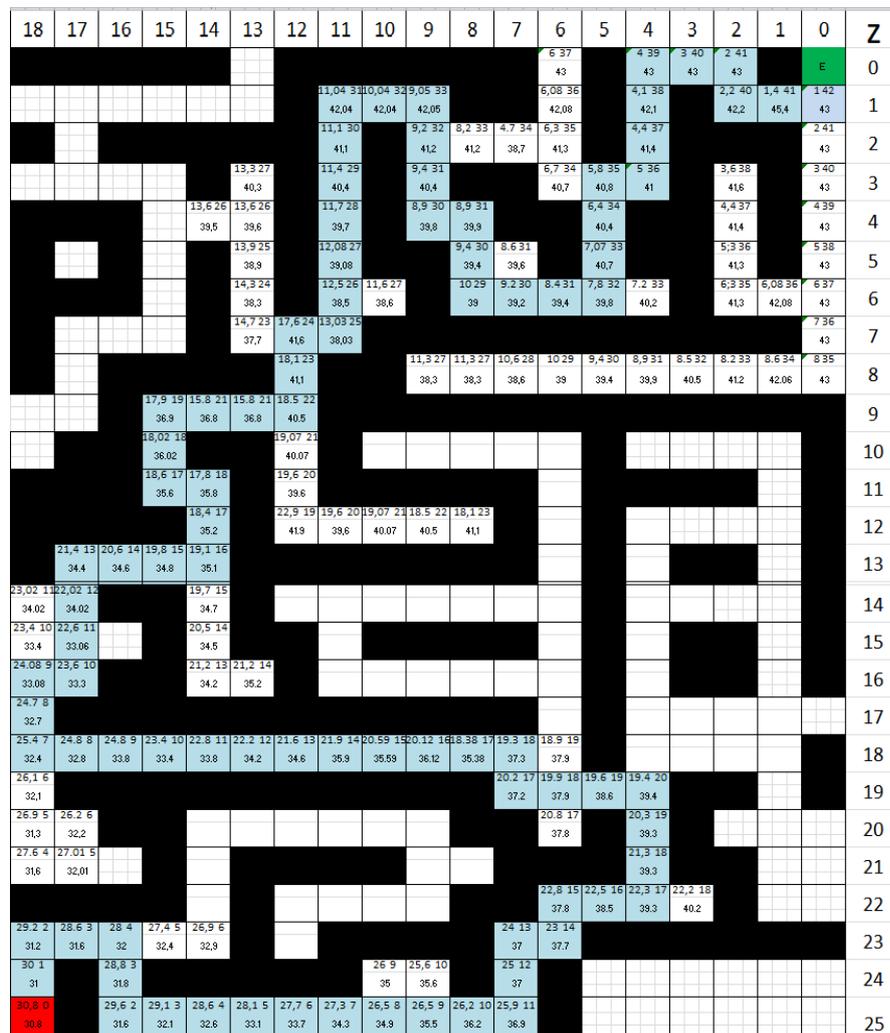
Gambar 8. *Event* Sistem Pengurangan Darah pada Karakter

1. Pada baris 38 dan 39 menunjukkan apabila karakter bertabrakan atau bersentuhan dengan objek berbahaya ataupun musuh, maka sistem akan mengurangi 1 nilai pada *global* variabel nyawa dan sistem akan mengatur animasi darah menjadi berkurang.
2. Pada baris 40 dan 41 menunjukkan bahwa jika nyawa telah mencapai nol (0), maka sistem akan memicu sistem *pop-up game over* dan menghilangkan karakter.

**c. Pengujian Algoritma A\* (*A-Star*) [12]**

Sebelum di implementasikan Algoritma *A-Star* akan diujicoba terlebih dahulu pada musuh yang digunakan sebagai pencarian jalur terpendek menuju Karakter. Jika Algoritma ini diterapkan pada musuh, maka musuh akan mencari karakter dan jika ditemukan (bertabrakan) maka akan mengurangi point darah.

Untuk mempermudah dalam mengetahui nilai jarak dan nilai heuristik, maka dalam stage pengujian akan dibentuk seperti labirin.



Gambar 9. Hasil Perhitungan

Misalnya terdapat sebuah kondisi dimana musuh berada pada koordinat (0,0) dan target berada pada koordinat (18,25). Maka cara pertama yaitu mengeksplorasi node yang berada pada sekitar musuh yang sekiranya dapat dilalui oleh musuh, lalu hitung semua node yang dapat dilewati tersebut. Nilai G pada algoritma A\* adalah sebuah nilai yang merepresentasikan biaya yang diperlukan untuk sampai ke node yang sedang ditinjau dari node awal. Nilai G ini biasanya dihitung dengan menambahkan biaya untuk bergerak dari node sebelumnya ke node yang sedang ditinjau. Dengan demikian, nilai G merupakan jumlah biaya minimum yang diperlukan untuk sampai ke node yang sedang ditinjau dari node awal. Nilai H merupakan perkiraan biaya minimum yang diperlukan untuk sampai ke node tujuan dari node yang sedang ditinjau. Nilai F merupakan perkiraan total biaya minimum yang diperlukan untuk sampai ke node tujuan dari node yang sedang ditinjau. Nilai F ini biasanya digunakan untuk menentukan node mana yang akan diproses selanjutnya

dalam algoritma A\*. Node yang memiliki nilai F terkecil akan diproses lebih dulu, karena dianggap sebagai jalur yang paling efisien untuk sampai ke node tujuan. Dengan menggunakan rumus di bawah [12]:

$$G(n) = \sqrt{Xn^2 + Yn^2}$$

$$H(n) = X(goal) - X(n) + Y(goal) - Y(n)$$

$$F(n) = G(n) + H(n)$$

$$\text{Posisi awal} = (0,0)$$

$$\text{Posisi tujuan} = (18,25)$$

$$G(n) = 0,1$$

$$= 1$$

$$H(n) = 18 - 0 + 25 - 1$$

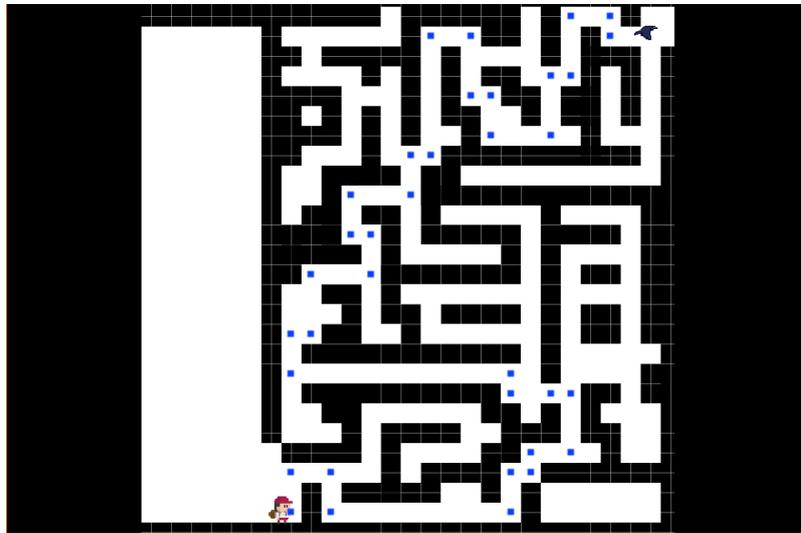
$$= 18 + 24$$

$$= 42$$

$$F(n) = 1 + 42$$

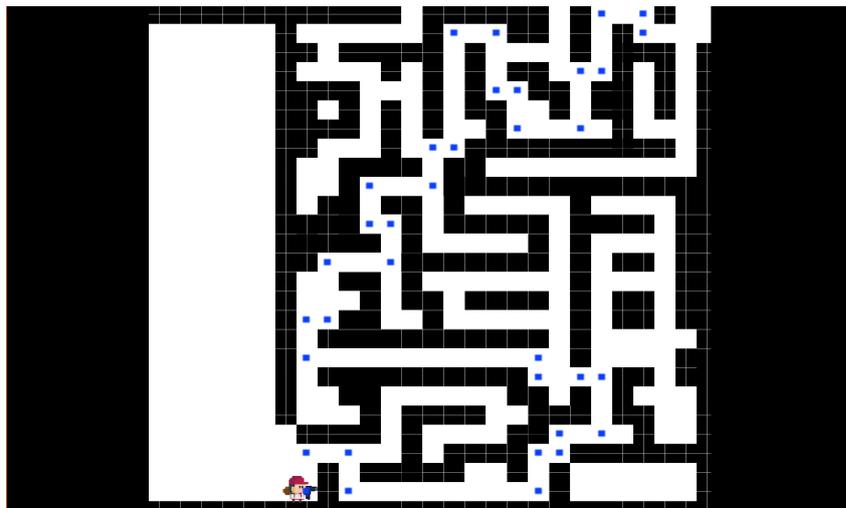
$$= 43$$

Untuk mencari jalur terpendek memerlukan 89 langkah dengan node yang diperiksa yaitu 150 node. Untuk lebih jelasnya dapat dilihat pada gambar 10.



Gambar 10. Tampilan Pengujian Musuh Mencari Jalur Target

Gambar 10 menampilkan rute perjalanan musuh kepada target yang dituju dan ditandai dengan lingkaran biru. Lingkaran biru tersebut hanya sebagai alat bantu bahwa jalur itulah yang akan dilalui musuh berdasarkan perhitungan yang telah diuji.

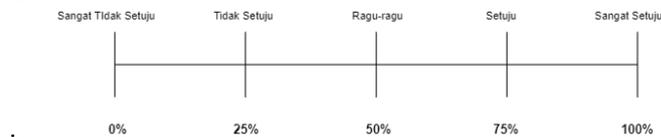


Gambar 11. Tampilan Pengujian Musuh Menemukan Jalur Target

Gambar 11 menampilkan bahwa musuh telah sampai pada titik target dan jika diterapkan pada permainan akan mengurangi darah daripada karakter utama.

### 3.2 Pengujian Beta

Kuesioner disebarikan setelah Penulis memberikan akses bermain terlebih dahulu kepada anggota sample dari populasi dipilih berdasarkan umur responden. Kuesioner disebarikan kepada 10 responden dengan usia 9 tahun (siswa MI Ash Shomadiyah Tuban kelas III), dari hasil kuesioner tersebut akan dilakukan perhitungan agar dapat ditarik kesimpulan terhadap penilaian penerapan sistem yang dibangun. Kuesioner terdiri dari 8 pertanyaan dengan menggunakan skala pengukuran yang digunakan adalah skala likert. berikut skor untuk jawaban kuesioner



Gambar 12. Skala Likert

Tabel 2. Skor Jawaban pada kuesioner

| Skala Jawaban | Keterangan          | Skor |
|---------------|---------------------|------|
| SS            | Sangat Setuju       | 5    |
| ST            | Setuju              | 4    |
| RG            | Ragu-ragu           | 3    |
| TS            | Tidak setuju        | 2    |
| STS           | Sangat tidak setuju | 1    |

Data hasil kuesioner yang telah diberikan dapat dicari persentasenya dari masing-masing jawaban dengan menggunakan rumus:

$$P = \frac{S}{\text{Skor ideal}} \times 100\%$$

Keterangan:

P = Nilai persentase yang dicari

S = Jumlah frekuensi dikalikan dengan nilai yang ditetapkan jawabannya.

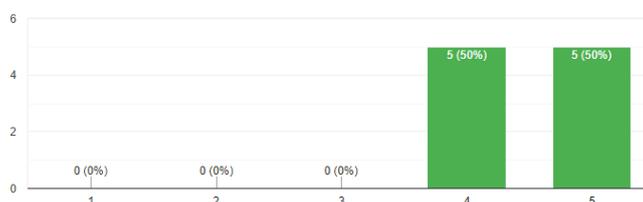
Skor ideal = Nilai tertinggi dikalikan dengan jumlah sampel.

Berikut ini adalah hasil persentase masing-masing jawaban yang sudah dihitung hasilnya dengan menggunakan rumus di atas.

1. Menurut Anda, apakah *game* bejo *adventure* dapat berpengaruh dalam mendukung aktivitas belajar dimana saja dan dapat diakses oleh siapa saja sebagai sarana informasi dan edukasi.?

Tabel 3. Perhitungan persentase pertanyaan ke-1

| Pertanyaan | Keterangan          | Skor | Frekuensi | S  |
|------------|---------------------|------|-----------|----|
| 1          | Sangat Setuju       | 5    | 5         | 25 |
|            | Setuju              | 4    | 5         | 20 |
|            | Ragu-ragu           | 3    | 0         | 0  |
|            | Tidak setuju        | 2    | 0         | 0  |
|            | Sangat tidak setuju | 1    | 0         | 0  |
| Jumlah     |                     |      | 10        | 45 |



Gambar 13. Grafik skor pertanyaan ke-1

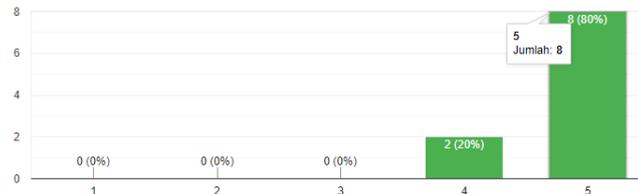
$$P = \frac{45}{50} \times 100\% = 90\%$$

Berdasarkan hasil persentase nilai di atas, maka dapat disimpulkan bahwa penilaian terhadap pertanyaan “Menurut Anda, apakah *game* bejo *adventure* dapat berpengaruh dalam mendukung aktivitas belajar dimana saja dan dapat diakses oleh siapa saja sebagai sarana informasi dan edukasi” adalah 90% dari yang diharapkan 100%, atau dapat dikategorikan sangat setuju.

2. Aplikasi yang di bangun memiliki desain antarmuka yang menarik

Tabel 4. Perhitungan persentase pertanyaan ke-2

| Pertanyaan | Keterangan          | Skor | Frekuensi | S  |
|------------|---------------------|------|-----------|----|
| 4          | Sangat Setuju       | 5    | 8         | 40 |
|            | Setuju              | 4    | 2         | 8  |
|            | Ragu-ragu           | 3    | 0         | 0  |
|            | Tidak setuju        | 2    | 0         | 0  |
|            | Sangat tidak setuju | 1    | 0         | 0  |
| Jumlah     |                     |      | 10        | 48 |



Gambar 14. Grafik skor pertanyaan ke-2

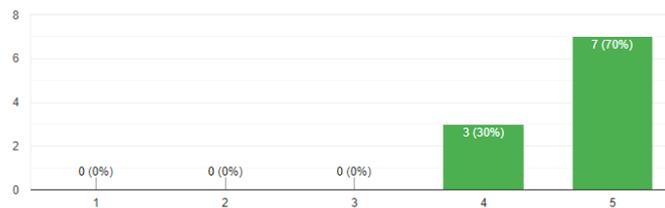
$$P = \frac{48}{50} \times 100\% = 96\%$$

Berdasarkan hasil persentase nilai di atas, maka dapat disimpulkan bahwa penilaian terhadap pertanyaan “Aplikasi yang di bangun memiliki desain antarmuka yang menarik” adalah 96% dari yang diharapkan 100%, atau dapat dikategorikan sangat setuju.

3. Aplikasi yang di bangun dapat membantu dalam kegiatan belajar

Tabel 5. Perhitungan persentase pertanyaan ke-3

| Pertanyaan | Keterangan          | Skor | Frekuensi | S  |
|------------|---------------------|------|-----------|----|
| 8          | Sangat Setuju       | 5    | 7         | 35 |
|            | Setuju              | 4    | 3         | 12 |
|            | Ragu-ragu           | 3    | 0         | 0  |
|            | Tidak setuju        | 2    | 0         | 0  |
|            | Sangat tidak setuju | 1    | 0         | 0  |
| Jumlah     |                     |      | 10        | 47 |



Gambar 15 Grafik skor pertanyaan ke-3

$$P = \frac{47}{50} \times 100\% = 94\%$$

Berdasarkan hasil persentase nilai di atas, maka dapat disimpulkan bahwa penilaian terhadap pertanyaan “Aplikasi yang di bangun dapat membantu dalam kegiatan belajar” adalah 94% dari yang diharapkan 100%, atau dapat dikategorikan sangat setuju.

### 3.3 Pengujian Alpha

Pengujian Alpha berfokus ke persyaratan fungsionalitas perangkat lunak. Skenario pengujian menjelaskan sistem yang ada pada aplikasi *game* *Bejo Adventure*. Skenario pengujian yang dilakukan pada aplikasi ini dapat dilihat pada tabel tabel 6 [11]:

Tabel 6. Skenario Pengujian

| NO. | Komponen yang di uji | Skenario Pengujian       |
|-----|----------------------|--------------------------|
| 1.  | Menu <i>Play</i>     | Klik <i>Play</i>         |
| 2.  | Menu <i>Credit</i>   | Klik Menu <i>Credit</i>  |
| 3.  | Menu <i>Setting</i>  | Klik Menu <i>Setting</i> |
| 4.  | Menu Petunjuk        | Klik Menu Petunjuk       |
| 5.  | Berjalan             | Tekan Tombol Panah Kanan |
|     |                      | Tekan Tombol Panah Kiri  |

|     |                                  |   |
|-----|----------------------------------|---|
| 6.  | Menembak                         | Tekan Tombol "A"  |
| 7.  | Membuka Pintu                    | Tekan Tombol "Space"  |
| 8.  | Mengambil Buku                   | Mengarahkan Karakter pada Buku  |
| 9.  | Melompat                         | Tekan Tombol Panah Atas   |
| 10. | Menjawab Soal                    | Mengarahkan dan Klik Tombol Jawaban   |
| 11. | Mengambil Kunci                  | Mengarahkan Karakter pada Kunci   |
| 12. | Menghentikan Permainan           | Mengarahkan dan Klik Tombol "Esc"   |
| 13. | <i>Pop-up Game Over</i>          | Pengguna tidak berhasil menyelesaikan tantangan atau karakter kehabisan nyawa |
| 14. | <i>Pop-up Level Complete</i>     | Pengguna berhasil menyelesaikan tantangan                                     |
| 15. | <i>AI Musuh Tingkat Mudah</i>    | Fungsi dan perilaku Musuh   |
| 16. | <i>AI Musuh Tingkat Menengah</i> | Fungsi dan perilaku Musuh   |
| 17. | <i>AI Musuh Tingkat Sulit</i>    | Fungsi dan perilaku Musuh   |

Berikut ini adalah kasus untuk menguji perangkat lunak yang sudah dibuat dengan menggunakan metode *black box testing* berdasarkan skenario pengujian aplikasi yang terdapat pada tabel 6 yang telah dijabarkan. Hasil pengujian yang akan dilakukan pada aplikasi ini selengkapnya dapat dilihat pada tabel 7 [11].

Tabel 7. Hasil Skenario Pengujian

| NO. | Kasus yang di uji              | Skenario Pengujian  | Hasil Yang diharapkan   | Kesimpulan |
|-----|--------------------------------|---|---|------------|
| 1.  | Menu <i>Play</i>               | Klik <i>Play</i>  | Sistem menampilkan pilihan level                                | Berhasil   |
| 2.  | Menu <i>Credit</i>             | Klik Menu <i>Credit</i>   | Sistem menampilkan menu credit                                  | Berhasil   |
| 3.  | Menu <i>Setting</i>            | Klik Menu <i>Setting</i>  | Sistem menampilkan menu setting dan tombol <i>fullscreen</i>    | Berhasil   |
| 4.  | Menu Petunjuk                  | Klik Menu Petunjuk  | Sistem menampilkan menu petunjuk                                | Berhasil   |
| 5.  | Berjalan                       | Tekan Tombol Panah Kanan  | Karakter bergerak ke kanan                                      | Berhasil   |
|     |                                | Tekan Tombol Panah Kiri   | Karakter bergerak ke kiri                                       | Berhasil   |
| 6.  | Menembak                       | Tekan Tombol "A"  | Karakter menembakkan apel                                       | Berhasil   |
| 7.  | Membuka Pintu                  | Tekan Tombol "Space"  | Pintu terbuka dan mengeluarkan suara                            | Berhasil   |
| 8.  | Mengambil Buku                 | Mengarahkan Karakter pada Buku  | Slot buku terisi dan sistem menampilkan <i>pop-up</i> soal      | Berhasil   |
| 9.  | Melompat                       | Tekan Tombol Panah Atas   | Karakter akan melompat  | Berhasil   |
| 10. | Menjawab Soal                  | Mengarahkan dan Klik Tombol Jawaban   | Pengguna mengklik jawaban yang tersedia                         | Berhasil   |
| 11. | Mengambil Kunci                | Mengarahkan Karakter pada Kunci   | Slot kunci terisi sesuai dengan warna kunci                     | Berhasil   |
| 12. | Menghentikan Permainan         | Mengarahkan dan Klik Tombol "Esc"   | Permainan terhenti  | Berhasil   |
| 13. | <i>Pop-up Game Over</i>        | Pengguna tidak berhasil menyelesaikan tantangan atau karakter kehabisan nyawa | Sistem Menampilkan <i>Pop-up Game Over</i>                      | Berhasil   |
| 14. | <i>Pop-up Level Complete</i>   | Pengguna berhasil menyelesaikan tantangan                                     | Sistem Menampilkan <i>Pop-up level complete</i>                 | Berhasil   |
| 15. | <i>AI Musuh Tingkat Mudah</i>  | Fungsi dan perilaku Musuh   | Musuh berjalan sendiri  | Berhasil   |
| 16. | <i>AI Musuh Tingkat Sedang</i> | Fungsi dan perilaku Musuh   | Musuh berjalan sendiri dan menembakkan objek                    | Berhasil   |
| 17. | <i>AI Musuh Tingkat Sulit</i>  | Fungsi dan perilaku Musuh   | Musuh berjalan sendiri, menembakkan objek dan mengejar Karakter | Berhasil   |

#### 4. KESIMPULAN

Berdasarkan hasil implementasi dan pengujian yang telah dilakukan pada *game* bejo *adventure* maka dapat diambil kesimpulan yaitu dengan dibuatnya *game* bejo *adventure*, maka pengguna dapat menambah pemahaman dan mengingat pelajaran matematika. Dengan *game* bejo *adventure*, maka dapat menjadi salah satu alternatif media pembelajaran. Lalu *game* bejo *Adventure* juga bisa digunakan sebagai sarana hiburan. Dari hasil implementasi Algoritma A\* (*A-Star*) pada penerapan NPC (*non-player character*) dapat berjalan dan sesuai dengan apa yang diharapkan. Kemudian kekurangan dari penelitian ini adalah belum bisa memunculkan nilai jarak pada Algoritma A Star, sehingga harus dilakukan perhitungan secara manual.

#### DAFTAR PUSTAKA

- [1] Agustin, R. D. (2017). Kerangka Analisis Komponen Konsep dan Desain Game. *Jurnal Ilmiah Teknologi Infomasi Terapan*, 3(2).
- [2] Clark, D. (2003). *Games and e-learning*. Epic Group.
- [3] Ginting, Budi Serasi, Hermansyah Sembiring (2013). Perancangan Game *Become a King* Berbasis *Artificial Intelligence*. *Jurnal Kaputama*, Vol.7 No.1:26-27.
- [4] Kent, S. L. (2010). *The Ultimate History of Video Games, Volume 1: From Pong to Pokemon and Beyond... the Story Behind the Craze That Touched Our Lives and Changed the World* (Vol. 1). Crown.
- [5] Kristanto, Andri. (2017) Kecerdasan buatan, Yogyakarta: Graha Ilmu.
- [6] Mahafi, A. G., & Hermawan, G. (2013). Game Edukasi Penyakit Malaria dan Cara Pencegahannya. *TEKNIK INFORMATIKA*, 2.
- [7] Ramadhani Aristama, (2008), Menggerakkan Karakter *Game* menggunakan Algoritma *Breadth-First Search (BFS)* dan Algoritma *A\*(A Star)*. *Jurnal Institut Teknologi Bandung*
- [8] Rifai, Wafda Adita. 2015. Pengembangan Game Edukasi Lingkungan Berbasis Android. Yogyakarta: Universitas Negeri Yogyakarta.
- [9] Rogers, S. (2014). *Level Up! The guide to great video game design*. John Wiley & Sons.
- [10] Sulaeman, F. S., & Permana, R. (2020). Pembuatan Game Platformer 2D Puzzle Logika Berbasis Android Menggunakan Metode Assets Workflow. *Media Jurnal Informatika*, 10(2), 17-20.
- [11] Widiyanto, W. (2017). Aplikasi Sistem Pakar Pendeteksi Kerusakan Pada Smartphone Menggunakan Metode Dempster Shafer. *Jurnal Bangkit Indonesia*, 6(2).
- [12] Yuliana, A., & Surya, I. (2012). Implementasi Algoritma A Star pada Pemecahan Puzzle 8. *Jurnal Teknik Informatika (Politeknik Caltek Riau)*, 1(2), 1-8.
- [13] Hardiyanti, H., Mustami, M. K., & Mu'nisa, A. (2020). Pengembangan Game Puzzle Berbasis Construct 2 sebagai Media Pembelajaran Sistem Peredaran Darah Kelas XI di SMA Negeri 1 Selayar. *Biolearning Journal*, 7(1), 6-11.
- [14] Ridoi, M. (2018). Cara mudah membuat game edukasi dengan Construct 2: tutorial sederhana Construct 2.