

# Penerapan Algoritma *A Star* ( $A^*$ ) pada *Game* Petualangan Labirin Berbasis Android

Imam Ahmad, Wahyu Widodo\*

Program Studi S1 Informatika

Fakultas Teknik dan Ilmu Komputer, Universitas Teknokrat Indonesia

\*wahyuwidodo497@gmail.com

**Abstrak-***Game* memiliki arti dasar permainan, dalam hal ini merujuk pada pengertian kelincihan intelektual. Salah satu contoh *game* yang populer adalah *game* labirin yang merupakan sebuah *puzzle* dalam bentuk percabangan jalan yang kompleks. Tulisan ini menyajikan laporan hasil pengembangan *game* petualangan labirin yang menceritakan mengenai petualangan kelinci melewati sebuah labirin untuk mencari makanannya. Suatu *game* memerlukan AI (*Artificial Intelligence*). AI yang digunakan dalam pembangunan *game* petualangan labirin ini adalah algoritma  $A^*$  (*A Star*) dengan *euclidean distance*. Algoritma ini merupakan algoritma pencarian untuk menemukan rute terpendek dengan *cost* paling minimum, algoritma  $A^*$  mencari rute terpendek dengan menjumlahkan jarak sebenarnya dengan jarak perkiraan sehingga membuatnya optimum dan *complete*. *Genre* dari *game* ini adalah *adventure* dan *puzzle*, dibangun dengan bahasa pemrograman java dengan *tools* Android Studio, AI yang digunakan adalah algoritma  $A^*$  dengan *euclidean distance* yang digunakan pada bantuan untuk melakukan pencarian jalur guna menemukan makanan kelinci. Hasil uji dari *game* petualangan labirin ini adalah jika pemain dalam kesusahannya menemukan jalur menuju makanan kelinci, maka pemain dapat menggunakan tombol bantuan yang akan dicarikan jalur terpendek oleh algoritma  $A^*$  (*A Star*) dengan *euclidean distance* untuk menuju lokasi tempat makanan kelinci berada.

**Kata Kunci:** *Adventure*, Algoritma  $A^*$ , *Artificial Intelligence*, *Game*, Labirin.

## 1. Pendahuluan

*Game* merupakan salah satu sarana untuk mengisi waktu luang dan melepas penat [1]. Hiburan dianggap penting bagi seseorang dikarenakan dengan adanya hiburan akan mampu menyegarkan kembali pikiran setelah disibukkan dengan berbagai aktivitas yang menguras tenaga dan otak. Tetapi seiring perkembangan teknologi, *Game* tidak hanya dijadikan sebagai sarana hiburan semata namun sekarang *Game* telah menjadi luas fungsinya, misalnya *Game* dapat dijadikan sarana pembelajaran yang biasa disebut *game* edukasi [2], lahan bisnis, dan dipertandingkan sebagai salah satu dari cabang olahraga oleh para profesional. Pada saat ini *Game* telah banyak dibuat mulai dari *Game* yang beraliran aksi, pertarungan, simulasi, strategi, olahraga, musik, edukasi, hingga petualangan.

*Game* petualangan (*adventure*) merupakan *Game* yang menuntut kemampuan berfikir untuk menganalisis tempat secara visual, memecahkan teka-teki maupun menyimpulkan rangkaian peristiwa dan percakapan karakter [3]. Hampir semua *Game* juga memiliki unsur *Game adventure* walaupun tidak spesifik. *Game adventure* biasanya memiliki satu tokoh atau obyek utama yang dimainkan dan dijalankan secara langsung dari awal sampai akhir. Salah satu contoh *Game* petualangan yaitu permainan labirin.

Labirin adalah sebuah *puzzle* dalam bentuk percabangan jalan yang kompleks dan memiliki banyak jalan buntu [4]. Prinsip permainannya adalah pemain harus

menemukan jalan keluar dari sebuah pintu masuk ke satu atau lebih pintu keluar. Untuk menyelesaikan permainan ini memerlukan penalaran-penalaran akan ketidakpastian, selanjutnya proses pemecahan masalah dalam melakukan pencarian jalur yang dilakukan secara alami oleh manusia.

Algoritma  $A^*$  (*A-Star*) merupakan metode pencarian yang membuang langkah-langkah yang tidak perlu dengan pertimbangan bahwa langkah-langkah yang dibuang sudah pasti merupakan langkah yang tidak akan mencapai solusi yang diinginkan [5]. Jadi, algoritma ini cocok untuk diterapkan dalam pencarian jalur di permainan labirin.

*Mobile game* adalah sebuah *game* yang didesain dan dimainkan oleh *mobile devices*, seperti *PDA*, *smartphone*, *tablet PCs*, dan *portable media player* [6]. Sekarang ini, *mobile game* telah dibuat dalam berbagai *platform* seperti Symbian, Apple IOS, Android serta Windows Phone. Keuntungan *mobile game* yaitu pengguna dapat bermain *game* dimana saja mereka mau selama mereka mempunyai *mobile devices* yang mampu menjalankan *mobile games*. Tulisan ini menyajikan hasil penelitian terkait dengan pengembangan *mobile game* khusus pada *smartphone* berbasis Android. Android dipilih karena telah menjadi sistem operasi *mobile* yang populer dan banyak diminati oleh pengguna.

## 2. Landasan Teori

### 2.1 Algoritma $A^*$ (*A Star*)

Menurut Arhami dkk (2006) algoritma  $A^*$  merupakan algoritma *best first search* dengan pemodifikasian fungsi

heuristik. Algoritma ini meminimalkan total biaya lintasan, dan pada kondisi yang tepat akan memberikan solusi yang terbaik dalam waktu yang optimal [6].

Algoritma A\* membutuhkan dua antrean, yaitu *OPEN* dan *CLOSED* [6]. *OPEN* adalah senarai (*list*) yang digunakan untuk menyimpan simpul-simpul yang pernah dibangkitkan dan nilai heuristiknya telah dihitung tetapi belum dipilih sebagai simpul terbaik (*best node*). Dengan kata lain, *OPEN* berisi simpul-simpul yang masih memiliki peluang untuk terpilih sebagai simpul terbaik, sedangkan *CLOSED* adalah senarai untuk menyimpan simpul-simpul yang sudah pernah dibangkitkan dan sudah pernah dipilih sebagai simpul terbaik. Artinya, *CLOSED* berisi simpul-simpul yang tidak mungkin dipilih sebagai simpul terbaik (peluang untuk terpilih sudah tertutup). Selain antrean tersebut, ada juga fungsi *heuristik* yang memprediksi keuntungan setiap *node* yang dibuat. Hal ini akan memungkinkan algoritma untuk melakukan pencarian-pencarian lintasan yang lebih dapat diharapkan. Fungsi tersebut disebut sebagai pendekatan dari fungsi yang merupakan fungsi evaluasi yang sebenarnya terhadap *node n*. Dalam banyak penerapan, akan lebih baik jika fungsi ini didefinisikan sebagai kombinasi atau jumlah dua komponen yaitu  $g$  dan  $h$ . Fungsi  $f'$  merupakan ukuran biaya yang dikeluarkan dari keadaan awal sampai ke *node n*. Nilai yang diperoleh  $f'$  merupakan jumlah biaya penerapan setiap aturan yang dilakukan pada sepanjang lintasan terbaik menuju suatu simpul dan bukan merupakan hasil estimasi. Adapun fungsi  $g$  merupakan pengukur biaya tambahan yang harus dikeluarkan dari *node n* sampai mendapatkan tujuan. Perlu diketahui bahwa  $h$  tidak negatif karena bila negatif, maka lintasan yang membalik siklus pada *grafikan* tampak lebih baik dengan semakin panjangnya lintasan.

Secara matematis, fungsi sebagai estimasi fungsi evaluasi terhadap *node n* dapat dituliskan sesuai Persamaan 1.

$$f'(n) = g(n) + h(n) \quad (1)$$

dengan

$f'(n)$  = fungsi evaluasi,

$g(n)$  = biaya yang sudah dikeluarkan dari keadaan awal sampai keadaan  $n$ ,

$h(n)$  = estimasi biaya untuk sampai pada suatu tujuan mulai dari  $n$ .

## 2.2 Game (Permainan)

Sebuah permainan adalah sebuah sistem di mana pemain terlibat dalam konflik buatan. Disini pemain berinteraksi dengan sistem dan konflik dalam permainan merupakan rekayasa atau buatan. Dalam permainan terdapat peraturan yang bertujuan untuk membatasi perilaku pemain dan menentukan permainan [8]. *Game* bertujuan untuk menghibur, biasanya *game* banyak disukai oleh anak-anak hingga orang dewasa. *Game* sebenarnya penting untuk perkembangan otak, untuk meningkatkan konsentrasi dan melatih untuk memecahkan masalah dengan tepat dan cepat karena dalam *game* terdapat berbagai konflik atau masalah yang menuntut pemain untuk menyelesaikannya dengan cepat dan tepat.

## 2.3 Labirin

*Maze* atau labirin adalah sebuah *puzzle* dalam bentuk percabangan jalan yang kompleks dan memiliki banyak jalan buntu [4]. Tujuan permainan ini adalah pemain harus menemukan jalan keluar dari sebuah pintu masuk ke satu atau lebih pintu keluar. Bisa juga kondisi pemain menang yaitu ketika dia mencapai suatu titik di dalam *maze* tersebut. *Maze* dalam dunia nyata banyak dibuat di taman atau ruangan-ruangan dengan pembatas berupa pagar tanaman, tembok atau pagar. Ukurannya bervariasi, tergantung ukuran ruangan atau taman tersebut.

## 2.4 Android

Android merupakan sebuah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Android menyediakan *platform* terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak [9]. Android umum digunakan di *smartphone* dan juga tablet PC. Fungsinya sama seperti sistem operasi Symbian di Nokia, iOS di Apple, dan BlackBerry OS.

## 3. Metode Penelitian/Eksperimen

### 3.1 Pengumpulan Data

Penyusunan proposal penelitian ini tentu membutuhkan berbagai keterangan-keterangan lengkap. Peneliti mengumpulkan data-data tersebut dengan dua metode, yaitu:

a. Studi Literatur.

Proses pengumpulan data yang dilakukan dengan mempelajari berbagai bentuk bahan tertulis yang berupa buku-buku, artikel, dokumen-dokumen, termasuk laporan yang ada kaitannya secara langsung. Pada metode ini, penulis mempelajari buku-buku penunjang dan dokumen yang terkait serta dapat dijadikan acuan untuk mengadakan maupun menguraikan pembahasan.

b. Wawancara

Wawancara adalah salah satu cara yang digunakan untuk menggali informasi berupa data secara lisan. Data yang diperoleh dari wawancara kepada narasumber akan digunakan sebagai bahan penelitian tentang penerapan algoritma pada *game* petualangan labirin. Dalam penelitian ini, peneliti melakukan wawancara kepada guru Sekolah Dasar.

### 3.2 Analisis Pembangunan Aplikasi Game Petualangan Labirin

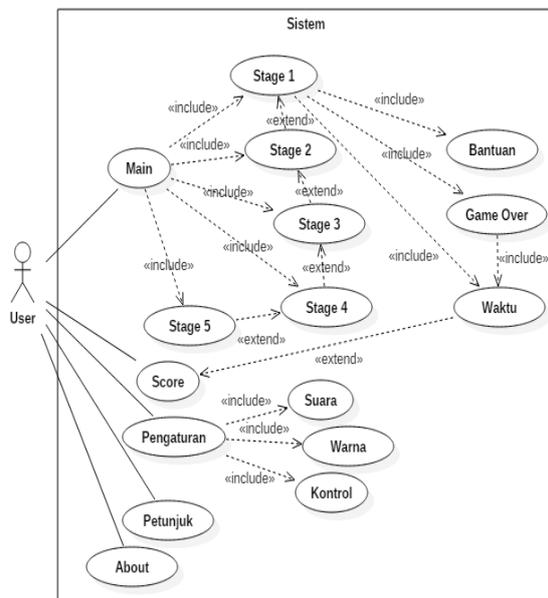
Pada bagian ini akan dijelaskan mengenai analisis-analisis *game* yang akan dibuat dari pengenalan *game* sampai dengan metode dan algoritma yang akan digunakan.

#### a. Pengenalan Game

Nama Pada Aplikasi *game* ini adalah Petualangan Labirin. *Game* ini merupakan *game* ber *genre* Adventure. Pemberian nama pada *game* ini dilihat dari permainannya karena pemain harus menemukan jalur keluar dari pintu masuk menuju pintu keluar dari sebuah labirin. *Game* ini merupakan *game* dua dimensi, dan dimainkan dengan menggunakan *smartphone* berbasis android. Permainan ini bersifat *singleplayer* dan tidak memerlukan jaringan *internet* untuk memainkannya (*offline*).

Tabel 1. *Gameplay*

No	Aksi
1	Pemain masuk ke menu utama
2	Pemain memilih menu main
3	Pemain memilih <i>stage</i>
4	Menampilkan pilihan <i>stage</i> 1 sampai 5
5	Setelah memilih <i>stage</i> , <i>game</i> ditampilkan
6	Pemain menggerakkan kelinci untuk menemukan makanannya, dimana dalam pencarian tersebut pemain harus melewati banyaknya jalur dalam labirin.
7	Jika pemain terjebak dalam jalur buntu, maka karakter dapat berbalik arah
8	Jika pemain kesulitan mencari jalur keluar, pemain bisa menggunakan tombol bantuan untuk membantu menemukan jalur terdekat.
9	Tombol bantuan hanya bisa digunakan dua kali
10	Permainan ini dibatasi oleh waktu
11	Menampilkan <i>score</i> dan lanjut <i>stage</i>
12	Menampilkan <i>stage</i> selanjutnya
13	Jika pemain tidak dapat mencari makanan kelinci atau pintu keluar dari waktu yang sudah ditentukan maka <i>game over</i> .
14	Pemain dapat memilih memainkan ulang <i>game</i> atau kembali ke menu utama
15	Pemain memilih menu <i>score</i> untuk mengetahui <i>score</i> yang sudah didapat
16	Pemain memilih menu pengaturan untuk mengatur suara
17	Pemain memilih menu petunjuk untuk mempelajari petunjuk permainan
18	Pemain memilih menu keluar untuk keluar dari <i>game</i> .

Gambar 1. *Use Case Diagram*

### b. Alur Permainan

Alur permainan dalam *game* Petualangan Labirin adalah sebagai berikut:

- Kelinci bergerak menggunakan sentuhan tampilan layarmelalui jari.
- Pemain harus mencari makanan kelinci, atau pintu keluar.
- Bila pemain terjebak dalam jalur buntu, pemain dapat berbalik arah.
- Bila pemain tidak dapat mencari makanan kelinci atau pintu keluar dari waktu yang sudah ditentukan maka *game over*.
- Jika *game over*, maka pemain dapat memilih untuk memainkan ulang *game* atau kembali ke menu utama.
- Jika pemain sampai ke tujuan per *stage*, maka skor akan ditentukan dari sisa waktu yang diberikan dan menampilkan pilihan lanjut *stage* berikutnya.

### c. Misi Permainan

Misi dari permainan ini adalah melakukan pencarian jalur dari tempat awal permainan menuju tempat makanan kelinci. Selain itu pemain juga harus menemukan makanan kelinci dari waktu yang disediakan.

### d. Menentukan *Gameplay*

*Gameplay* merupakan sifat interaktivitas pemain dimana pemain dapat berinteraksi dengan *game* petualangan labirin dan *game* tersebut dapat menerima reaksi yang dipilih pemain. *Gameplay* dapat dilihat pada tabel 1.

### e. Sasaran *User* dan Kegunaan Aplikasi Permainan

Menurut Piaget dalam Santrock (1986) membagi perkembangan intelektual anak ke dalam 4 periode utama yang berkorelasi dengan seiring pertambahan usia: Periode sensori motor (usia 0–2 tahun), Periode praoperasional (usia 2–7 tahun), Periode operasional konkret (usia 7–11 tahun), Periode operasional formal (usia 11 tahun sampai dewasa). Tahapan operasional konkret, Tahapan ini adalah tahapan ketiga dari empat tahapan. Muncul antara usia tujuh sampai sebelas tahun dan mempunyai ciri berupa penggunaan logika yang memadai [9].

Oleh sebab itu, untuk menyeimbangkan penggunaan *game* dengan umur anak, dalam hal ini pemain yang memainkan aplikasi permainan petualangan labirin. Anak umur 7 sampai 11 tahun, alasannya adalah pada umur ini penggunaan logika yang memadai dalam memainkan *game* petualangan labirin yang memerlukan penalaran-penalaran logika untuk memainkannya.

Kegunaan dari aplikasi permainan petualangan labirin ini selain menjadi sarana hiburan bagi para pemainnya juga dapat melatih kecepatan otak dalam berpikir, melatih cara pemecahan masalah, melatih koordinasi mata dan tangan. Pengambilan keputusan pada *game* ini akan melatih otak pemain untuk terampil dan cepat dalam mengambil tindakan serta melatih kesabaran dan konsentrasi anak.

### 3.3 Analisis Algoritma A\* (A Star)

AI merupakan faktor terpenting dalam meningkatkan *gameplay* pada semua jenis permainan. Hal penting dalam menentukan keberhasilan suatu aplikasi *game* yang menerapkan AI adalah kesuksesan dalam pencarian dan pelacakan. Pada dasarnya ada dua teknik pencarian dan pelacakan yang digunakan, yaitu pencarian buta (*blindsearch*)

dan pencarian terbimbing (*heuristics search*). Dalam pencarian terbimbing ada beberapa algoritma A\*. Algoritma ini dapat menyelesaikan permasalahan secara optimal untuk menemukan jalan terpendek menuju suatu tujuan.

### 3.4 Rancangan Use Case Diagram

*Use case diagram* menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Sebuah *use case* merepresentasikan suatu interaksi antara aktor dengan sistem seperti pada Gambar 1.

### 3.5 Skenario Testing (Pengujian)

Skenario pengujian menggunakan standar kualitas ISO 9126. Menurut Assaf Ben David (2011), untuk mengukur kualitas suatu aplikasi *mobile* cukup dengan melakukan empat aspek pengujian, yaitu *functional testing*, *portability testing*, *usability testing*, dan *efficiency testing*, sehingga yang diuji dalam penelitian ini hanya 4 aspek tersebut saja [10].

## 4. Hasil dan Pembahasan

### 4.1 Implimentasi Perangkat Keras

Perangkat keras atau *device* yang digunakan untuk menjalankan aplikasi *game* petualangan labirin tidak harus yang berspesifikasi tinggi. Spesifikasi minimum yang dapat digunakan untuk menjalankan aplikasi *game* petualangan labirin adalah sebagai berikut:

- 1) RAM 512 MB
- 2) Processor Dual Core
- 3) Unit Pengolahan Grafis (GPU) OpenGL ES 2.0 dan 3.0

### 4.2 Implimentasi Perangkat Lunak

Adapun spesifikasi perangkat lunak yang digunakan untuk menjalankan aplikasi *game* petualangan labirin adalah perangkat *device* yang sudah berbasis android dengan sistem operasi versi Jelly Bean, Kitkat, Lolipop, dan Marshmallow.

### 4.3 Implementasi Antarmuka

#### a. Antarmuka Menu Utama

Antarmuka menu utama ditunjukkan dalam Gambar 2. Menu yang tersedia dalam menu utama adalah:

- Pilih Main untuk mulai masuk ke menu pilih *stage* permainan.
- Pilih *Score* untuk melihat hasil *score*.
- Pilih *Setting* untuk mengatur suara, warna, dan tipe kontrol.
- Pilih *Help* untuk melihat petunjuk permainan.
- Pilih *About* untuk melihat petunjuk permainan.
- Pilih *iconSound* untuk mengatur suara



Gambar 2. Antarmuka Menu Utama



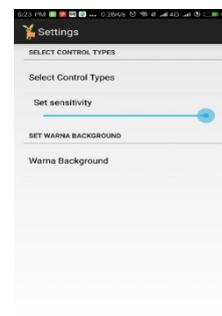
Gambar 3. Antarmuka Pilih Stage



Gambar 4. Antarmuka Permainan



Gambar 5. Antarmuka Score



Gambar 6. Antarmuka Setting



Gambar 7. Antarmuka Petunjuk

**b. Antarmuka Select Stage**

Gambar 3 menunjukkan antarmuka menu Pilih Stage. Menu yang tersedia dalam menu Pilih Stage adalah:

- Pilih *Stage 1* untuk mulai masuk ke arena permainan 1
- Pilih *Stage 2* untuk mulai masuk ke arena permainan 2
- Pilih *Stage 3* untuk mulai masuk ke arena permainan 3
- Pilih *Stage 4* untuk mulai masuk ke arena permainan 4
- Pilih *Stage 5* untuk mulai masuk ke arena permainan 5

**c. Antarmuka Permainan**

Antarmuka menu Permainan ditunjukkan dalam Gambar 4. Menu yang tersedia dalam menu Permainan adalah:

- Pilih Bantuan untuk bantuan mencari jalur terdekat.
- Pilih *icon* dipojok kanan atas untuk mengganti *type control game*.
- Waktu adalah batas waktu untuk menyelesaikan permainan.

**d. Antarmuka Score**

Antarmuka menu Score ditunjukkan dalam Gambar 5. Score pada menu ini diurutkan berdasarkan waktu terkecil ke waktu terbesar.

- *Score* diurutkan berdasarkan waktu terkecil ke waktu terbesar.

**e. Antarmuka Setting**

Antarmuka menu Setting ditunjukkan dalam Gambar 6. Menu yang tersedia dalam menu Setting adalah:

- Pilih *selectcontroltypes* untuk mengatur tipe kontrol permainan yang akan digunakan.
- *Setsensitivity* untuk mengatur sensitifitas *touchscreen*.
- *Setlabyrinthcolor* untuk mengatur warna *background labirin*

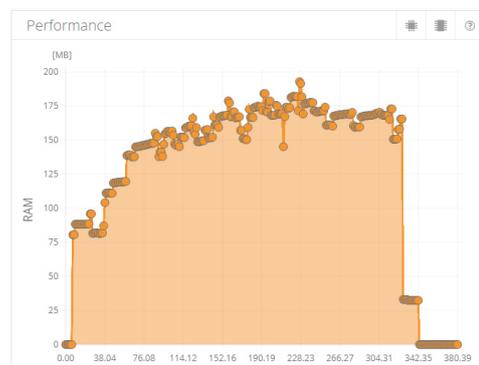
**f. Antarmuka Petunjuk**

Antarmuka menu Petunjuk ditunjukkan dalam Gambar 7. Menu ini berisi petunjuk permainan.

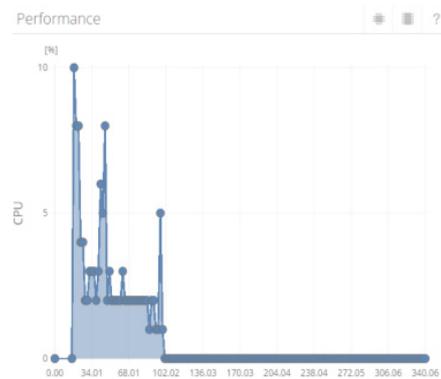
- Petunjuk berisi petunjuk permainan.
- Petunjuk juga berisi petunjuk *game*.

Tabel 2. Hasil Aspek *portability*

No	Jenis Perangkat	Jenis Android	Proses Instalasi	Proses <i>Running Game</i>
1	Evercroos A80A	4.2.2 (Jelly Bean )	Berhasil	Berjalan tanpa ada <i>error</i>
2	Asus Zenfone 5	4.4 (Kitkat)	Berhasil	Berjalan tanpa ada <i>error</i>
3	LG G3	5.0 (Lolipop)	Berhasil	Berjalan tanpa ada <i>error</i>
4	Samsung J5 Prime	6.0 (Marshmallow)	Berhasil	Berjalan tanpa ada <i>error</i>
5	Xiaomi MI 4C	7.0 (Nougat)	Berhasil	Berjalan tanpa ada <i>error</i>



Gambar 8. Penggunaan *memory* oleh aplikasi



Gambar 9. Penggunaan CPU oleh aplikasi

**4.4 Pengujian**

**a. Pengujian *Usability***

Pada aspek ini, metode yang digunakan yaitu dengan menggunakan kuesioner yang diberikan kepada siswa berumur 7-11 tahun dengan cara mencoba aplikasi terlebih dahulu sebelum mengisi kuisisioner. Kuisisioner ini mengadaptasi dari J.R Lewis.

Hasil yang diperoleh adalah persentase 79,92% dari pengujian *usability*. Berdasarkan skor persentase yang didapat, maka kualitas perangkat lunak dari sisi *usability* telah sesuai dengan atribut *usability* atau bisa dikategorikan sebagian besar responden sangat setuju.

Gambar 10. Bantuan pada *stage 2*

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
0	61.61	59.61	42.69	54.13	72.11														
1	61.61	52	59.41	54.31	65.29	71	71.69	72.25											
2	61.61		54.45	59.41	63.94	67.49	68.07	72.35											
3	62.83		52.80	54.62	55.07	55.83		72.40		112.8		132.3	147.2	159.09					
4	63.31		42.68	55.23		62.70	63.23	72.47		97.84	111.3	120.8							
5	63.38		58.24	55.41	59.52	59.82	72.54			87.88		130.3							
6	64.87	55.17	55.48	58.20	55.80	57.02	57.47	57.93	74.02	72.61		80.92		139.4					
7	65.17									72.69	73.31	73.97		148.4	158.24				
8	66.49	58.75	40.04	40.34	40.87	41.02	41.40	41.80	42.24	42.72		73.38							
9												73.46	74.08	78.74	80.45				
10												73.55		81.52					
11												73.65		84.86	83.56				
12												73.76		84.92					
13														85	85.64	87.36	89.16		
14																			
15																			
16																			
17																			
18																			
19																			
20																			
21																			
22																			
23																			
24																			
25																			

Gambar 11. Hasil perhitungan

### b. Pengujian Functionality

Pada aspek ini, metode yang digunakan yaitu dengan menggunakan *checklist* angket yang diisi oleh dua orang ahli dibidang *software engineering* dengan cara menguji aplikasi terlebih dahulu sebelum mengisi angket (Tabel 2).

Pada Gambar 8 persentase yang diperoleh yaitu 93,33% dari pengujian *functionality*. Dari analisis skala *Guttman*, titik kesesuaian di atas 50% yaitu 93,33%, sehingga dapat dikatakan kualitas perangkat lunak dari sisi *functionality* telah sesuai dengan atribut *functionality*.

### c. Pengujian Portability

Pada aspek ini, metode yang digunakan yaitu dengan menggunakan percobaan ke beberapa *smartphone* dengan sistem operasi android versi Jelly Bean, Kitkat, Lolipop, Marshmallow, dan Nougat untuk melihat apakah aplikasi *Game Adventure Labyrinth* dapat di *install* dan dijalankan (tabel 2).

### d. Pengujian Efficiency

Pengujian aspek *efficiency* menggunakan tools *Testdroid* yang diakses di <https://cloud.testdroid.com>. Dengan *Testdroid* dapat diamati penggunaan *memory* (RAM),

dan CPU [11]. Dalam pengujian aspek *efficiency* ini digunakan *device* yang sudah disediakan oleh *Testdroid* yaitu menggunakan *device* LG Google Nexus 5 D820 5.0. Berikut ini hasil pengujian aspek *efficiency* dengan tools *Testdroid*:

#### a) Memory (RAM)

Berdasarkan Gambar 8, penggunaan *memory* oleh aplikasi memang cukup aman dan aplikasi ini tidak mengalami kekurangan *memory* yang menyebabkan terjadinya *memoryleak* yang dapat mengakibatkan aplikasi berhenti.

#### b) CPU,

Berdasarkan Gambar 9, penggunaan CPU rata-rata berada di angka 10%. Angka tersebut masih berada di bawah batas aman yang ditetapkan oleh *Little Eye (mobile app analysis tools)* yaitu 15%.

Gambar 10 menyajikan hasil pengujian aplikasi algoritma A\* (A Star) untuk menguji perangkat lunak yang sudah dibangun sudah berjalan seperti yang diharapkan atau belum. Pengujian dilakukan berdasarkan algoritma A\* (A Star) yang ada pada

#### e. Pengujian Algoritma A\* (A Star)

Di bawah ini merupakan pengujian aplikasi algoritma A\* (A Star) untuk menguji perangkat lunak yang sudah dibangun sudah berjalan seperti yang diharapkan atau belum. Pengujian dilakukan berdasarkan algoritma A\* (A Star) yang ada pada tombol bantuan *stage 2*, adapun dapat dilihat pada gambar 9.

Perhitungan algoritma A\* menggunakan *euclidean heuristic* yang digunakan untuk mencari jalur terpendek antara dua buah simpul horizontal dan vertikal. Nilai  $n.x, n.y = 0$  dan  $goal.x, goal.y = 1$  yang diambil dari jarak simpul pada *game*. Nilai *node (grid)* yaitu  $20 \times 26$  *pixel* dimana 1 *grid* = 1 *pixel* ke arah *horizontal* atau 1 *pixel* ke arah *vertical*. Nilai  $h(n)$  *heuristic* di dapat dari jarak perkiraan dari simpul awal ke simpul tujuan. Nilai  $f(n)$  didapat dari penjumlahan dari  $g(n)$  dan  $h(n)$ . Untuk lebih jelasnya dapat dilihat seperti pada gambar 11.

$$\text{Posisi simpul awal} = n.x: 0, n.y: 0$$

$$\text{Posisi simpul tujuan} = \text{Goal.x: } 19, \text{Goal.y: } 25$$

$$g(0,1) = 1$$

$$h(n) = \sqrt{(n.x - goal.x)^2 + (n.y - goal.y)^2}$$

$$h(0,1) = \sqrt{(0 - 19)^2 + (1 - 25)^2}$$

$$h(0,1) = \sqrt{(-19)^2 + (-24)^2}$$

$$h(0,1) = \sqrt{361 + 576} = \sqrt{937}$$

$$h(0,1) = 30,61$$

$$f(n) = g(n) + h(n)$$

$$f(0,1) = 1 + 30,61 = 31,61$$

Untuk mencari jalur terpendek dari *stage 1* membutuhkan 156 langkah dengan simpul yang diperiksa yaitu 186 simpul (data rinci terlampir), untuk lebih singkatnya dari perhitungan algoritma A\* (A Star) dapat dilihat pada Gambar 11.

Dari hasil pengujian algoritma A\*(A star) yang dilakukan, dapat disimpulkan bahwa bantuan yang ada pada *game* petualangan labirin sudah berjalan menggunakan *euclidean heuristic*.

## 5. Kesimpulan

### 5.1 Kesimpulan

Penerapan algoritma A\* (A Star) untuk menu bantuan pada *game* petualangan labirin sudah berjalan menggunakan *euclidean heuristic*. Kegunaan algoritma A\* (A Star) ini adalah untuk mencari jalur terpendek bagi kelinci (karakter) menuju tempat makanannya.

*Game* petualangan labirin dapat dijalankan menggunakan *smartphone* dengan sistem operasi Android versi Jelly Bean, Kitkat, Lolipop, Marshmallow, dan Nougat. *Game* petualangan labirin bisa dimainkan dengan kontrol *touchscreen* dan *sensor gravity*.

### 5.2 Saran

Saran yang dapat diberikan untuk pengembangan lebih lanjut dari Penerapan Algoritma A\* (A Star) Pada *Game* Petualangan Labirin antara lain:

- a. Dapat menggunakan algoritma selain A\* (A Star) dalam hal pencarian jalur terdekat.
- b. *Stage* ditambah dan dibuat dinamis atau acak di setiap *stage* nya.
- c. Kontrol permainan pada *game* ditambah dengan sensor suara dan getaran.
4. *Game* petualangan labirin bisa dibuat versi 3 dimensinya.

## 6. Daftar Pustaka

- [1] Khairy, M. S., Herumurti, D., & Kuswardayan, I. 2016. Analisis Pengaruh Penggunaan Game Edukasi pada Penguasaan Kosakata Bahasa Asing dengan Studi Kasus Game Edukasi Bahasa Arab. *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika*, 2(2), 42-48.
- [2] Sudarmilah, E., & Wibowo, P.A., 2016. Aplikasi Augmented Reality Game Edukasi untuk Pengenalan Organ Tubuh Manusia *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika*, 2(1), 20-25..
- [3] Wicaksono, R. M., & R Sandhika Galih Amalga, D. S., 2016. Pembuatan Game Petualangan Menggunakan Construct2 (Doctoral dissertation, Fakultas Teknik Unpas).
- [4] Tilawah, H., 2011. Penerapan Algoritma A-Star (A\*) Untuk Menyelesaikan Masalah Maze. *Jurnal, Teknik Informatika, Institut Teknologi Bandung, Bandung*.
- [5] Putra, R. D., Aswin, M., & Djurianto, W. 2012. Pencarian Rute Terdekat Pada Labirin Menggunakan Merode A\*. *Jurnal EECCIS*, 6(2).
- [6] Rahadian, M., Agustri, S., & Suhandi, N. 2016. Pembangunan Game Ayo Tarik Berbasis Android. *Jurnal Informatika Global*, 6(1).
- [7] Anita, D., & Arhami, M. 2006. Konsep Kecerdasan Buatan. Yogyakarta: Penerbit Andi.
- [8] Pribadi, B. 2013. Pembangunan Game Edukasi Smart Labyrinth Berbasis Dekstop (Doctoral dissertation, Universitas Komputer Indonesia).
- [9] Pratama, D., Hakim, D. A., Prasetya, Y., Febriandika, N. R., Trijati, M., & Fadlilah, U. 2016. Rancang Bangun Alat dan Aplikasi untuk Para Penyandang Tunanetra Berbasis Smartphone Android. *Khazanah Informatika: Jurnal Ilmu Komputer dan Informatika*, 2(1), 14-19.
- [10] Santrock, J.W., 1986, Life-span development (Perkembangan Masa Hidup), Penerbit Erlangga, Jakarta.
- [11] Assaf B, D., 2011. Mobile Application Testing (Best Practices to Ensure Quality). Amdocs, 2.
- [12] Testdroid. 2017. Demo Project. Diakses 07 Agustus, 2017, dari <http://cloud.testdroid.com/>.