

PENERAPAN ARTIFICIAL INTELLIGENCE PADA NON-PLAYER CHARACTER MENGGUNAKAN ALGORITMA COLLISION AVOIDANCE SYSTEM DAN RANDOM NUMBER GENERATOR PADA GAME 2D “BALAP EGRANG”

Asep Saeful Milak*¹, Eka Wahyu Hidayat², Aldy Putra Aldya³

^{1,2,3}Infomatika, Fakultas Teknik, Universitas Siliwangi
Email: ¹asep.saepul.milak14@student.unsil.ac.id, ²ekawahyu@unsil.ac.id, ³aldy@unsil.ac.id
*Penulis Korespondensi

(Naskah masuk: 20 Februari 2019, diterima untuk diterbitkan: 07 Oktober 2020)

Abstrak

Permainan Egrang merupakan salah satu permainan tradisional di Indonesia yang menggunakan bahan bambu sepanjang dua meter dan diberi pijakan dibawahnya. Permainan ini dapat dimainkan oleh semua umur dengan cara berpijak pada bagian pijakan dan menjaga keseimbangan saat berjalan dalam jarak dan waktu tertentu. Permainan ini sudah mulai tergerus zaman dan perlu dilakukan pelestarian. Penelitian ini mencoba untuk mengangkat kembali permainan tradisional kedalam *game* berbasis digital dengan menerapkan *Artificial Intelligence*. *Game* sejenis yang telah diobservasi, seperti *Game Balap Karungs* dari *Playstore* yang bergenre *racing*, memiliki kelemahan bersifat *endless run* dan juga tidak memiliki NPC (*Non-Player Character*). Hasil dari penelitian ini adalah permainan berbasis *Android* dengan menerapkan Algoritma *Collision Avoidance System* untuk membuat NPC dapat menghindari *Obstacle* dan Algoritma *Random Number Generator* untuk membuat *Obstacle* muncul secara acak. Pengembangan *Game* Balap Egrang ini menggunakan metode *Game Development Life Cycle* (GDLC). Berdasarkan pengujian yang telah dilakukan, hasil pengujian *alpha* sudah sesuai secara fungsional dan dari pengujian beta hasil uji fungsionalitas *User Acceptance Test* (UAT) didapat nilai sebesar 82,58% dinyatakan layak untuk digunakan dengan interpretasi “Baik” yang berarti *game* ini layak digunakan dan dapat dikembangkan.

Kata kunci: *Artificial Intelligence, Collision Avoidance System, Game, Obstacle, Random Number Generator*

APPLICATION OF ARTIFICIAL INTELLIGENCE IN NON-PLAYER CHARACTER USING COLLISION AVOIDANCE SYSTEM ALGORITHM AND RANDOM NUMBER GENERATOR IN 2D GAMES "EGRANG RACING"

Abstract

Egrang game is one of the traditional games in Indonesia that uses two meters of bamboo material and is given a footing below. This game can be played by all ages by stepping on the footing and maintaining balance when walking at a certain distance and time. This game has begun to erode the era and needs to be preserved. This study tries to revive traditional games into digital-based games by implementing Artificial Intelligence. Similar games that have been observed, such as the Balap Karungs Game from Playstore which is in the racing genre, have endless run weaknesses and also have no Non-Player Character NPCs. The results of this study are Android-based games by applying the Collision Avoidance System Algorithm to make NPCs avoid the Obstacle and Random Number Generator Algorithms to make the Obstacle appear randomly. The development of the Balap Egrang Game uses the Game Development Life Cycle (GDLC) method. Based on the tests that have been carried out, the alpha test results are functionally appropriate and from the beta testing the results of the User Acceptance Test (UAT) functionalities obtained a value of 82.58% which is declared feasible to use with the interpretation of "Good" which means that the game is feasible developed.

Keywords: *Artificial Intelligence, Collision Avoidance System, Game, Obstacle, Random Number Generator*

1. PENDAHULUAN

Permainan tradisional merupakan simbolisasi dari pengetahuan yang turun temurun dan mempunyai bermacam-macam fungsi atau pesan

dibaliknya, permainan tradisional merupakan hasil budaya yang besar nilainya bagi anak-anak dalam rangka berfantasi, berekreasi, berkreasi, berolahraga yang sekaligus sebagai sarana berlatih untuk hidup bermasyarakat, keterampilan, kesopanan serta

ketangkasan. Permainan tradisional dapat mengembangkan potensi setiap anak yang ditunjukkan dalam perilaku penyesuaian sosial dengan tetap melestarikan dan mencintai budaya bangsa (Andriani, 2012).

Permainan Egrang merupakan salah satu permainan tradisional di Indonesia yang memiliki berbagai nama di berbagai daerah. Permainan ini dulu sangat populer dan hampir diketahui di seluruh daerah. Permainan egrang yaitu permainan tradisional yang menggunakan bahan bambu sepanjang dua meter dan diberi pijakan 50cm dari tanah, manfaat permainan egrang yaitu menjaga keseimbangan, kerja keras, keuletan, kreatifitas, keterampilan dan kecepatan (Hikmah, 2011), kenyataan yang ada sekarang permainan tradisional yang merupakan warisan budaya kini tidak lagi diminati, sangat sedikit anak-anak yang masih mengenal ataupun memainkan permainan tradisional. Penyebab sedikitnya peminat permainan tradisional antara lain tergeser oleh permainan modern, ada mata rantai yang terputus dalam pewarisan permainan tradisional, makin terbatasnya lahan untuk bermain dan kehidupan modern yang makin individualistik (Wijaya, Purnawan, & Wibawa, 2016).

Game atau permainan adalah sesuatu yang dapat dimainkan dengan aturan tertentu sehingga ada yang menang dan ada yang kalah, bermain *game* sudah dapat dikatakan sebagai *life style* masyarakat dimasa kini, dimulai dari usia anak – anak hingga orang dewasa pun menyukai *video game*, itu semua dikarenakan bermain *video game* adalah hal yang menyenangkan (Satriawan & Alfatta, 2011).

Game berbasis kecerdasan buatan dikhususkan pada kemampuan komputer mengendalikan unsur-unsur dalam *game* membuat keputusan cerdas ketika suatu kondisi memiliki beberapa pilihan dengan hasil akhir yang berbeda, sehingga menghasilkan perilaku yang relevan, efektif, dan berguna. Kecerdasan buatan dalam permainan sangat berorientasi pada hasil, dan dengan demikian, kita dapat mengatakan bahwa dunia *game* sangat berkaitan dengan perkembangan ilmu kecerdasan buatan (Troy, 2015).

Penerapan AI (*Artificial Intelligence*) dibutuhkan dalam sebuah *game* agar *game* lebih menarik, yang biasanya terdapat pada NPC (*Non-Player Character*). NPC adalah karakter yang mampu berinteraksi dengan objek di dalam dunia *game* dan tindakannya tidak dapat dikendalikan oleh pemain, tindakan NPC yang tidak dapat diprediksi salah satunya adalah pergerakannya dari suatu tempat ketempat lain, dengan diterapkannya AI pada NPC sehingga membuat NPC mampu menemukan rute pergerakannya didalam *game* (Athariq & Putra, 2017).

Algoritma *Collision Avoidance System* adalah suatu algoritma yang berfungsi untuk melakukan suatu interaksi terhadap rintangan yang paling dekat kemudian menghindarinya yang di sisipkan kedalam *Non-Player Character* (NPC), *Collision Avoidance*

System bukan termasuk algoritma *Path Finding* yakni algoritma yang dapat membuat NPC bergerak menghindari rintangan (*Obstacle*) (Bevilacqua, 2013).

Algoritma *Random Number Generator* (RNG) adalah sebuah program atau alat untuk menghasilkan urutan angka atau simbol secara tidak teratur. Sistem ini diaplikasikan ke dalam banyak bidang, seperti sampel statistika, simulasi komputer, kriptografi, bahkan untuk desain (Hidayat, 2011).

Game bergenre *racing* sejenis seperti *Balap Karungs* dan *Egrang Run* dapat di *download* secara gratis dari *Playstore*. Hanya saja kedua *game* tersebut memiliki kekurangan yaitu bersifat *endless run*, tidak memiliki NPC (*Non Player Character*), tidak memiliki *levelling* dan tidak ada manfaat dari *reward* koin yang didapat. Selain itu, hanya *game Balap Karungs* saja yang memunculkan *Obstacle* pada saat dimainkan.

Berdasarkan latar belakang tersebut, maka dalam penelitian ini dirancang dan dibangun *Game Balap Egrang* berbasis 2D dan bergenre *racing side scroller* dengan menerapkan AI didalamnya. Penelitian ini berfokus pada pengembangan *Game* yang menerapkan Algoritma *Collision Avoidance System* dan *Random Number Generator* sebagai AI (*Artificial Intelligence*) pada NPC (*Non-Player Character*) yang berfungsi untuk menghindari rintangan (*obstacle*) yang ada.

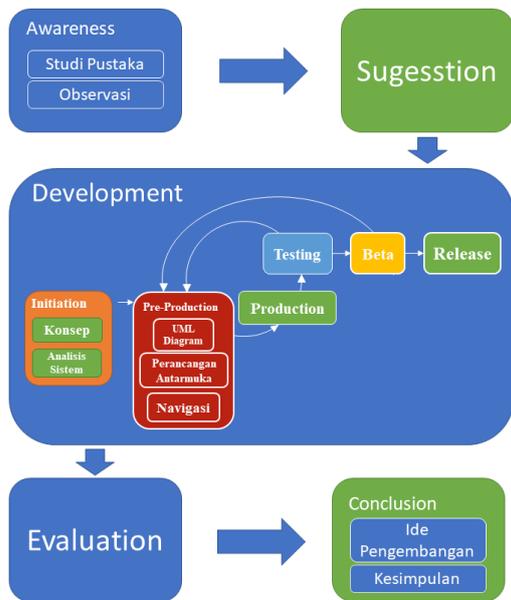
Rumusan masalah yang menjadi pokok dalam penelitian ini yaitu, bagaimana merancang dan membangun *Game* 2D “Balap Egrang” Pada *Platform Android*, bagaimana menerapkan AI (*Artificial Intelligence*) untuk NPC (*Non-Player Character*) untuk menghindari rintangan (*Obstacle*) dan munculnya *Obstacle* secara acak pada *game* 2D “Balap Egrang”, serta bagaimana menguji algoritma *Collision Avoidance System* dan *Random Number Generator* yang digunakan dalam *Game* Balap Egrang.

Tujuan yang ingin dicapai dalam penelitian ini yaitu, merancang dan membangun *game* 2D “Balap Egrang” berplatform *Android* menggunakan metode GDLC (*Game Development Life Cycle*), menerapkan AI (*Artificial Intelligence*) untuk NPC (*Non-Player Character*) menggunakan algoritma *Collision Avoidance System* dan *Random Number Generator* untuk menghindari rintangan (*Obstacle*) dan munculnya *Obstacle* secara acak pada *game* 2D “Balap Egrang” serta menguji Algoritma *Collision Avoidance System* dan *Random Number Generator* yang digunakan dalam *Game* Balap Egrang.

2. METODE PENELITIAN

Metode penelitian yang digunakan dalam penelitian ini menggunakan metode *Design and Creation* (Oates, 2006) yang terdiri dari tahapan yaitu *Awareness, Suggestion, Development, Evaluation* dan *Conclusion*. Tahap *Awareness* melakukan perumusan masalah dalam melakukan penelitian

yaitu dengan cara studi pustaka dan observasi. Tahap *Suggestion* melakukan pemberian saran untuk solusi dari permasalahan yang sudah didapat dalam tahap sebelumnya dan membuat rancangan konsep untuk menyelesaikan masalah dalam penelitian. Tahap *Development* merupakan tahap pengembangan yang dilakukan dalam Rekayasa produk multimedia dalam penelitian ini mengacu pada metodologi GDLC (*Game Development Life Cycle*) (Ramadan, 2013) metode ini terdiri dari *Initiation, Pre-Production, Production, Testing, Beta* dan *Release*. Tahap *Evaluation* melakukan evaluasi hasil dari penelitian yang sudah dilakukan sesuai dengan permasalahan dan tujuan yang diusulkan dalam penelitian. Tahap *Conclusion* melakukan masukan untuk ide pengembangan selanjutnya dari kekurangan yang sudah dibuat untuk peneliti atau pengembang selanjutnya, dan juga memberikan kesimpulan dari hasil penelitian yang sudah dilakukan. Adapun skema dari tahapan diatas digambarkan dalam diagram metode penelitian sebagai berikut:



Gambar 1. Metodologi Penelitian

2.1 Initiation

Tahapan melakukan gambaran konsep untuk pengembangan *game* “Balap Egrang” yang dikembangkan dan dibangun berdasarkan ide yang ada, untuk menentukan tujuan, mengidentifikasi pengguna, jenis produk, dasar aturan, ukuran, dan target.

2.2 Pre-Production

Tahapan ini dilakukan secara rinci mulai dari gambaran awal produk seperti apa, mulai dari arsitektur aplikasi, antarmuka aplikasi, serta elemen pendukung lainnya, tahapan ini dapat mempengaruhi atau menjadi acuan untuk tahapan selanjutnya.

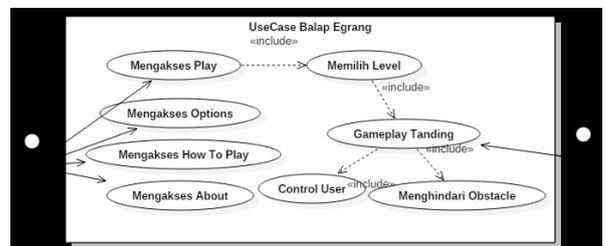
Tabel 1. Deskripsi Konsep

Konsep	Keterangan
Judul	Balap Egrang
Audiens	Umum
Durasi	Tidak terbatas
Animasi	Animasi karakter 2D (*.PNG)
Audio	Instrumen (*.MP3)
Image	Gambar, Icon, Button (*.PNG dan *.JPG)
Teks	Tulisan pada setiap menu, perintah dan petunjuk <i>game</i>
Interaktifitas	Tombol
Tema	Racing, Side Scroller
Karakter	Terdapat 3 Karakter
Deskripsi Aplikasi	Game ini merupakan game racing dengan menggunakan egrang sebagai alat yang diperlombakannya dengan harus berkompetisi dengan karakter musuh sampai garis akhir dan juga memiliki level yang berbeda dan juga tempat yang berbeda.

Tabel 2. Story Board Ringkas

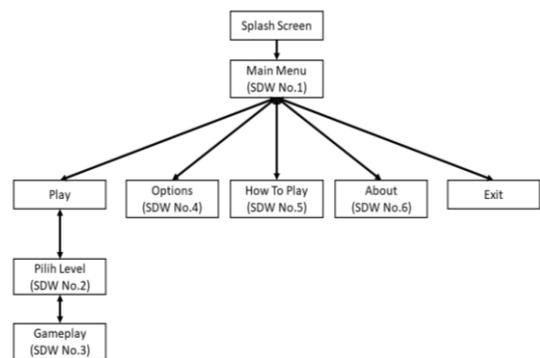
Scene	Keterangan
Scene 1	Tampilan Menu Utama
Scene 2	Tampilan Pilih Level
Scene 3	Tampilan <i>Gameplay</i> Balap Egrang
Scene 4	Tampilan Menu <i>Options</i>
Scene 5	Tampilan Informasi Cara Bermain
Scene 6	Tampilan Tentang Aplikasi

Rancangan tampilan menggunakan *Screen Design Worksheet* (SDW) yang berisi rancangan tampilan aplikasi, tombol navigasi dan keterangan tampilan dari rancangan objek yang dibuat. Sedangkan untuk menggambarkan sistem secara keseluruhan menggunakan pendekatan UML seperti gambar dibawah ini.



Gambar 2. Use Case Diagram Balap Egrang

Untuk struktur navigasi, model interaksi pengguna menggunakan Struktur Navigasi Hirarki, dimana penelusuran dilakukan mulai dari level tertinggi ke level dibawahnya secara hirarki.



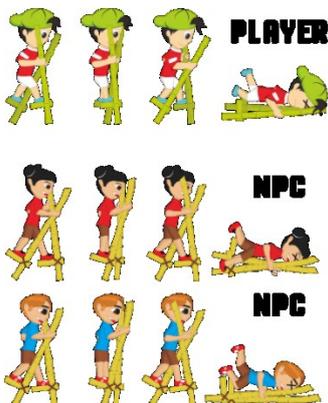
Gambar 3. Struktur Navigasi Aplikasi

2.3 Production

Tahapan ini merupakan tahapan inti dari proses pengembangan *Game* yang meliputi pembuatan objek, kode program dan penggabungan keduanya menjadi sebuah produk multimedia berupa *Game* “Balap Egrang”.

1. Objek 2 Dimensi

Karakter 2 Dimensi digunakan sebagai objek Player dan juga NPC yang terdiri dari 1 objek untuk Player dan 2 objek untuk NPC, karakter terdiri dari beberapa gerakan dalam 1 gambar dengan format file *.PNG yang nantinya digabungkan untuk pembuatan animasi tiap karakter.



Gambar 4. Karakter yang digunakan untuk aplikasi

2. Pembuatan Sound Effect

Pembuatan dan juga pengumpulan *sound effect* difungsikan untuk menambahkan suara pada *game* agar terlihat lebih menarik, dibuat dengan menggunakan aplikasi FL Studio dan juga mengambil beberapa asset dari internet yang berlisensi gratis, suara ini digunakan pada setiap menu yang ada pada *game* “Balap Egrang” seperti menu utama, *options*, *how to play*, *about*, pilih *level* dan *gameplay*.

3. Pembuatan User Interface

Pembuatan *user interface* ini ditunjukkan untuk mengimplementasikan rancangan antar muka yang diterapkan pada aplikasi yang dibuat, *design user interface* menggunakan aplikasi pengolah gambar yaitu *Adobe Photoshop CC 2014*.

4. Pembuatan Kode Program

Pembuatan kode program untuk pembuatan *game* “Balap Egrang” dengan menggunakan Bahasa pemrograman C#, proses penulisan kode program menggunakan aplikasi *Visual Studio 2017*, terdapat beberapa program yang dibuat mulai dari perpindahan layer, *control* karakter, pergerakan NPC, *unlock level*, kode untuk menghindari rintangan, kode program posisi.

Implementasi algoritma *Collision Avoidance System* diimplementasikan dalam sebuah program yang dimasukkan kedalam karakter NPC yang berfungsi untuk mencari *Obstacle* yang ada di jalurnya kemudian menghindari *Obstacle* tersebut, proses penentuan karakter NPC untuk menghindari

Obstacle menggunakan parameter *Collider* yaitu sebuah batasan antara objek agar tidak terjadi tubrukan.

Tabel 3. Hasil Analisis Algoritma *Collision Avoidance System*

Program	Analisis
<pre>npcTransform.position = Vector3.MoveTowards(npcTransfor m.position, Waypoint.position, 0.1f); Timing += Time.deltaTime; if (Timing >= timeList[Urutan]) { Urutan = Random.Range(0, 3); Timing = 0; Penentuan(); } void Penentuan() { if(Obstacle == false) { Lurus(); } else { Miring(); } } public void Miring() { if (Atas == false) { Waypoint.position = new Vector3(Waypoint.position.x + 1.5f, Waypoint.position.y + 2, Waypoint.position.z); Gambar(); Timing = 1; Atas = true; } else { Waypoint.position = new Vector3(Waypoint.position.x + 1.5f, Waypoint.position.y - 2, Waypoint.position.z); Gambar(); Timing = 1; Atas = false; } }</pre>	<p>Program penentuan pergerakan NPC mulai dari kecepatan gerak dan proses penentuan untuk pencarian <i>Obstacle</i> (Penentuan).</p> <p>Program penentuan untuk menghindari <i>Obstacle</i> dengan parameter <i>Collider</i>, jika didepan tidak ada <i>Obstacle</i> maka berjalan lurus sedangkan kalo ada <i>Obstacle</i> berjalan miring atau menghindari <i>Obstacle</i>.</p> <p>Program penentuan pergerakan NPC untuk menghindari <i>Obstacle</i>, jika NPC berada diatas (Atas = true) maka ketika ada <i>Obstacle</i> dijalurnya NPC akan menghindar ke bawah dengan perubahan posisi x dan y, sedangkan kalau NPC berada di bawah (Atas=false) maka ketika ada <i>Obstacle</i> NPC akan menghindari ke atas dengan perubahan posisi x dan y.</p>

Implementasi Algoritma *Random Number Generator* diimplementasikan dalam sebuah kode program yang berfungsi untuk menampilkan *Obstacle* secara acak pada setiap *level* yang dimainkan dalam *game* “Balap Egrang”, kode program menggunakan parameter *Random Range* yakni pengacakan pada posisi tertentu yang sudah ditentukan sebelumnya.

Tabel 4. Hasil Analisis Algoritma *Random Number Generator*

Program	Analisis
<pre>void Start () { Urutan = Random.Range(0, 3); transform.position = new Vector3(transform.position.x, PosisiY[Urutan], transform.position.z); }</pre>	<p>Program penentuan munculnya <i>Obstacle</i> dengan <i>Random Range</i> dan perubahan penempatan pada posisi Y sesuai urutan antara 1-3.</p>

5. Pembuatan Game

Tahapan penggabungan semua elemen multimedia seperti teks, gambar, suara dan animasi yang sudah dibuat sebelumnya, semua elemen tersebut digabungkan kedalam *Software Unity* untuk tahap pembuatan *Game* Balap Egrang.

3. HASIL DAN PEMBAHASAN

3.1 Antarmuka Game Balap Egrang

Hasil dari pembuatan *game* yang telah dilakukan pada tahapan *Production* menghasilkan antarmuka sebagai berikut:



Gambar 5. Menu Utama



Gambar 6. Tampilan *Gameplay*



Gambar 7. Tampilan Menu Pilih Level



Gambar 8. Tampilan Menu *About*

3.2 *Alpha Testing*

Tahapan ini merupakan tahap selanjutnya setelah *production* dilakukan yaitu tahap *testing* atau pengujian *game* yang sudah dibuat untuk melihat kemungkinan adanya kesalahan, tahap ini dilakukan dengan metode *Alpha test* untuk menguji aplikasi secara fungsional dan pengujian algoritma,

dilanjutkan dengan *Beta test* untuk mendapatkan *feedback* dari pengguna.

Tahapan Pengujian *Alpha Test* menggunakan metode *black-box testing*. Metode *black-box* digunakan untuk mengetahui apakah *game* berfungsi dengan benar, pengujian fungsional dilakukan untuk menguji tampilan Menu Utama, Pilih Level, *Gameplay*, Menu *Options*, Menu *How to play* dan Menu *About*. Hasil dari pengujian tersebut secara fungsi sudah sesuai dan dapat diterima. Untuk pengujian Menu Utama dapat dilihat pada 989able berikut ini:

Tabel 5. Hasil Pengujian Menu Utama

No	Skenario Pengujian	Hasil yang diharapkan	Respon Aplikasi
1	Tombol navigasi Permainan Utama	Menampilkan permainan utama dari <i>game</i> "Balap Egrang"	Perpindahan <i>scene</i> dari menu utama ke menu <i>gameplay</i> berhasil berjalan dengan lancar
2	Tombol navigasi <i>Options</i>	Menampilkan pengaturan untuk mengubah suara dalam <i>game</i>	Perpindahan <i>scene</i> dari menu utama ke menu <i>Options</i> berhasil berjalan dengan lancar
3	Tombol navigasi <i>How to play</i>	Menampilkan tutorial cara bermain <i>game</i> "Balap Egrang"	Perpindahan <i>scene</i> dari menu utama ke menu <i>How to Play</i> berhasil berjalan dengan lancar
4	Tombol navigasi <i>About</i>	Menampilkan informasi tentang <i>game</i>	Perpindahan <i>scene</i> dari menu utama ke menu <i>About</i> berhasil berjalan dengan lancar
5	Tombol navigasi Pilih Level	Menampilkan level yang tersedia dalam <i>game</i> Balap Egrang	Perpindahan <i>scene</i> dari Pilih level ke <i>scene</i> level yang dipilih berhasil berjalan dengan lancar
6	Judul <i>Game</i>	Menampilkan judul <i>game</i> "Balap Egrang"	Informasi judul berhasil ditampilkan dengan baik.

Selanjutnya melakukan pengujian Algoritma *Collision Avoidance System*. Pengujian Algoritma ini dilakukan untuk mengetahui hasil apakah algoritma yang dipakai pada NPC bisa melewati *Obstacle* yang berada pada lintasan, pengujian ini melakukan metode analisis kualitatif dengan membandingkan NPC 1 dan NPC 2 dalam melewati *Obstacle* ketika dalam berbeda jalur dengan jalur yg sama antara NPC, dengan Hasil yang ada pada Tabel 6.

Hasil dari tabel diatas dilakukan beberapa percobaan didalam *Game* untuk menguji Algoritma *Collision Avoidance System* yang diterapkan pada Kedua NPC yang awalnya mempunyai jalur masing-masing pada lintasan, hasil dari percobaan bahwa Algoritma berjalan dengan baik dalam menghindari *Obstacle*, tapi ada percobaan yang membuat NPC tidak menghindar atau bahkan menembus *Obstacle* yakni ketika NPC1 dan NPC2 berada dalam jalur yang sama dan saling rapat maka salah satu saja yang dapat menghindar sedangkan yang satu lagi akan menembus NPC, hal tersebut disebabkan karena

Algoritma yang dipakai menggunakan Parameter *Collider* jadi algoritma tidak bisa membaca 2 *Collider* secara bersamaan.

Tabel 6. Pengujian Algoritma *Collision Avoidance System*

Percobaan Ke-	NPC 1		NPC 2	
	Posisi NPC 2	Status	Posisi NPC 1	Status
1.1	Beda Jalur	Ke bawah	Beda Jalur	Tidak Ada <i>Obstacle</i>
1.2	Satu Jalur	Ke atas	Satu Jalur	Menembus <i>Obstacle</i>
1.3	Beda Jalur	Ke bawah	Beda Jalur	Tidak Ada <i>Obstacle</i>
2.1	Beda Jalur	Ke Bawah	Beda Jalur	Tidak Ada <i>Obstacle</i>
2.2	Satu Jalur	Ke atas	Satu Jalur	Menembus <i>Obstacle</i>
2.3	Beda Jalur	Ke bawah	Beda Jalur	Tidak Ada <i>Obstacle</i>
3.1	Beda Jalur	Ke Bawah	Beda Jalur	Tidak Ada <i>Obstacle</i>
3.2	Satu Jalur	Ke atas	Satu Jalur	ke atas
4.1	Beda Jalur	Tidak Ada <i>Obstacle</i>	Beda Jalur	ke Atas
4.2	Satu Jalur	Menembus <i>Obstacle</i>	Satu Jalur	Menembus <i>Obstacle</i>
4.3	Satu Jalur	Ke Bawah	Satu Jalur	ke bawah
4.4	Satu Jalur	Ke Atas	Satu Jalur	ke Atas
5.1	Beda Jalur	ke bawah	Beda Jalur	Tidak Ada <i>Obstacle</i>
5.2	Satu Jalur	Ke Atas	Satu Jalur	Menembus <i>Obstacle</i>
5.3	Beda Jalur	Tidak Ada <i>Obstacle</i>	Beda Jalur	Ke atas

Kesimpulannya Algoritma *Collision Avoidance System* berhasil diterapkan pada NPC dan berhasil menghindari *Obstacle* dengan catatan tiap NPC tidak saling berdempetan.

Tahapan terakhir pengujian adalah pengujian Algoritma *Random Number Generator*. Pengujian Algoritma ini dilakukan untuk mengetahui hasil apakah algoritma yang dipakai berhasil membuat posisi *Obstacle* secara acak, pengujian ini dilakukan dalam *game* langsung untuk mengetahui Algoritma bekerja dengan baik atau tidak, algoritma ini memiliki parameter perubahan posisi Y dari 0 sampai 4 dengan menggunakan *Random Range*, hasil dari pengujian terdapat pada tabel 7 sebagai berikut:

Tabel 7. Pengujian Algoritma *Random Generator Number*

Percobaan Ke-	Posisi <i>Obstacle</i> (Posisi Y)							
	1	2	3	4	5	6	7	8
1	0	2	2	0	0	0	0	4
2	2	4	4	0	0	4	0	4
3	2	0	2	4	0	2	0	2
4	0	0	0	4	4	2	2	2
5	0	2	2	2	2	0	0	4
6	4	4	2	4	2	4	0	0

Hasil dari tabel pengujian diatas pengujian dilakukan pada 8 *Obstacle* yang ada pada tiap lintasan dan dilakukan percobaan sebanyak 6 kali, menunjukkan bahwa *Obstacle* akan muncul secara

acak pada setiap percobaan. Kesimpulannya bahwa Algoritma *Random Number Generator* berhasil diterapkan pada *Obstacle* yang membuat muncul secara acak.

3.3 Beta Testing

Pengujian Beta ini dilakukan pengambilan data dengan menggunakan metode survey berupa pengisian kuisioner kepada pengguna akhir. Tujuan dari penyebaran kuisioner ini untuk mendapatkan umpan balik dari pengguna yang memainkan game yang dibuat, kuisioner menggunakan pendekatan VISUALS dengan perhitungan menggunakan metode Skala Likert (Sugiyono, 2013). Rancangan pertanyaan-pertanyaan pada kuisioner dengan pendekatan VISUALS, Materi-materi yang terkumpul disajikan dengan bahasa yang sederhana dan komunikatif dilengkapi dengan komponen-komponen multimedia lainnya dengan memperhatikan konsep VISUALS (Hidayat & Irawan, 2013), yaitu, *Visible* atau mudah dilihat, yaitu materi yang disajikan secara visual terlihat jelas, tingkat keterbacaan tinggi, resolusi atau ketajaman grafis tinggi, mengandung satu makna. *Interesting* atau menarik, yaitu isi pesan sesuai dengan kebutuhan audien, tampilan baik dan memikat sehingga menimbulkan rasa ingin tahu, dan berusaha menjaga kelangsungan proses komunikasi atau interaksi dan belajar. *Simple* atau sederhana, yaitu pesan terfokus, pemilihan kata, huruf, gambar tidak mengubah makna pesan, bahasa dan tampilan lugas. *Useful* atau berguna, yaitu sesuai dengan kebutuhan audien dan tujuan pembelajaran maupun hasil belajar yang diinginkan. *Accurate* atau tepat, yaitu isi pesan mempunyai makna yang tepat, sesuai dengan kebutuhan, penyampaian cermat, didasarkan pada sumber yang dapat di pertanggung jawabkan. *Legitimate* atau benar, yaitu isi pesan benar, disusun secara logis, mengikuti kaidah keilmuan, dan masuk akal. *Structure* atau terstruktur, yaitu rangkaian pesan disampaikan secara sistematis, dengan urutan-urutan yang logis dan mudah dipahami.

Proses *beta testing* dengan kuisioner yang disebar kepada responden, populasi yang diambil adalah orang yang pernah atau giat dalam bermain *game*, sedangkan sampel yang telah diambil berjumlah 30 orang, Kemudian persentase dari masing-masing jawaban kuisioner diidentifikasi dengan menggunakan rumus Skala Likert (Sugiyono, 2013) sebagai berikut:

Rumus Interval

$$Hasil = T \times Px \dots\dots(1)$$

$$Index = \frac{Jumlah}{Y} \times 100 \dots\dots(2)$$

$$I = 100 / Jumlah\ skor\ (likert)$$

(1)

$$I = 100 / 3$$

$$I = 33,33\%$$

Keterangan:

T = hasil responden
 Px = jumlah skor likert
 Y = skor tertinggi likert

Total responden yang mengisi kuesioner sebanyak 30 orang. Berikut adalah persentase hasil pengolahan kuesioner seperti pada Tabel 8.

Tabel 8. Hasil Kuesioner

Aspek	Soal	B	C	K	Skor	Presentase
Visible	1	81	6	0	87	96,67%
	2	63	18	0	81	90,00%
	3	45	30	0	75	83,33%
	4	57	22	0	79	87,78%
Interesting	5	42	32	0	74	82,22%
	6	54	24	0	78	86,67%
	7	66	16	0	82	91,11%
Simple	8	54	24	0	78	86,67%
	9	81	6	0	87	96,67%
	10	48	28	0	76	84,44%
Useful	11	69	14	0	83	92,22%
	12	42	32	0	74	82,22%
Accurate	13	48	28	0	76	84,44%
	14	36	36	0	72	80,00%
	15	36	36	0	72	80,00%
	16	24	42	1	67	74,44%
Legitimate	17	63	18	0	81	90,00%
	18	72	12	0	84	93,33%
	19	36	36	0	72	80,00%
	20	72	12	0	84	93,33%
Structure	21	60	20	0	80	88,89%
	22	60	20	0	80	88,89%
Jumlah		Skor : 1635/22= 74,31			1816,67%/22=	82,58%

Ket: B=Baik C=Cukup K=Kurang

Tabel 9. Nilai Interval Persentase Skala likert

Persentase	Kriteria
0% - 33,32%	Kurang
33,33% - 66,65%	Cukup
66,66% - 100%	Baik

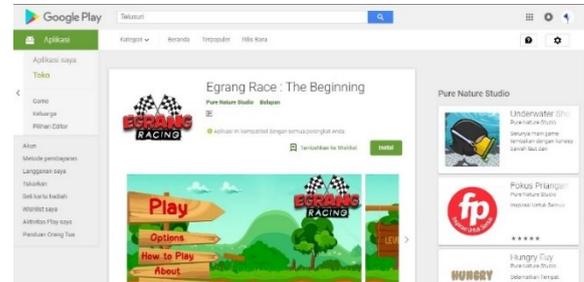
Kesimpulan yang bisa diambil hasil dari kuisisioner menunjukkan kategori tingkat validasi diatas, maka dapat disimpulkan bahwa *game* ini memiliki interpretasi 'Baik', sehingga dapat dikatakan *game* ini layak untuk digunakan.

Aplikasi ini memiliki kelebihan dan kekurangan Kelebihan aplikasi "Balap Egrang" yaitu Konsep *Game* mengacu pada permainan tradisional yaitu Egrang, Penambahan NPC pada *gameplay* membuat Balap Egrang lebih realistis, Penambahan *System Leveling* membuat tingkatan bermain lebih susah dan tidak monoton. Terdapat rintangan (*Obstacle*) yang muncul secara acak yang menambah tingkat kesulitan *game*. Sedangkan kekurangan dari aplikasi "Balap Egrang" yaitu *Game* tidak bisa dimainkan secara *multiplayer*, *Game* tidak bisa berjalan dibawah OS Android Jelly Bean, *Game* hanya tersedia 3 *Level*, NPC tidak dapat melewati *Obstacle* ketika NPC menempel dengan NPC lain.

3.4 Release

Tahapan ini merupakan tahapan terakhir dalam proses pembuatan produk multimedia, hasil pada

penelitian ini menghasilkan sebuah produk multimedia yakni berupa *game* yang bergenre *racing* yaitu "Balap Egrang" (*Egrang Race: The Beginning*). Produk ini didistribusikan pada *Playstore* dan bisa di unduh secara gratis.



Gambar 9. *Game* Egrang di *Playstore*

4. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan dapat diambil kesimpulan yaitu, telah berhasil dirancang dan dibangun *Game* Balap Egrang dengan menggunakan Metode GDLC (*Game Development Life Cycle*) yang dapat berjalan diperangkat *smartphone Android*, yang memberikan pengalaman baru dalam bermain *game Racing* karena mengangkat permainan tradisional yaitu Egrang dengan menerapkan Algoritma *Collision Avoidance System* pada AI NPC untuk menghindari *Obstacle* dan Algoritma *Random Number Generator* untuk memunculkan rintangan secara acak. Hasil pengujian terhadap Algoritma *Collision Avoidance System* yang diterapkan berhasil membuat NPC menghindari *Obstacle* dan hasil pengujian terhadap Algoritma *Random Number Generator* berhasil membuat *Obstacle* muncul secara acak.

Saran untuk aplikasi ini yaitu Aplikasi lebih menarik apabila ditambahkan lebih banyak karakter dan ditambah fitur kostumisasi karakter, seperti perubahan tampilan Egrang nya, Menambahkan level dan tingkatan kesulitan dalam *Game* Balap Egrang, Karakter dalam *Game* diubah menjadi 3 Dimensi (3D) agar lebih realistis, dapat dimainkan secara *multiplayer*, dapat menyempurnakan pergerakan NPC dalam menghindari *Obstacle* dengan algoritma A* (A Star).

DAFTAR PUSTAKA

- ANDRIANI, T., 2012. Permainan Tradisional Dalam Membentuk Karakter Anak Usia Dini. *Jurnal Sosial Budaya*, Vol. 9 No. 1.
- ATTHARIQ, A., DAN PUTRA, D.A., 2017. Penentuan Pergerakan *Non-Player Character* Menggunakan Algoritma A* Pada *Game Action- Role-Playing Game*. *Jurnal Infomedia* Vol. 2 No. 2 Desember 2017, pp.35-40.
- BEVILACQUA, F., 2013. *Game Development Programming: Understanding Steering Behaviors: Collision Avoidance*.

<https://gamedevelopment.tutsplus.com/tutorials/understanding-steering-behaviors-collision-avoidance--gamedev-7777>,
[Diakses : 12 Januari 2019]

- HIDAYAT, E.W., DAN IRAWAN, E. P., 2013. *Prototype Informasi Digital Jurusan Teknik Informatika UNSIL Berbasis Multimedia*. STMIK BUMIGORA MATARAM : ISBN : 978-602-17488-0-0.
- HIDAYAT, T.R., 2011. *Random Number Generator*. Makalah IF2091 Struktur Diskrit, pp.1-5.
- HIKMAH, M., 2011 Perancangan Game Tradisional Jawa Egrang Bambu. Mulok Perpustakaan UM, pp.1-3.
- OATES, B. J., 2006. *Researching Information Systems and Computing*, SAGE Publications, London, ISBN:1-4129-0223-1, pp.108-124.
- RAMADAN, R., DAN WIDYANI, R., 2013. *Game Development Life Cycle Guidelines*. ICAC SIS 2013, ISBN: 978-979-1421-19-5, pp.95-100.
- SATRIAWAN, A., DAN FATTA, H.A., 2011. *Game Edukasi Baby Lost in The Jungle* dengan Adobe Flash Cs3. *Jurnal Data Manajemen dan Teknologi Informasi*, Vol. 12 No. 4 Desember 2011, pp.42-45.
- SUGIYONO, S., 2013. *Metode Penelitian Pendidikan Pendekatan Kuantitatif, Kualitatif, dan R&D*. Bandung: Alfabeta.
- TROY, T., 2015. Tinjauan Historis Kecerdasan Buatan Dalam *Games*. *Journal of Animation and Games Studies* Vol. 1 No. 2 Oktober 2015, pp.135-164.
- WIJAYA, A.A.M.A., PURNAWAN, I.K.A. & WIBAWA, K.S., 2016. Rancang Bangun *Game Tajog Race* Berbasis Android. *Merpati* Vol. 4 No. 2 Agustus 2016, ISSN: 2252-3006, pp.166-177.