

IMPLEMENTASI RULE-BASED METHOD UNTUK NPC PADA PERMAINAN RACING CAR IMPLEMENTATION OF RULE-BASED METHOD FOR NPC ON A RACING CAR GAME

M Sultan Aldo¹, Purba Daru Kusuma², Ratna Astuti Nugrahaeni³

^{1,2,3} Universitas Telkom, Bandung

sitnaldo@student.telkomuniversity.ac.id¹, purbodaru@telkomuniversity.ac.id², ratnaan@telkomuniversity.ac.id³

Abstrak

Pada setiap *game* di komputer ataupun konsol, terdapat *Non-Player Character* (NPC). Keberadaan NPC bertujuan untuk membuat *game* agar terlihat lebih hidup, khususnya untuk *game* yang berfokus pada *game offline* atau *single-player*. Terdapat beberapa situasi dimana NPC pada *game* perlu diperbarui. Sebagai contoh, misalnya ketika ekspansi konten baru dalam sebuah *game* balapan terjadi dan terdapat sebuah sirkuit baru yang akan dimuat ke dalam *game* tersebut. Untuk mempermudah proses pembaruan serta melakukan pengembangan terhadap NPC, penulis mengembangkan NPC berbasis kecerdasan buatan dengan metode *rule-based* (RB). Berdasarkan pengujian performansi dengan menggunakan metode RB, diperoleh bahwa NPC dapat melaju dengan baik pada lingkungan yang telah dikembangkan oleh penulis. Dengan pengujian pada 2 jenis lingkungan yang telah dikembangkan, pada arena 1 didapatkan hasil dimana NPC dari 10 kali pertandingan dengan 6 orang *player* dapat memenangkan 50% pertandingan dengan parameter *reward* sebesar 3.5, lalu pada pengujian di arena 2 didapatkan hasil dimana NPC mendapatkan persentase kemenangan 23,3% dari 5 kali pertandingan. Sedangkan pada pengujian performansi tanpa pengaplikasian metode RB terhadap NPC pada lingkungan arena 1 didapatkan hasil dimana NPC dari 10 kali pertandingan memenangkan 26,67% pertandingan melawan *player* dan pada arena 2 didapati NPC kesulitan untuk dapat bersaing melawan *player* yang menjadikan mereka memiliki 0% kemenangan dari 5 kali total pertandingan.

Kata Kunci: Kecerdasan Buatan, *Rule-based*, NPC, *Game Development*.

Abstract

In every game on a computer or console, there is a Non-Player Character (NPC). The existence of NPCs aims to make the game look more alive, especially for games that focus on offline-gameplay or single-player games. There are some situations where the NPCs in the game need to be updated. For example, when a new content expansion in a racing game occurs and a new circuit is loaded into the game. To simplify the process of updating and developing NPCs, the authors developed artificial intelligence-based NPCs using the rule-based (RB) method. Based on testing using the RB method, it was found that NPCs could run well in an environment that had been developed by the author. By testing on 2 types of environments that have been developed, in arena 1 the results are obtained where NPCs from 10 matches with 6 players can win 50% of matches with a reward parameter of 3.5, then in the test in arena 2 the results were obtained where the NPC got a winning percentage of 23.3% from 5 matches. Whereas in performance testing without the application of the RB method to NPCs in arena 1, the results obtained where NPCs from 10 matches won 26.67% of matches against players and in arena 2 it was found that NPCs had difficulty competing against players which made them have 0% wins from a total of 5 matches.

Keywords: *Artificial Intelligence, Rule-based, NPC, Game Development.*

1. Pendahuluan

Game adalah sebuah istilah lain dari kata permainan. Game engine yang semakin canggih kaya akan fitur-fitur, dan mudah digunakan, membuat pengembang game pemula dapat membuat sebuah game dengan mudah, bahkan tanpa perlu memiliki keahlian programming. Rule-based AI adalah metode yang paling banyak digunakan untuk game berbasis agen [9]. Aturan, terutama statement bersyarat, disimpan dalam kelompok yang tidak berurutan. Aturan dicocokkan dengan fakta yang ada. Kumpulan pengetahuan dan fakta adalah satu set dinamis, terus-menerus disesuaikan [9]. Rule-based systems (RB) berada di garda depan penelitian AI selama tahun 1970 dan awal 1980. Banyak program AI yang paling terkenal dikembangkan dengan metode tersebut dan dalam inkarnasi "expert systems" mereka adalah teknik AI yang paling terkenal [11].

Salah satu permasalahan dalam mengembangkan sebuah NPC pada suatu *game*, khususnya pada *game* balapan, adalah penambahan kode secara kontinyu. Penambahan kode berguna supaya NPC melaju dengan baik pada sirkuit atau lingkungan baru. Meskipun wajar, implementasi kecerdasan buatan kepada NPC dapat mempermudah pengembangan *game* serta NPC dengan cara yang lebih praktis dan efisien. Oleh karena itu, pada penelitian ini, metode RB ditambahkan untuk mengembangkan NPC yang dapat melaju dengan baik dan dapat menyaingi *player* pada saat pertandingan di lingkungan arena yang telah dikembangkan oleh penulis.

2. Landasan Teori

2.1 Teori Pendukung

Sports Games merupakan salah satu genre game yang sudah berkembang lama. Genre ini dikenal dengan genre game yang memiliki model *fluid gameplay* [3]. Beberapa contoh genre *sports games* antara lain: permainan sepak bola, permainan hoki, atau bola basket [3]. Gaya permainan *sport game* terbilang cepat, sangat dinamis, dan juga berlanjut untuk kurun waktu yang lama dengan sedikit atau tanpa waktu berhenti [3]. Genre balapan sangatlah menarik, baik dari sudut pandang *gameplay* maupun dari sudut pandang AI. Genre ini sebagian besar dibagi menjadi dua kelompok utama, yaitu kendaraan dan spesialisasi [3, 10]. Pada pengelompokan tersebut memang memiliki relasi benang merah, yang setidaknya mempunyai kemiripan dengan simulasi balapan yang berbasis hukum fisika. Pada beberapa *game* balapan (bahkan yang modern) memang mengambil kebebasan dengan bagaimana hukum fisika berkerja terhadap sistem mereka, tetapi itulah *video game*, berpegang pada area realitas yang kita tidak keberatan dibatasi olehnya, dan menghilangkan bagian dari realitas yang kita lakukan [3, 10].

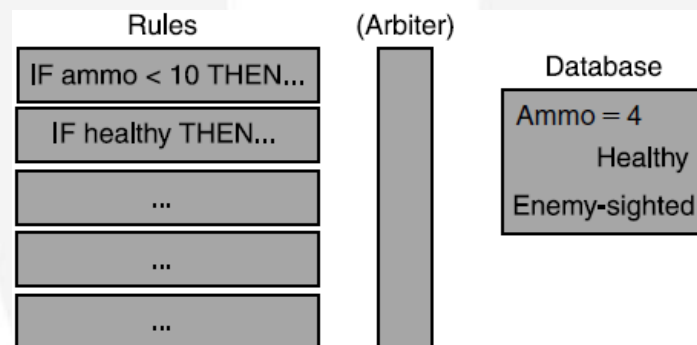
2.2 Artificial Intelligence pada Game

Game AI (Game Artificial Intelligence) merupakan sebuah program di dalam suatu *game* dimana membuat lawan yang dikendalikan oleh komputer atau *Non-Player Character (NPC)* tampak membuat keputusan yang cerdas ketika di dalam *game* memiliki banyak pilihan untuk situasi tertentu yang menghasilkan perilaku yang relevan, efektif, dan bermanfaat [3, 7]. Pada kecerdasan buatan terdapat berbagai macam metode, salah satunya adalah metode yang akan dipakai pada penelitian ini, yaitu *Rule Based Method (RB)*.

2.3 Rule-Based Systems (RB)

Rule Based Systems (RB) memecahkan masalah dengan aturan yang berasal dari pengetahuan ahli. Aturan memiliki bagian kondisi dan tindakan, kondisi *if-then* dan diumpkankan ke mesin inferensi, yang memiliki memori kerja informasi tentang masalah, pencocokan pola, dan penerapan aturan. Pencocokan pola mengacu pada memori kerja untuk memutuskan aturan mana yang relevan, kemudian pembuat aturan memilih aturan apa yang akan diterapkan. Informasi baru yang dibuat oleh tindakan bagian dari aturan yang diterapkan ditambahkan ke memori kerja dan siklus pencocokan-pilih-tindakan antara memori kerja dan basis pengetahuan diulang sampai tidak ada lagi aturan yang relevan ditemukan [6, 8]. Sistem RB memiliki struktur umum yang terdiri dari dua bagian: database berisi pengetahuan yang tersedia untuk AI dan seperangkat aturan *if-then* [11].

Aturan dapat memeriksa database untuk menentukan apakah kondisi "if" mereka terpenuhi. Aturan yang kondisinya terpenuhi dikatakan memicu. Aturan yang dipicu dapat dipilih untuk diaktifkan, di mana komponen "then"-nya dieksekusi [11].



Gambar 1 Skema Rule-Based Systems

2.4 Teori Pengukuran dan Pengujian R-Validation

Dalam penelitian, baik berbentuk kualitatif maupun kuantitatif, kriteria utama yang harus diperhatikan adalah valid dan reliabel. Validitas adalah derajat ketepatan antara data yang terdapat di lapangan dan data yang dilaporkan oleh peneliti. Reliabilitas adalah keandalan alat ukur. Pada tahap ini pengujian *usability testing* dilakukan untuk mengevaluasi sebuah sistem berdasarkan penilaian pengguna dengan kuesioner USE (*Usefulness, Satisfaction, Ease of Use*). Persamaan *usability testing* dengan menggunakan model skala *likert* adalah sebagai berikut [12]:

$$rumus\ index\ \% = \frac{total\ skor}{Y \times 100} \quad (1)$$

Dalam penelitian, baik berbentuk kualitatif maupun kuantitatif, kriteria utama yang harus diperhatikan adalah valid dan reliabel. Validitas adalah derajat ketepatan antara data yang terdapat di lapangan dan data yang dilaporkan oleh peneliti. Reliabilitas adalah keandalan dari alat ukur tersebut. Untuk mengukur pertanyaan pada kuesioner telah valid atau tidak maka dapat digunakan pembuktian $r_{hitung} > r_{banding}$. Nilai $r_{banding}$ dapat diperoleh dari jumlah responden, sedangkan r_{hitung} diperoleh menggunakan persamaan sebagai berikut:

$$r = \frac{n \sum XY - (\sum X)(\sum Y)}{\sqrt{(n \sum X^2 - (\sum X)^2)(n \sum Y^2 - (\sum Y)^2)}} \quad (2)$$

Dimana:

N: Jumlah Responden
 X: Jumlah nilai soal per responden
 Y: Jumlah nilai total per responden

Untuk mengukur apakah setiap pertanyaan dari kuesioner telah dibuat dengan reliabel atau tidak dapat menggunakan pembuktian $r_{hitung} > r_{banding}$ kembali. Nilai r_{hitung} didapatkan menggunakan persamaan sebagai berikut:

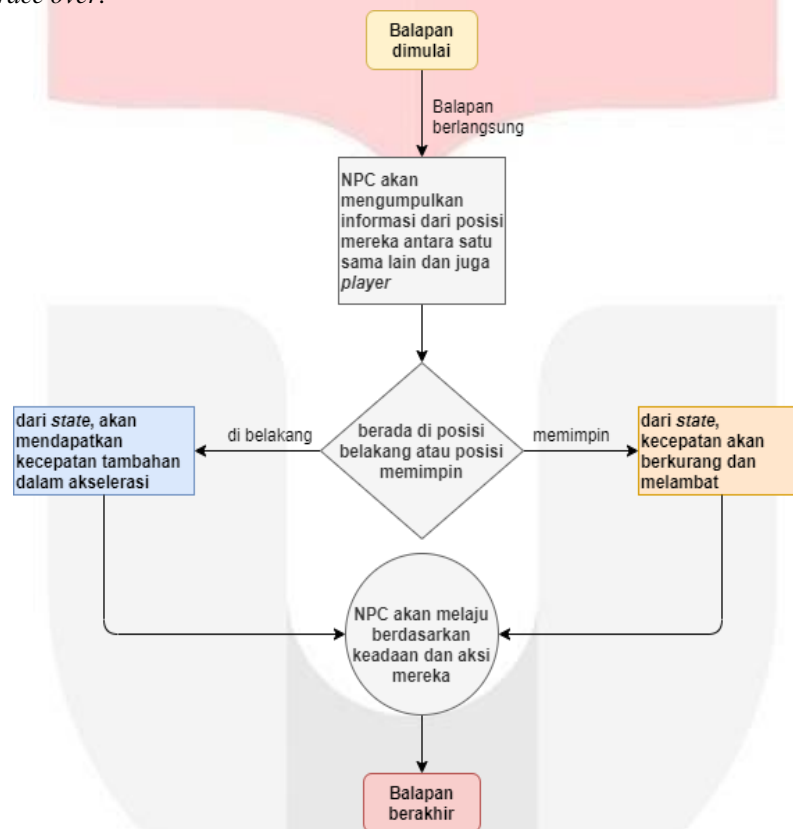
Dimana:
$$r_{hitung} = \left[\frac{n}{n-1} \right] \left[1 - \frac{\sum i}{j} \right]$$
 (3)

n: Jumlah pertanyaan
 i: Varian item
 j: Varian total

3. Pembahasan

3.1 Gambaran Umum Sistem

NPC dalam game balapan ini adalah mobil-mobil balap selain mobil balap *player*. Proses yang terjadi meliputi *race begin*, *stating*, dan *race over*.



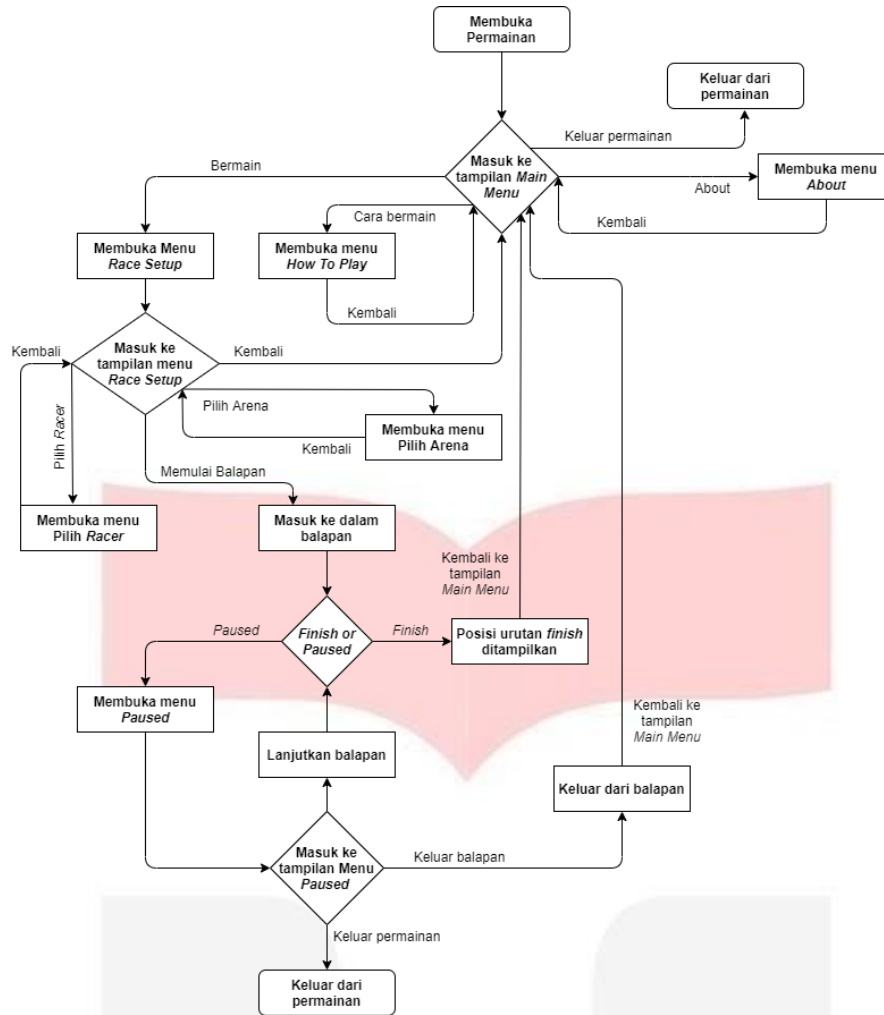
Gambar 2 Gambaran Umum Sistem

Pada awal balapan NPC akan mengumpulkan informasi dari keadaan lingkungan sekitar sehingga nantinya dia dapat mengambil sebuah aksi berdasarkan informasi tersebut. Dan hasil yang diinginkan nantinya NPC dapat bersaing melawan *player* dengan lebih seimbang.

3.2 Perancangan Sistem Game

3.2.1. Flowchart

Flowchart sistem di bawah dirancang untuk mengetahui serta memahami bagaimana alur sistem dari *game* ini bekerja.

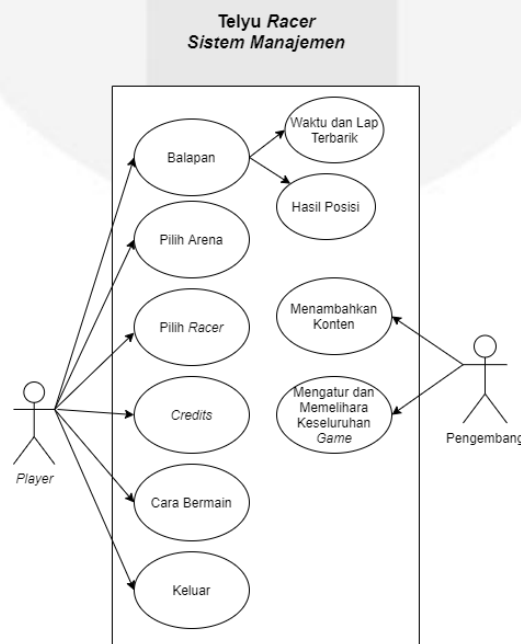


Gambar 3 Flowchart Sistem

Pada Gambar 3 dapat dilihat bahwa terdapat menu-menu utama yang terdapat dalam sistem game, seperti *Main Menu*, *Race Setup Menu*, *Racer Select Menu*, *Arena Select Menu*, *Paused Menu*, dan juga *Result Menu*.

3.2.1. Use Case Diagram

Use case diagram berikut dibuat untuk memberikan informasi untuk memahami apa saja yang dapat dilakukan oleh para aktor dimana pada Gambar 4 di bawah ini terdapat 2 aktor yaitu *player* dan pengembang.



Gambar 4 Use Case Sistem

Di atas merupakan *use case diagram* dari game yang telah dibuat, dimana bila sebagai *player* kita diberikan akses

untuk dapat bermain balapan, lalu kita juga dapat melakukan pemilihan terhadap arena dan juga mobil yang ingin digunakan serta menu tambahan seperti *credits*, *how to play*, dan *quit*.

4. Implementasi dan Pengujian Sistem

4.1 Pengujian USE (*Usefulness, Satisfaction, & Ease of Use*)

Proses hasil pengujian ini dilakukan dengan masyarakat umum sebagai target pengujinya (*tester*), serta mereka juga diberikan kuesioner untuk kebutuhan data pengujian. Diharapkannya dari hasil pengujian ini dapat memberikan gambaran kepada penulis tentang bagaimana *game* ini dapat dimainkan serta beroperasi dengan baik. Di bawah ini merupakan hasil kuesioner *satisfaction* dari para *tester* dan keterangan lebih detail dari Tabel 1 adalah sebagai berikut:

Tabel 1 Hasil Kuesioner *Satisfaction*

No.	Daftar Pertanyaan	1	2	3
Q1	Apakah Anda menyukai game balapan?	-	38,2%	61,8%
Q2	Apakah game Telyu Racer menarik bagi Anda?	2,9%	23,5%	73,5%
Q3	Apakah dari bermain game Telyu Racer Anda mendapatkan hiburan?	2,9%	17,6%	79,4%
Q4	Apakah Anda kesulitan memainkan game Telyu Racer?	41,2%	47,1%	11,8%
Q5	Apakah Anda seorang <i>gamer</i> ?	25%	55%	20%

Keterangan:

3 = Iya

2 = Mungkin

1 = Tidak

Dari hasil data kuesioner pada Tabel 1 di atas menunjukkan bahwa hampir keseluruhan *player* memberikan respon 3 terhadap *game* Telyu Racer dan pada beberapa *player* ada ditemukannya sedikit kesulitan saat memainkan *game*-nya serta terdapat 20% *player* merupakan seorang *gamer*.

Tabel 2 Hasil Kuesioner *Usefulness & Ease of Use*

No.	Daftar Pertanyaan	1	2	3	4	5
Q6	Bagaimanakah menurut Anda kualitas grafis dari game Telyu Racer?	-	5,9%	23,5%	41,2%	29,4%
Q7	Bagaimana menurut Anda tampilan User Interface pada game Telyu Racer?	-	5,9%	23,5%	41,2%	29,4%
Q8	Apakah fitur-fitur dari game Telyu Racer berfungsi dengan baik? (Contoh: Fitur Kontrol, Paused Menu, dll.)	-	-	5,9%	44,1%	50%

Keterangan:

1 = Sangat Kurang Baik

2 = Kurang Baik

3 = Cukup Baik

4 = Baik

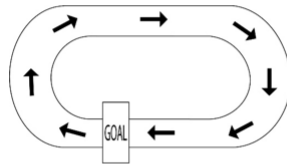
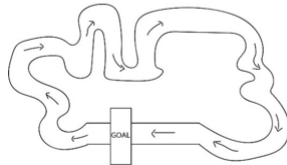
5 = Sangat Baik

Dari hasil data kuesioner pada Tabel 2 di atas menunjukkan bahwa hampir seluruh *player* merasa cukup puas dengan desain tampilan *game* Telyu Racer dan juga fitur-fiturnya dapat dikatakan sudah berfungsi dengan baik.

4.2 Pengujian Performa NPC

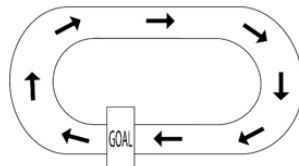
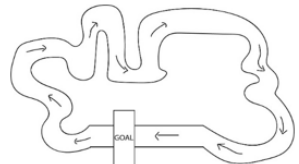
Berikut ini adalah hasil pengujian beserta data persentase dari kemenangan yang dimiliki oleh NPC dapat dilihat pada Tabel 3 dan 4 di bawah ini:

Tabel 3 Hasil Pengujian Performansi Dengan Metode RB

Kelas Uji	Poin Uji	Jumlah Penguji Sebagai Player	Persentase Kemenangan NPC
 <p>Gambar 4.14 Arena 1</p>	Parameter <i>reward</i> sebesar 3.5 dengan 10 kali pertandingan dalam 4 lap	6 Orang	50%
 <p>Gambar 4.15 Arena 2</p>	Parameter <i>reward</i> sebesar 3.5 dengan 5 kali pertandingan dalam 3 lap	6 Orang	23,33333%

Dapat dilihat hasil pengujian terhadap NPC pada Tabel 4.12 dengan penggunaan metode RB bahwa pada percobaan di arena 1 persentase kemenangan pada tiap percobaan menghasilkan persentase kemenangan yang cukup dengan persentase yaitu 50% kemenangan dan pada percobaan di arena 2 persentase kemenangan pada tiap percobaan menghasilkan persentase kemenangan yang cukup rendah dimana mendapatkan hasil dengan persentase kemenangan yaitu 11,66667%, dikarenakan NPC pada arena 2 cukup kesulitan untuk dapat bersaing melawan *player*.

Tabel 4 Hasil Pengujian Performansi NPC Tanpa Metode RB

Kelas Uji	Poin Uji	Jumlah Penguji Sebagai Player	Persentase Kemenangan NPC
 <p>Gambar 4.16 Arena 1</p>	Tanpa parameter dari metode RB dengan 10 kali pertandingan dalam 4 lap	6 Orang	26,666667%
 <p>Gambar 4.17 Arena 2</p>	Tanpa parameter dari metode RB dengan 5 kali pertandingan dalam 3 lap	6 Orang	0%

Pada Tabel 4 di atas dapat dilihat hasil dari pengujian NPC tanpa penggunaan metode RB dimana hasil persentase kemenangannya mendapatkan hasil yang cukup rendah ketimbang saat pengujian menggunakan metode RB, dikarenakan NPC tanpa adanya sistem *reward* membuat NPC selalu melaju dengan kondisi kecepatan tertentu yang di-*random*.

5. Kesimpulan dan Saran

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan pada tugas akhir ini, maka penulis dapat menarik kesimpulan bahwa:

1. Sistem kecerdasan buatan pada *game* yang efektif itu adalah sistem yang dapat mempermudah pengembang dalam melakukan *maintenance* serta kalibrasi terhadap NPC-nya dimana dalam kasus sistem kecerdasan buatan pada NPC Telyu Racer telah berhasil di implementasikan dan dapat dengan mudah di atur parameter dari sistem kecerdasan buaatannya.
2. Berdasarkan hasil pengujian performansi pada penerapan sistem kecerdasan buatan dengan menggunakan metode RB terhadap NPC di *game* Telyu Racer didapatkan hasil persentase kemenangan 50% pada arena 1, sedangkan pada hasil pengujian performansi NPC tanpa adanya penerapan sistem kecerdasan buatan mendapatkan hasil persentase kemenangan yang rendah yaitu 26,666% dan kesulitan untuk dapat bersaing melawan *player*.

5.2 Saran

Berdasarkan hasil penelitian dan pengujian yang telah dilakukan pada tugas akhir ini, maka saran yang dapat diusulkan untuk penelitian lebih lanjut yaitu:

1. Membuat pengaturan serta pilihan konten yang ada di dalam *game* Telyu Racer lebih bervariasi seperti pengaturan *Volume*, *Lap*, *Racer*, *Arena* dan juga sensitifitas input dari *keyboard*.
2. Dapat dibuat versi *mobile* nya atau ditambahkan fitur kontrol tambahan agar dapat menggunakan *game controller*.

REFERENSI

- [1] J. Gao, H.-S. J. Tsao and Y. Wu, *Testing and Quality Assurance for Component-based Software*, Nordwood: ARTECH HOUSE, INC., 2003.
- [2] S. Nelson. United States of America Patent NASA/CR-2003-212806, 2003.
- [3] B. Schwab, *AI Game Engine Programming*, Hingham, Massachusetts: Charles River Media, 2004.
- [4] T. Vaughan, *Multimedia :Maing It Work*. McGraw-hill. 8th Edition., New York, 2011.
- [5] Kenney, "KENNEY," KENNEY, 2021. [Online]. Available: <https://www.kenney.nl/>. [Accessed 2021].
- [6] A. J. J. P. N. Serena H. Chen, "Artificial Intelligence techniques: An introduction to their use for modelling environmental systems," in *Elsevier*, Canberra, 2008.
- [7] N. Kirby, *Introduction to Game AI; Course Technology, a Part of Cengage Learning*, Boston: ELSEVIER, 2009.
- [8] E. Pangkatodi, Liliana and G. Satiabudhi, "Implementasi Rule Base System dan Fuzzy Logic Artificial Intelligence pada Game Kartu Capsa," Surabaya, 2016.
- [9] Sugiyanto, G. Fernando and W.-K. Tai, "A Rule-Based AI Method for an Agent Playing Big Two," *MDPI Computing and Artificial Intelligence*, 2021.
- [10] M. A. Akbar, T. Afirianto, S. W. Sanjaya and R. K. Dewi, "NPC Braking Decision for Unity Racing Game Starter Kit Using Naïve Bayes," *Fountain of Informatics Journal*, vol. 4, no. 2, 2019.
- [11] I. MILLINGTON and J. FUNGE, *ARTIFICIAL INTELLIGENCE FOR GAMES 2nd Edition*, Burlington: Elsevier, 2009.
- [12] J. R. Batmetan, J. Mamonto, H. Legesan and Z. Sagai, "Pengukuran Usability sistem operasi android menggunakan use questionnaire di universitas negeri manado," *Jurnal Pendidikan Teknologi Informasi*, 2018.