

**PERBANDINGAN PENERAPAN METODE FUZZY TSUKAMOTO
DAN METODE FUZZY SUGENO TERHADAP PENENTUAN
DURASI SPAWN NPC MUSUH PADA
GAME TOWER DEFFENSE
ANAK MUSLIM**

SKRIPSI

Oleh :
OKI FIRMANSYAH
NIM. 12650021



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2019**

**PERBANDINGAN PENERAPAN METODE FUZZY TSUKAMOTO DAN
METODE FUZZY SUGENO TERHADAP PENENTUAN
DURASI SPAWN NPC MUSUH PADA
GAME TOWER DEFFENSE
ANAK MUSLIM**

SKRIPSI

Diajukan kepada :
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh :
OKI FIRMANSYAH
NIM 12650021

JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2019

LEMBAR PERSETUJUAN

PERBANDINGAN PENERAPAN METODE FUZZY TSUKAMOTO DAN
METODE FUZZY SUGENO TERHADAP PENENTUAN
DURASI SPAWN NPC MUSUH PADA
GAME TOWER DEFFENSE
ANAK MUSLIM

SKRIPSI

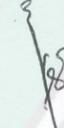
Oleh :
OKI FIRMANSYAH
NIM 12650021

Telah Diperiksa dan Disetujui untuk Diuji
Tanggal, 24 Mei 2019

Pembimbing I


Dr. Suhartono, M. Kom
NIP. 19680519 200312 1 001

Pembimbing II


Roro Inda Melani, MT., MSc
NIP. 19780925 200501 2 008

Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Cahyo Crysian
NIP. 19740424 200901 1 008

LEMBAR PENGESAHAN

**PERBANDINGAN PENERAPAN METODE FUZZY TSUKAMOTO DAN
METODE FUZZY SUGENO TERHADAP PENENTUAN
DURASI SPAWN NPC MUSUH PADA
GAME TOWER DEFFENSE
ANAK MUSLIM**

SKRIPSI

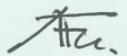
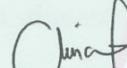
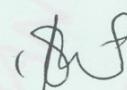
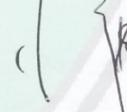
Oleh :
OKI FIRMANSYAH
NIM 12650021

Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)

Tanggal : 24.05.2019

Susunan Dewan Penguji

Tanda Tangan

- | | | |
|-----------------------|-------------------------------------|-------------------------------------------------------------------------------------------|
| 1. Penguji Utama | : <u>Fatchurrohman, M.Kom</u> | () |
| | NIP. 19700731 200501 1 002 | |
| 2. Ketua Penguji | : <u>Ainatul Mardhiyah, M.CS</u> | () |
| | NIDT. 19860330 20160801 2 075 | |
| 3. Sekretaris Penguji | : <u>Dr. Suhartono, M. Kom</u> | () |
| | NIP. 19680519 200312 1 001 | |
| 4. Anggota Penguji | : <u>Roro Inda Melani, MT., MSc</u> | () |
| | NIP. 19780925 200501 2 008 | |

Mengesahkan,

Ketua Jurusan Teknik Informatika

Fakultas Sains dan Teknologi

Universitas Islam Negeri Maulana Malik Ibrahim Malang



PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Oki Firmansyah
NIM : 12650021
Jurusan : Teknik Informatika
Fakultas/ Jurusan : Sains dan Teknologi
Judul Skripsi : Perbandingan Penerapan Metode Fuzzy Tsukamoto Dan Metode Fuzzy Sugeno Terhadap Penentuan Durasi Spawn NPC Musuh Pada Game Tower Defense Anak Muslim

Menyatakan dengan sebenar-benarnya bahwa hasil penelitian saya ini tidak terdapat unsur-unsur penjiplakan karya penelitian atau karya ilmiah yang pernah dilakukan atau dibuat oleh orang lain, kecuali yang secara tertulis dikutip dalam naskah ini dan disebutkan dalam sumber kutipan dan daftar pustaka.

Apabila ternyata hasil penelitian ini terbukti terdapat unsur-unsur penjiplakan, maka saya bersedia untuk mempertanggungjawabkan, serta diproses sesuai peraturan yang berlaku.

Malang, 24 Juni 2019
Yang membuat pernyataan



Oki Firmansyah
NIM 12650021

MOTTO

فَإِنَّ مَعَ الْعُسْرِ يُسْرًا

Sesungguhnya sesudah kesulitan itu ada kemudahan

“No pain, no gain”

Tidak ada pengorbanan, tidak ada hasil

HALAMAN PERSEMBAHAN

Alhamdulillah, segala puji bagi Allah SWT, Rabb semesta alam yang telah memberikan nikmat berlimpah kepada penulis. Shalawat serta salam semoga tetap tercurahkan kepada Baginda Muhammad SAW, yang telah memberi suri tauladan yang luar biasa kepada umatnya.

Pertama, penulis ingin mengucapkan terima kasih yang tak terhingga, kepada orang tua tercinta, Bapak Sulih Hardjoko dan Ibu Eko Purwati. Terima kasih atas dukungan, doa, cinta kasih, dan kesabaran dalam menunggu penulis bisa menyelesaikan segala kewajiban akademis ini.

Kedua, penulis juga berterima kasih kepada dosen-dosen yang telah membimbing penulis menempuh berbagai pelajaran. Terutama kepada Bapak Dr. Suhartono, M. Kom dan Ibu Roro Inda Melani, MT., MSc yang telah memberi banyak masukan untuk terselesaikannya skripsi ini.

Ketiga, terima kasih juga kepada kawan-kawan TI angkatan 2012 yang telah berbagi pengalaman tentang indahnya hidup ini. Semoga skripsi yang telah ada ini bisa bermanfaat bagi orang lain. Amin.

KATA PENGANTAR

Segala puji bagi Allah SWT, Rabb semesta alam. Shalawat dan salam semoga senantiasa ditujukan bagi Rasulullah SAW, keluarga, para sahabat, dan siapa saja yang meneladani mereka dengan baik hingga hari kiamat.

Dalam menyelesaikan skripsi ini, banyak pihak yang telah memberikan bantuan baik secara moril, nasihat, dan semangat maupun materil. Atas segala bantuan yang telah diberikan, penulis ingin menyampaikan doa dan ucapan terimakasih yang sedalam-dalamnya kepada :

1. Bapak Prof. Dr. Abdul Haris, M.Ag selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Ibu Dr. Sri Harini, M.Si selaku dekan Fakultas Sains dan Teknologi Universitas Maulana Malik Ibrahim Malang beserta seluruh staf.
3. Bapak Dr. Cahyo Crysdayan, selaku ketua jurusan teknik informatika yang telah memberikan motivasi untuk terus berjuang.
4. Bapak Dr. Suhartono, M. Kom dan Ibu Roro Inda Melani, MT., MSc selaku dosen pembimbing penulis yang telah meluangkan waktu untuk membimbing, mengarahkan dan memberi masukan kepada penulis dalam pengerjaan skripsi ini hingga akhir.
5. Seluruh Dosen, Laboran, dan Staff Administrasi Jurusan Teknik Informatika UIN Malang, terima kasih atas segala ilmu dan bimbingannya.
6. Segenap dosen teknik informatika yang telah memberikan bimbingan keilmuan kepada penulis selama masa studi.
7. Seluruh rekan-rekan studi yang tidak dapat disebutkan satu persatu, terima kasih atas segala kebaikan yang diberikan kepada penulis.

Penulis menyadari bahwa dalam penyusunan Skripsi ini masih terdapat kekurangan. Oleh karena itu, penulis menerima saran dan kritik yang membangun dari pembaca sekalian. Penulis juga berharap agar Skripsi ini bisa memberikan manfaat kepada para pembaca khususnya, dan bagi penulis secara pribadi. *Amin Ya Rabbal Alamin.*

Malang, 24 Juni 2019

Penulis



DAFTAR ISI

HALAMAN JUDUL.....	i
LEMBAR PERSETUJUAN.....	ii
LEMBAR PENGESAHAN.....	iii
PERNYATAAN KEASLIAN TULISAN.....	iv
MOTTO.....	v
HALAMAN PERSEMBAHAN.....	vi
KATA PENGANTAR.....	vii
DAFTAR ISI.....	ix
DAFTAR GAMBAR.....	xii
DAFTAR TABEL.....	xiv
DAFTAR KODE SUMBER.....	xv
ABSTRAK.....	xvi
ABSTRACT.....	xvii
<u>الملخص.....</u>	xviii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah.....	3
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	4
1.6 Metode Penelitian.....	4
1.7 Sistematika Penulisan Skripsi.....	5
 BAB II TINJAUAN PUSTAKA.....	 7
2.1 Definisi Non Playable Character.....	7
2.2 Genre Game.....	7
2.3 Definisi Tower Defense.....	11
2.4 Dampak Positif dan Negatif Video Game.....	12
2.5 Definisi Animasi.....	15
2.6 Logika Fuzzy.....	15

2.7	Unity 3D.....	18
2.7.1	Antarmuka Unity.....	18
2.7.2	Antarmuka Develop.....	20
2.8	Penelitian Terkait.....	21
 BAB III DESAIN DAN IMPLEMENTASI.....		25
3.1	Desain Sistem.....	25
3.1.1	Keterangan Umum Game.....	25
3.1.2	Storyboard.....	25
3.1.3	Storyline.....	28
3.1.4	Rancangan FSM.....	28
3.1.5	Deskripsi Tower.....	29
3.1.6	Deskripsi Karakter Jahat.....	31
3.2	Algoritma Game.....	34
3.2.1	Skenario Perubahan Perilaku Karakter Jahat atau NPC.....	34
3.3	Desain Fuzzy Karakter Jahat atau NPC.....	35
 BAB IV UJI COBA DAN PEMBAHASAN.....		51
4.1	Game Perspektif Islam.....	51
4.2	Implementasi Antarmuka.....	55
4.2.1	Main Menu.....	55
4.2.2	Game.....	55
4.3	Implementasi Sistem Game.....	56
4.3.1	Pengaturan Lane.....	56
4.3.2	Pengaturan Gold.....	60
4.4	Implementasi Fuzzy Tsukamoto dan Fuzzy Sugeno pada Durasi Spawn NPC.....	61
4.4.1.	Durasi Spawn NPC Tsukamoto.....	67
4.4.2.	Durasi Spawm NPC Sugeno.....	68
4.5	Uji Coba.....	70
4.6	Hasil Pengujian Durasi Spawn Musuh Pada Game Tower Deffense....	70

BAB V KESIMPULAN DAN SARAN.....	77
5.1. Kesimpulan.....	77
5.2. Saran.....	77
DAFTAR PUSTAKA.....	79



DAFTAR GAMBAR

Gambar 2.1 Antarmuka Unity 2D.....	18
Gambar 2.2 Panel Inspector Unity.....	19
Gambar 2.3 Panel Heirarchy Unty.....	19
Gambar 2.4 Panel Project Unity.....	20
Gambar 2.5 Antarmuka Develop Unity.....	20
Gambar 3.1 Storyboard Menu Utama.....	25
Gambar 3.2 Storyboard Menu Map.....	26
Gambar 3.3 Storyboard Map Level 1.....	26
Gambar 3.4 Storyboard Map Level 2.....	27
Gambar 3.5 Storyboard Map Level 3.....	27
Gambar 3.6 Rancangan FSM.....	29
Gambar 3.7 Karakter Rony Dead.....	30
Gambar 3.8 Karakter Rony Idle.....	30
Gambar 3.9 Karakter Rony Reload.....	31
Gambar 3.10 Karakter Rony Shoot.....	31
Gambar 3.11 Karakter Jahat Idle.....	32
Gambar 3.12 Karakter Jahat Attack.....	33
Gambar 3.13 Karakter Jahat Walk.....	33
Gambar 3.14 Karakter Jahat Dead.....	34
Gambar 3.15 Desain Fuzzy.....	35
Gambar 3.16 Derajat Keanggotaam Jumlah Musuh.....	35
Gambar 3.17 Derajat Keanggotaan Jumlah Tower.....	36
Gambar 3.18 Derajat Keanggotaan Level.....	38
Gambar 3.19 Derajat Keanggotaan Durasi Spawn NPC.....	39
Gambar 4.1 Menu Utama.....	55
Gambar 4.2 Tampilan Game Tower Deffense Anak Muslim.....	55
Gambar 4.3 Tampilan Lane.....	56
Gambar 4.4 Tampilan Gold.....	60
Gambar 4.5 Hasil Perubahan Durasi Spawn NPC Tsukamoto 15 detik.....	72
Gambar 4.6 Hasil Perubahan Durasi Spawn NPC Tsukamoto 12 detik.....	72

Gambar 4.7 Hasil Perubahan Durasi Spawn NPC Tsukamoto 9 detik.....	73
Gambar 4.8 Hasil Perubahan Durasi Spawn NPC Tsukamoto 6 detik.....	73
Gambar 4.9 Hasil Perubahan Durasi Spawn NPC Sugeno 15 detik.....	74
Gambar 4.10 Hasil Perubahan Durasi Spawn NPC Sugeno 15 detik.....	74
Gambar 4.11 Hasil Perubahan Durasi Spawn NPC Sugeno 15 detik.....	75
Gambar 4.12 Hasil Perubahan Durasi Spawn NPC Sugeno 13 detik.....	75
Gambar 4.13 Hasil Perubahan Durasi Spawn NPC Sugeno 9 detik.....	76

DAFTAR TABEL

Tabel 3.1 Perubahan Kecepatan Durasi Spawn NPC.....	34
Tabel 3.2 Variabel Linguistik Input Jumlah Musuh.....	36
Tabel 3.3 Variabel Linguistik Input Jumlah Tower.....	37
Tabel 3.4 Variabel Linguistik Input Level.....	38
Tabel 3.5 Variabel Linguistik Output Karakter Jahat.....	39
Tabel 3.6 Aturan Fuzzy Durasi Kemunculan NPC.....	40
Tabel 4.1 Pengujian Fuzzy Tsukamoto dan Fuzzy Sugeno pada Durasi Spawn NPC Musuh.....	71

DAFTAR KODE SUMBER

Kode Sumber 4.1 Pengaturan Lane.....	57
Kode Sumber 4.2 Pengaturan Gold.....	60
Kode Sumber 4.3 Implementasi Fuzzy.....	62
Kode Sumber 4.4 Durasi Spawn NPC Musuh Tsukamoto.....	67
Kode Sumber 4.5 Durasi Spawn NPC Musuh Sugeno.....	69

ABSTRAK

Firmansyah, Oki. 2019. **Perbandingan Penerapan Metode Fuzzy Tsukamoto Dan Metode Fuzzy Sugeno Terhadap Penentuan Durasi Spawn NPC Musuh Pada Game Tower Deffense Anak Muslim.** Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Pembimbing : (I) Dr. Suhartono, M. Kom (II) Roro Inda Melani, MT., MSc

Kata Kunci : Fuzzy Tsukamoto, Fuzzy Sugeno, *Game Tower Defense, Non Playable Character NPC, Android.*

Perkembangan dunia Game dari masa ke masa terus mengalami perkembangan, hingga kini bermunculan beberapa *genre* game dari yang sangat sederhana berbentuk text hingga yang memiliki bentuk animasi 3D. Game Tower defense merupakan subgenre dari game *real-time strategy*, tujuan utama dari game Tower defense adalah untuk melindungi suatu area agar tidak sampai dilewati oleh musuh yang berjalan pada jalur yang sudah ditentukan dengan cara membangun strategi penempatan tower. Konsep game Tower defense saat ini masih sangat jarang yang mengangkat unsur Pendidikan Agama Islam dan menggunakan karakter yang menggambarkan ciri khas seorang Muslim-Muslimah yang didalamnya menyisipkan pengetahuan mengenai keberadaan jin di alam semesta ini berdampingan dengan manusia. Dengan metode Fuzzy Tsukamoto dan Fuzzy Sugeno maka akan didapatkan manakah diantara metode tersebut yang paling efektif dalam memunculkan karakter lawan. Berdasarkan hasil pengujian metode 12 data yang berbeda, durasi yang dihasilkan dari menggunakan Fuzzy Tsukamoto tersebut adalah durasi spawn NPC musuh sedikit 35.00%, durasi spawn NPC musuh sedang 30.00% dan durasi spawn NPC musuh banyak 35.00% dari 12 data yang ada. Sedangkan durasi yang dihasilkan dari menggunakan Fuzzy Sugeno tersebut adalah durasi spawn NPC musuh sedikit 30.00%, durasi spawn NPC musuh sedang 20.00% dan durasi spawn NPC musuh banyak 50.00% dari 12 data yang ada.

ABSTRACT

Firmansyah, Oki. 2019. **Comparison of the Application of Fuzzy Tsukamoto Method and Fuzzy Sugeno Method to Determination NPC Duration of Enemies in Tower Defense Game for Muslim Children.** Thesis. Informatics Engineering Department of the Science and Technology Faculty of the State Islamic University of Maulana Malik Ibrahim Malang.
Mentor: (I) Dr. Suhartono, M. Kom (II) Roro Inda Melani, MT., MSc

Keywords : Fuzzy Tsukamoto, Fuzzy Sugeno, Tower Tower Game, Non Playable Character NPC, Android.

The development of the world The game from time to time continues to experience development, until now there are emerging several game genres from very simple in the form of text to those that have the form of 3D animation. The Tower Defense game is a subgenre of the real-time strategy game, the main goal of the Tower defense game is to protect an area from being overtaken by enemies who are running on a predetermined path by building a tower placement strategy. The Tower defense game concept is still very rare today that elevates the elements of Islamic Education and uses characters that describe the characteristics of a Muslim woman in which inserts knowledge about the existence of jinn in this universe alongside humans. With the Fuzzy Tsukamoto and Fuzzy Sugeno methods, it will be found that among the methods that are the most effective in generating opposing characters. Based on the results of testing 12 different data methods, the duration resulting from using Fuzzy Tsukamoto is the duration of enemy NPC spawn of 35.00%, spawn duration NPC enemies are 30.00% and the duration of enemy NPC spawn is 35.00% of the 12 data available. While the duration generated from using the Fuzzy Sugeno is the duration of enemy NPC spawning of a little 30.00%, the duration of enemy NPC spawn is 20.00% and the duration of enemy NPC spawn is 50.00% of the 12 data available.

الملخص

فرمنس، أوكي. ٢٠١٩. طريقة بين مقارنة تفرخ مدة لتحديد سو غينو فوزي و سوكمنتو فوزي NPC المسلمين للأطفال طاويرديفنس للعبة في للأداء. أطروحة. والتكنولوجيا العلوم بكلية المعلوماتية قسم مالانج إبراهيم مالك بمولانا الإسلامية الدولة بجامعة.

المستشار: سو هرتونو ٢ ميلني راراند ١

الكلمات: سو غينو فوزي و سوكمنتو فوزي, NPC, اندرويت

من للعبة ناشئة أنواع عدة هناك الآن حتى التطوير تجربة لآخر وقت من اللعبة تستمر العالم تطور التي تلك نص شكل في للغاية بسيطة الأبعاد ثلاثة متحركة رسوم شكل لديها. طاويرديفنس لعبة الفعلى الوقت في الإستراتيجية الألعاب من فرعية لعبة هي . حماية هو طاويرديفنس لعبة من والهدف استراتيجيات بناء خلال من ثابتة مسارات على يركضون أعداء قبل من عليها التغلب من العدو من منطقة البرج وضع التربية عنصر اعتماد جداً النادر . متغير لا الحاضر الوقت في طاويرديفنس اللعبة مفهوم واستخدم الإسلامية في الجن وجود حول المعرفة فيها تدرج المسلمة المسلمين تصف شخصيات البشر جانب إلى الكون هذا أي على العثور سيتم عندها سو غينو فوزي و سوكمنتو فوزي الطريقة بالشخصية إظهار في فعالية الأكثر الطرق هذه من فوزي الاستخدام وقت من سوكمنتو، ٣٠٪ متوسط العدو، مختلفة بيانات ١٢ من ٣٥ قليلاً العدو الشعب مجلس تفرخ ٣٥٪ كثير العدو أن ٢٠٪ متوسط العدو، حين في ٣٠٪ قليلاً العدو الشعب مجلس تفرخ سو غينو فوزي استخدام من المتولدة المدة٪ ٥٠٪ . كثير العدو، المتاحة بيانات ١٢ من.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Menurut Fauzadin (2016:19) Perkembangan dunia game dari generasi ke generasi terus mengalami perubahan dan perkembangan, hingga saat ini tercipta beberapa genre game dari yang paling sederhana berbentuk teks sampai yang memiliki bentuk animasi tiga dimensi. Perkembangan game inipun tidak hanya dilakukan pada tampilan atau interfacenya saja melainkan juga pada kemampuan game itu sendiri untuk bisa mempelajari karakter permainan player dengan menerapkan sistem kecerdasan buatan yaitu *Artificial Intelegent (AI)*.

Tower Defense adalah subgenre dari video game Real-Time Strategy (Dugan, 2007), Game ini berfokus pada alokasi resource dan penempatan unit (Tower) (Avery, et al., 2011). Dimana tujuannya adalah untuk menghentikan musuh mencapai titik tertentu di peta dengan membangun berbagai menara yang berbeda yang akan menembak musuh saat mereka lewat. Musuh dan menara biasanya memiliki kemampuan yang bervariasi, biaya dan harga upgrade. Ketika musuh dikalahkan, pemain mendapatkan uang atau poin, yang bisa digunakan untuk membeli atau mengupgrade menara, upgrade jumlah uang atau poin yang diperoleh, atau bahkan meng-upgrade tingkat upgradenya sendiri (Rutkoff, 2007).

Game Tower defense yang ada saat ini kebanyakan masih jarang yang mengangkat unsur Pendidikan Agama Islam dan menggunakan karakter tower

yang menggambarkan ciri khas seorang Muslim-Muslimah yang didalamnya menyisipkan pengetahuan mengenai keberadaan jin di alam semesta ini berdampingan dengan manusia, karena kebanyakan mengangkat tema seputar perang, invasi ke satu daerah. Dengan ini nantinya diharapkan anak-anak Muslim akan lebih bisa termotivasi dan ter dorong untuk belajar tentang yang ghaib dan memperkuat iman mereka kepada Allah SWT. Dimana manusia hidup di dunia ini tidaklah sendiri melainkan berdampingan dengan hal-hal yang ghaib seperti yang tertulis dalam ayat berikut :

وَمَا خَلَقْتُ الْجِنَّ وَالْإِنْسَ إِلَّا لِيَعْبُدُونِ

“Dan aku tidak menciptakan jin dan manusia melainkan supaya mereka mengabdi kepada-Ku.” (QS. Adz-Dzariyat: 56)

Berdasarkan permasalahan yang ada, maka akan dibangun game tower defense dengan judul “Perbandingan Penerapan Metode Fuzzy Tsukamoto Dan Metode Fuzzy Sugeno Terhadap Penentuan Durasi Spawn NPC Pada Game Tower Defense Anak Muslim” dimana pada gameplay-nya menggambarkan beberapa karakter pemain anak-anak muslim yang sedang menjaga area suci masjid dari serangan musuh.

1.2 Rumusan Masalah

Berdasarkan latar belakang masalah yang telah diuraikan di atas, maka yang menjadi pokok permasalahan adalah manakah yang lebih efektif antara

metode Fuzzy Sugeno dan Fuzzy Tsukamoto dalam memunculkan karakter lawan pada Game Tower Defense ?

1.3 Batasan Masalah

Agar penelitian fokus, permasalahan yang ditinjau tidak terlalu luas dan sesuai dengan maksud dan tujuan yang ingin dicapai, maka perlu dibuat batasan masalah. Batasan masalah penelitian ini antara lain :

1. Tools yang digunakan untuk membangun game ini menggunakan Game Engine Unity
2. Game berupa offline single player game.
3. Target penggunanya adalah untuk anak-anak muslim.
4. Karakter dalam game diimplementasikan dengan ciri khas anak muslim.
5. Metode yang akan dibandingkan adalah Logika Fuzzy Sugeno dan Tsukamoto sebagai implementasi dari pergerakan musuh agar serangan yang dibangun lebih optimal dan dinamis

1.4 Tujuan Penelitian

Berdasarkan masalah yang telah dikemukakan, maka maksud dari penelitian ini adalah untuk membandingkan logika fuzzy Tsukamoto dengan fuzzy Sugeno dalam menentukan kemunculan karakter lawan berdasarkan Jumlah Tower, Level dan Jumlah Musuh pada Game Tower Defense.

1.5 Manfaat Penelitian

Hasil dari penelitian ini diharapkan akan memberikan manfaat terhadap pengembangan game di indonesia antara lain :

1. Mengembangkan teknologi virtual untuk simulasi pembelajaran Agama Islam di Indonesia
2. Membuat media pembelajaran pengetahuan Agama Islam berupa game interaktif.

1.6 Metode Penelitian

Berikut langkah-langkah yang digunakan dalam penelitian, terdiri dari :

1. Tahap Pengumpulan Data

Hal yang pertama dilakukan dalam analisis system adalah melakukan pengumpulan data. Teknik pengumpulan data yang digunakan dalam penelitian ini adalah dengan cara mengumpulkan hasil literatur, jurnal, buku-buku atau bacaan-bacaan serta dari internet yang berhubungan dengan pembangunan game tower defense Perbandingan Penerapan Metode Fuzzy Tsukamoto Dan Metode Fuzzy Sugeno Terhadap Penentuan Durasi Spawn NPC Pada Game Tower Defense Anak Muslim berbasis android.

2. Tahap Pengembangan Perangkat Lunak

Teknik analisis data dalam pembuatan perangkat lunak meliputi beberapa proses diantaranya:

- a. Requirements analysis and definition

Tahap ini mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh program

yang akan dibangun. Fase ini harus dikerjakan secara lengkap untuk bisa menghasilkan desain yang lengkap.

b. System and software design

Tahap ini merupakan kegiatan mengumpulkan kebutuhan secara lengkap kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh aplikasi yang akan dibangun. Tahap ini harus dikerjakan secara lengkap untuk bisa menghasilkan desain yang lengkap. Pada tahap ini juga dilakukan analisis algoritma yang akan dipakai pada game Tower defense ini.

c. Implementation and unit testing

Desain program diterjemahkan ke dalam kode-kode dengan menggunakan Game Engine Unity dan merapkan algoritma logika fuzzy tsukamoto sekaligus fuzzy sugeno untuk proses pergerakan pada NPC. Program yang dibangun langsung diuji baik secara unit.

d. Integration and system testing

Penyatuan unit-unit program kemudian diuji secara keseluruhan (system testing).

e. Pembuatan Laporan

Pembuatan Laporan Skripsi sebagai dokumentasi tugas akhir.

1.7 Sistematika Penulisan Skripsi

Sistematika dalam penulisan skripsi ini akan di bagi menjadi beberapa bab sebagai berikut :

BAB I Pendahuluan

Pada bab ini berisi latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, metodologi penelitian, dan sistematika penulisan tugas akhir ini.

BAB II Studi Pustaka

Pada bab ini menjelaskan tentang dasar-dasar teori yang digunakan sebagai penunjang untuk penyusunan tugas akhir ini. Dalam dasar teori yang akan dibahas yaitu dasar teori yang berkaitan dengan permasalahan yang diambil.

BAB III Desain dan Implementasi

Bab ini menguraikan Analisis Sistem, Analisis Masalah, Analisis Game yang akan dibangun, Spesifikasi Kebutuhan Perangkat Lunak, Analisis Kebutuhan Sistem, dan Perancangan Sistem pada Perbandingan Penerapan Metode Fuzzy Tsukamoto Dan Metode Fuzzy Sugeno Terhadap Penentuan Durasi Spawn NPC Pada Game Tower Defense Anak Muslim Berbasis Android.

BAB IV Uji Coba dan Pembahasan

Bab ini membahas tentang implementasi dari algoritma pada NPC musuh terhadap pengembangan game tower defense.

BAB V Kesimpulan dan Saran

Bab ini berisi kesimpulan dari keseluruhan dari laporan tugas akhir dan saran yang diharapkan dapat bermanfaat untuk pengembangan pembuatan game selanjutnya.

BAB II

STUDI PUSTAKA

2.1 Definisi Non Playable Character (NPC)

Autonomous character adalah jenis autonomous agent yang ditujukan untuk penggunaan komputer animasi dan media interaktif seperti games, simulasi dan virtual reality. Agen ini mewakili tokoh dalam cerita atau permainan dan memiliki kemampuan untuk improvisasi tindakan mereka. Ini adalah kebalikan dari seorang tokoh dalam sebuah film animasi yang tindakannya ditulis di muka, sedangkan karakter dalam sebuah permainan atau virtual reality, tindakannya diarahkan secara real time oleh pemain. Dalam permainan, *Autonomous character* biasanya disebut *NPC (Non Playable Character)* (Reynold, 1999:763).

2.2 Genre Game

Jenis game biasa disebut dengan istilah genre game. Selain berarti jenis, genre juga berarti gaya atau format dari sebuah game. Menurut Henry (2010:112) format sebuah game bisa murni sebuah genre atau bisa merupakan campuran (hybrid) dari beberapa genre lain. Game yang penulis buat juga merupakan campuran dari beberapa genre. Jenis-jenis game yang ada menurut Henry (2010:111) adalah sebagai berikut:

1. Maze Game

Jenis game ini biasanya menggunakan maze sebagai setting atau latar game. Jenis game maze ini termasuk jenis game yang paling awal muncul.

Contoh game ini adalah game Pacman dan Digger.

2. Board Game

Game jenis ini sama dengan game board tradisional seperti monopoli. Hanya saja permainan tradisional ini dimainkan melalui komputer.

3. Card Game

Jenis game kartu juga tidak jauh berbeda dari game tradisional aslinya. Namun, tampilannya lebih bervariasi dari versi tradisional. Game ini juga termasuk game yang awal muncul. Contoh game ini adalah Solitaire dan Hearts.

4. Battle Card Game

Contoh game ini yang populer yaitu Battle Card Pokemon. Game ini jarang ditemukan di Indonesia. Film kartun yang bercerita tentang permainan battle card ini pernah ditayangkan di stasiun televisi Indonesia.

5. Quiz Game

Game jenis ini merupakan game dengan bentuk kuis. Contoh Quiz Game yang pernah beredar yaitu game kuis Who Wants to Be Millionaire.

6. Puzzle Game

Jenis game ini memberi tantangan dengan cara menjatuhkan atau melenyapkan sesuatu dari sisi atas ke bawah atau dari kiri ke kanan. Contoh game ini adalah Tetris.

7. Shoot Them Up

Game jenis ini biasanya musuh berbentuk pesawat atau bentuk lain yang datang dari arah kanan, kiri, atau atas yang harus kita tembak sebanyak

dan secepat mungkin. Dulu game ini berbentuk dua dimensi (2D), tetapi sekarang sudah berkembang dan menggunakan efek tiga dimensi (3D).

8. Side Scroller Game

Saat pertama kali muncul game ini berbentuk 2D. Sekarang sudah banyak yang dibuat dengan efek 3D. Pada game jenis ini pemain diharuskan bergerak searah di alur yang disediakan. Dia diharuskan untuk berjalan, meloncat, merunduk serta menghindari rintangan-rintangan. Contoh game ini yang populer yaitu Mario Bros dan Prince of Persia.

9. Fighting Game

Jenis game ini sesuai dengan namanya berisi tentang pertarungan. Contoh game ini yaitu Street Fighter, Samurai Showdown, Virtual Fighter dan Kungfu.

10. Racing Game

Racing game adalah game tentang balapan. Contoh game ini yaitu Need for Speed Underground dan Toca Race Driver.

11. Turn-Based Strategy Game

Pemain dalam game ini melakukan gerakan setelah pemain lain melakukan gerakan jadi saling bergantian. Contoh game yang terkenal adalah Empire dan Civilization.

12. Real-Time Strategy Game

Game ini seperti game Turn-Based Strategy (RTS), namun pada game ini pemain tidak perlu menunggu pemain lain. Pemain tercepatlah yang akan menang. Contoh game ini yaitu Warcraft.

13. SIM

Game genre ini merupakan bentuk permainan simulasi. Di sini pemain membangun sebuah area, kota, negara atau koloni. Contoh game ini yaitu Ship Simulator, Train Simulator, dan Crane Simulator.

14. First Person Shooter

Disebut first person shooter karena pandangan pemain adalah pandangan orang pertama. Banyak baku tembak dan game ini mengutamakan kecepatan gerakan. Contoh game ini yaitu game Counterstrike dan Doom.

15. First Person Shooter 3D Vehicle Based

Game ini sama dengan FPS hanya saja pandangan pemain bukan dari orang pertama, tetapi dari kendaraan atau mesin yang digunakan. Kendaraan itu bisa berupa tank atau kapal.

16. Third person 3D Games

Game ini juga hampir sama dengan FPS hanya sudut pandang pemain merupakan sudut pandang orang ketiga.

17. Role Playing Game

Jenis game ini pemainnya memainkan sebuah tokoh atau karakter. Biasanya ada alur cerita yang harus dijalankan. Contoh game ini adalah Legacy of Kain, Blade of Sword, dan Beyond Divinity.

18. Adventure Game

Adventure game merupakan genre game petualangan. Di sepanjang perjalanan pemain akan menemukan peralatan yang akan disimpan dan berguna sebagai petunjuk perjalanan. Contoh game ini yaitu Sam and Max atau Beyond and Evil.

19. Educational and Edutainment

Game ini lebih mengacu pada isi dan tujuan dari game. Game ini bertujuan memancing minat belajar anak sambil bermain. Contoh game ini adalah game Boby Bola.

20. Sports

Jenis game ini memiliki tema olahraga. Game yang mengetengahkan genre olahraga disebut sport game.

2.3 Definisi Tower Defense

Tower Defense adalah subgenre dari video game Real-Time Strategy (Dugan, 2007), Game ini berfokus pada alokasi resource dan penempatan unit (Tower) (Avery, et al., 2011). Dimana tujuannya adalah untuk menghentikan musuh mencapai titik tertentu di peta dengan membangun berbagai menara yang berbeda yang akan menembak musuh saat mereka lewat. Musuh dan menara biasanya memiliki kemampuan yang bervariasi, biaya dan harga upgrade. Ketika musuh dikalahkan, pemain mendapatkan uang atau poin, yang bisa digunakan untuk membeli atau mengupgrade menara, upgrade jumlah uang atau poin yang diperoleh, atau bahkan mengupgrade tingkat upgradenya sendiri (Rutkoff, 2007). Pengembangan game sejatinya adalah untuk membuat Experience baru bagi para pengguna, bagaimana membuat suatu game menjadi lebih menarik dari sebelumnya, bahkan menurut Lennart Nacke (2014), game yang hebat adalah game yang berfokus pada pengalaman pemain yang hebat pula. Pengembangan game harus dilakukan untuk membuat game yang ada jadi lebih bervariatif dan lebih

menyenangkan dari waktu ke waktu, sehingga pengguna tidak akan pernah bosan memainkan game tersebut. Irshad dan Rohaya Bt Awang Rambli (2014), dalam penelitian mereka menjelaskan, User Experience juga diakui sebagai faktor kualitas yang mempunyai suatu nilai signifikan dari produk dan jasa interaktif. Penerapan desain yang ada bertujuan memastikan secara umum User Experience menyenangkan dan memuaskan. Namun, pendekatan dan metode untuk merancang experience tertentu masih langka. Hasilnya pasar menjadi lebih jenuh, User Experience menjadi faktor kualitas yang kompetitif dominan untuk produk dan layanan interaktif. Game Tower Defense dipilih karena gameplaynya yang bisa diatur tingkat kesulitannya dari mulai Tower defense dengan gameplay yang simple (hanya memasang Tower) hingga yang apling sulit (memasang dan memanajemen resource game).

2.4 Dampak Positif dan Negatif Video Game

(Kompasiana, 2015) Terdapat beberapa dampak positif dari bermain video game, yaitu:

1) Berfikir Kritis

Video game membuat seseorang berfikir lebih cepat dan mampu memahami masalah agar nantinya mampu mencapai tujuan inti game tersebut. Dengan pemahaman berfikir mereka tersebut biasanya akan lebih mudah bagi mereka untuk mempraktekannya di dunia nyata. Mereka akan berfikir lebih kritis dan mampu memahami perkara di sekitarnya.

2) Pengembangan Sosial

Banyak orang berfikir bahwa bermain video game menutup interaksi sosial mereka dengan orang - orang sekitar. Memang hal itu benar, namun juga dalam prakteknya video game itu sendiri memiliki sistem online yang artinya mereka bermain dengan orang dari berbagai tempat di seluruh dunia. Hal ini akan menambah koneksi pertemanan mereka dengan orang lain.

3) Pendidikan

Orang kebanyakan berfikir bahwa video game itu sama sekali tidak mendidik, namun tidak semua game seperti itu. Banyak video game yang mengusung tema praktek pendidikan dalam permainannya. Selain itu banyak pula cerita dalam video game yang menyangkut dengan sejarah dunia, lingkungan politik, dan bahkan ekonomi dunia.

4) Pelepas Stres

Setelah bercaktifitas yang penuh dengan tuntutan fisik dan pikiran, video game dapat dijadikan sebagai pelepas lelah dan stres. Dengan bermain video game, otak manusia dapat melegakan pikirannya sehingga tidak tegang dan mampu menjadi hiburan yang menyenangkan pula bagi otak.

(Kompasiana, 2015) Terdapat dampak negatif dari bermain video game, yaitu:

1) Melakukan Kekerasan

Dengan bermain game yang ada unsur kekerasannya khususnya untuk anak - anak akan mempengaruhi pikiran mereka, dan akan bertindak sesuai apa yang mereka mainkan dalam video game. Tidak jarang terjadi

kekerasan karena faktor video game bahkan dalam contoh ekstrem nya ada pula kasus pembunuhan karena video game.

2) Ketagihan

Seseorang yang bermain video game biasanya lambat laun akan merasa ketagihan. Video game memang memiliki sifat adiktif apabila si pemain belum dapat secara dewasa menahan emosi dan hawa nafsunya untuk bermain video game. Jika sudah ketagihan akan sulit bagi seseorang untuk berhenti bermain. Bahkan tak jarang kita dengar kasus kematian karena bermain video game tanpa henti.

3) Isolasi Sosial

Seseorang yang terus menerus bermain video game, biasanya bersikappa cuek pada dunia sekitarnya lama kelamaan ia justru menjauhi diri dari dunia sekitar dan lebih memilih terus menerus memainkan video game tanpa berkomunikasi lagi karena menganggap diri sendiri berbeda dari orang lain. Biasanya jika sudah begini ia pun mulai melupakan perannya dalam dunia nyata seperti pekerjaan, dan sekolahnya.

4) Membuat Stres

Bagaikan pisau bermata dua video game ini memang terkadang menjadi pelepas stres namun tidak jarang juga terjadi yang sebaliknya. Para pemain video game terkadang dilanda stres karena kegagalan dalam menyelesaikan tingkatan permainan pada video game tersebut. Stres ini semakin bertambah parah jika terjadi kegagalan - kegagalan berikutnya.

2.5 Definisi Animasi

Rona (2013:54) Animasi sebenarnya adalah rangkaian gambar yang disusun berurutan atau dikenal dengan istilah frame. Satu frame terdiri dari satu gambar jika susunan gambar tersebut ditampilkan bergantian dengan waktu tertentu maka akan terlihat bergerak. Satuan yang dipakai adalah frame per second (fps). Misalkan animasi diset 25 fps berarti animasi tersebut terdiri dari 25 gambar dalam satu detik. Semakin besar nilai fps, maka akan dapat terbentuk animasi yang terkesan halus. Animasi berasal dari bahasa latin yaitu “anima” yang berarti jiwa, hidup, semangat. Selain itu kata animasi juga berasal dari kata animation yang berasal dari kata dasar to anime di dalam kamus Indonesia Inggris berarti menghidupkan. Secara umum animasi merupakan suatu kegiatan menghidupkan, menggerakan benda mati. Suatu benda mati diberi dorongan, kekuatan, semangat dan emosi untuk menjadi hidup atau hanya berkesan hidup. Animasi bisa diartikan sebagai gambar yang membuat objek yang seolah – olah hidup, disebabkan oleh kumpulan gambar itu berubah beraturan dan bergantian ditampilkan. Objek dalam gambar bisa berupa tulisan, bentuk benda, warna atau special effect

2.6 Logika Fuzzy

Logika fuzzy merupakan salah satu pembentuk soft computing. Logika fuzzy pertama kali diperkenalkan oleh Prof. Lotfi A. Zadeh pada tahun 1965. Dasar logika fuzzy adalah teori himpunan fuzzy. Pada teori himpunan fuzzy, peranan derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangatlah penting. Nilai keanggotaan atau derajat keanggotaan atau

membership function menjadi ciri utama dari penalaran dengan logika fuzzy tersebut. (Kusumadewi & Purnomo, 2010)

Ada beberapa definisi logika fuzzy, diantaranya :

1. Logika fuzzy adalah logika yang digunakan untuk menjelaskan keambiguan, logika himpunan yang menyelesaikan keambiguan.
2. Logika fuzzy menyediakan suatu cara untuk merubah pernyataan linguistik menjadi suatu numerik.

Logika fuzzy memiliki derajat keanggotaan dalam rentang 0 hingga 1. Berbeda dengan logika digital yang hanya memiliki dua nilai 1 atau 0. Logika fuzzy digunakan untuk menerjemahkan suatu besaran yang diekspresikan menggunakan bahasa (linguistic), misalkan besaran kecepatan laju kendaraan yang diekspresikan dengan pelan, agak cepat, cepat, dan sangat cepat. Dan logika fuzzy menunjukkan sejauh mana suatu nilai itu benar dan sejauh mana suatu nilai itu salah. Logika fuzzy adalah suatu cara yang tepat untuk memetakan suatu ruang input kedalam suatu ruang output. Fuzzy dinyatakan dalam derajat dari suatu keanggotaan dan derajat dari kebenaran. Oleh sebab itu sesuatu dapat dikatakan sebagian benar dan sebagian salah pada waktu yang sama. (Kusumadewi & Purnomo, 2010)

Terdapat banyak model aturan Fuzzy akan tetapi ada dua model aturan yang yang di dalam game tower defense ini gunakan yaitu:

1. Fuzzy Tsukamoto

Saat proses evaluasi aturan dalam mesin inferensi, metode fuzzy Tsukamoto menggunakan fungsi implikasi MIN untuk mendapatkan nilai α -predikat tiap-tiap rule

($\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n$). Masing-masing nilai α -predikat digunakan untuk menghitung hasil inferensi secara tegas (crisp) masingmasing rule ($z_1, z_2, z_3, \dots, z_n$).

Proses defuzzyifikasi pada metode Tsukamoto menggunakan metode rata-rata (Average) dengan rumus berikut:

$$Z = \frac{\sum \alpha_i z_i}{\sum \alpha_i}$$

2. Fuzzy Sugeno

a. Metode Fuzzy Sugeno Orde-Nol

Secara umum bentuk model inferensi fuzzy Sugeno Orde-Nol adalah:

IF(x_1 is A_{1i}) o (x_2 is A_{2i}) o ... o (x_n is A_{ni}) THEN $z = k$

Dengan A_i adalah himpunan fuzzy ke-i sebagai anteseden, dan k adalah suatu konstanta (bersifat crisp) sebagai konsekuensi.

b. Metode Fuzzy Sugeno Orde-Satu

Secara umum bentuk model inferensi fuzzy Sugeno Orde-Satu adalah:

IF (x_1 is A_{1i}) o (x_2 is A_{2i}) o ... o (x_n is A_{ni}) THEN $z = p_1 * x_1 + \dots + p_n * x_n + q$

Dengan A_i adalah himpunan fuzzy ke-i sebagai anteseden, dan p_1 adalah suatu konstanta (tegas) ke-I dan q juga merupakan konstanta dalam konsekuensi. Apabila komposisi aturan menggunakan metode Sugeno, maka defuzzifikasi dilakukan dengan cara mencari nilai ratanya. (Kusumadewi & Purnomo, 2010).

2.7 Unity 3D

Unity3D adalah salah satu game engine cross-platform. Engine ini mendukung tiga bahasa dengan framework Mono open source, C#, JavaScript dan Phyton. Pada Unity3D terdapat fasilitas yang memudahkan untuk pengolongan seperti, asset, animasi, tekstur, suara dan juga memungkinkan untuk mengimport model 3D dari aplikasi modeller lain, seperti Blender, untuk membuat real-time grafis menggunakan mesin rendering buatan sendiri dan juga dikombinasikan dengan nVidia PhysX physics engine.

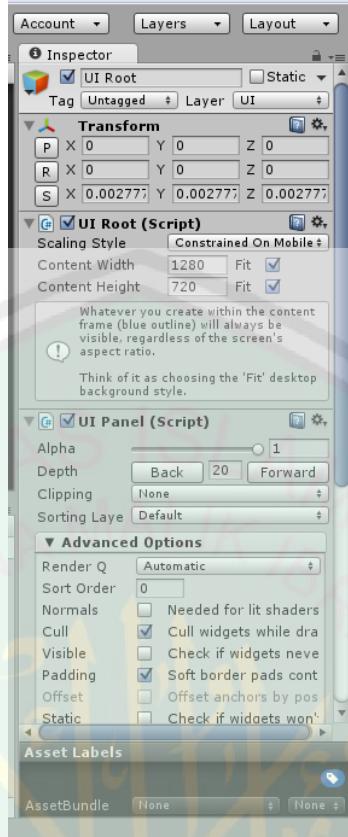
2.7.1 Antarmuka unity



Gambar 2.1 Antarmuka Unity 2D

1. Inspector

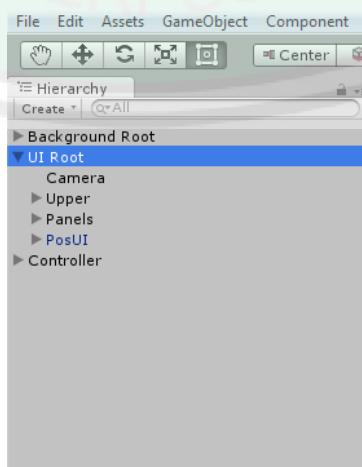
Berisi tentang semua detail objek yang di sorot pada scene, di dalamnya kita bisa mengatur berbagai hal, seperti ukuran objek, letak koordinat objek, model kamera dan lain-lain. Control objek bisa menggunakan C# yang mana dapat mengendalikan dari inputan yang telah di berikan.



Gambar 2.2 Panel Inspector Unity

2. Hierarchy

Hierarchy berisi tentang semua yang ada pada game, dan juga menggabungkan antara objek satu dengan yang lainnya, seperti halnya menggambarkan turunan dari suatu objek tertentu



Gambar 2.3 Panel Hierarchy Unity

3. Project

Project menampilkan semua file yang sudah kita buat pada proyek, pada gambar dapat dilihat pada folder asset terdapat model, texture, scene, script dan lain lain.



Gambar 2.4 Panel Project Unity

4. Scene

Scene menampilkan lembar kerja untuk memanipulasi objek yang ada pada tampilan, meliputi melakukan penggeseran objek, rotasi objek edit ukuran, perspektif objek.

5. Game

Untuk menampilkan objek hasil dari manipulasi scene secara realtime.

2.7.2 Antarmuka Develop

```

Assembly-CSharp-Editor - Editor/LevelEditor.cs - MonoDevelop-Unity
File Edit View Search Project Build Run Version Control Tools Window Help
Debug Unity Editor MonoDevelop-Unity Press 'Control+/' to search
No selected
1 using UnityEngine;
2 using System.Collections;
3 using System.Collections.Generic;
4 using System.IO;
5
6 class LevelEditor : EditorWindow
7 {
8     public static LevelDataController levelDataController;
9     bool isShowWindow;
10    bool[] listLevelSelect = new bool[200];
11    bool[] listNewLevelSelect = new bool[200];
12    bool[] listNewLevelSelected = new bool[200];
13    bool[] listNewLevelSelectedOutline = new bool[200];
14    bool[] listSequencesSelect = new bool[200];
15
16    bool[] listNewSequencesSelect = new bool[200];
17    bool[] listNewSequencesSelectOfSequence = new bool[200];
18    bool[] listNewSequencesSelect = new bool[200];
19
20    Vector2 scrollPos = Vector2.zero;
21
22    LevelDataController.levelDataCollection newLevelDataCollection;
23
24    static bool isShowWindow = false;
25
26    [MenuItem("Game Editor/Level Editor")]
27    public static void ShowWindow()
28    {
29        if (!isShowWindow)
30        {
31            isShowWindow = true;
32        }
33    }
34}

```

Gambar 2.5 Antarmuka Develop Unity

Pada Game engine untuk menggunakan Mono develop sebagai IDE untuk keperluan scripting pada game yang akan dibuat, di dalam game engine tersedia 3 pilihan bahasa pemrograman diantaranya C# Script , JavaScript, dan Boo Script ,Fungsi Monodevelop adalah di gunakan untuk memudahkan kita dalam menulis sebuah program / script yang akan di masukkan pada game, kemudahan tersebut diantaranya adalahmengkoreksi adanya kesalahan penulisan kode, dalam penulisan kode program biasanya tidak luput dalam penulisan huruf atau kekurangan suatu atribut, selain itu digunakan untuk debugging program, sehingga memudahkan proses penanganan error, dan mempermudah proses perbaikannya.

2.8 Penelitian Terkait

Beberapa penelitian yang memiliki hubungan dengan penelitian ini sebagai berikut:

1. Penelitian yang berjudul Rancang Bangun “Chomical”: Game Tower Defense dengan Menerapkan Eksperimen Kimia Menggunakan Kerangka Kerja libGDX, membahas mengenai permainan dengan gameplay tower defence sebagai media pembelajaran sebagai sarana simulasi untuk ekperimen kimia yang akan digunakan sebagai peluru di dalam permainan (Annisa U Istighotsah, Imam Kuswardayan, dan Dwi Sunaryono 2013). Aturan yang terdapat di dalam gameplay permainan ini termasuk unik yaitu pemain harus bereksperimen kimia untuk membuat peluru yang digunakan untuk menyerang musuh dalam game

tower defence. Pengembangan dilakukan dengan kerangka kerja libGDX. Game dikembangkan dengan bahasa pemrograman Java dan implementasi kode dilakukan pada perangkat lunak lingkungan pengembang Eclipse. Metode fungsional yang digunakan untuk uji coba perangkat ini adalah menggunakan metode kotak hitam (black box). Diharapkan dengan pengujian menggunakan metode tersebut, maka didapatkan hasil yang menunjukkan bahwa aplikasi dapat berjalan dengan baik. Game “Chomical” Tower Defence tersebut memiliki kekurangan yaitu efek visual yang kurang sehingga kurang bisa menarik perhatian para pemainnya serta kurangnya jumlah simulasi untuk eksperimen kimia sehingga pemain memiliki pilihan yang lebih bervariasi.

2. Penelitian yang berjudul Rancang Bangun Aplikasi EduGame Selamatkan Hutan Indonesia Berbasis UNITY 3D, membahas tentang permainan edukasi dengan menggunakan gameplay tower defence yang bertujuan untuk pengenalan jenis – jenis pohon yang dilindungi dan memberikan pengetahuan mengenai pentingnya fungsi hutan (Sudarwanto, Ari Budianto & Yoannita Yohannes n.d). Game tersebut menggunakan game Engine Unity 3D dan pembuatan objek pohon yang berbentuk 3D menggunakan aplikasi Autodesk Maya 2013. Metodologi yang digunakan yaitu metodologi prototyping, semua objek pohon berbentuk 3D dibuat terlebih dahulu kemudian membuat rancangan permainan dan mengimport objek 3D kedalam unity 3D. Game ini memiliki musuh yang cerdas dengan menerapkan algoritma fuzzy pada NPC untuk memilih rute teraman yang akan dilalui si musuh tersebut,

sehingga menambah tantangan yang ada di dalam game. Kekurangan dari penelitian ini adalah EduGame Selamatkan Hutan Indonesia masih berjalan pada komputer atau PC sehingga pemain hanya dapat memainkannya pada saat – saat tertentu saja, sedangkan perkembangan sekarang ini banyak aplikasi sudah berjalan pada device mobile agar pemain dapat memainkannya dimana saja dan setiap saat.

3. Penelitian yang berjudul Dynamic Difficulty Adjustment in Tower Defence, membahas tentang sebuah metode agar menciptakan sebuah permainan yang bisa beradaptasi dengan kemampuan para pemain sehingga kesulitan permainan menjadi dinamis (Rhio Sutoyo, Davies Winata, Katherine Oliviani and Dedy Martadinata Supriyadi 2015). Setiap game pasti akan memiliki kesulitan yang berbeda sesuai dengan kemampuan para pemainnya. Tingkat kesulitan yang tinggi akan diterapkan jika pemain menggunakan strategi yang baik dan tingkat kesulitan yang rendah akan ditetapkan jika pemain menggunakan strategi yang buruk. Inti jurnal ini akan membahas tentang penentuan kesulitan berdasarkan life point dari pemain, jumlah musuh yang tersisa, dan keterampilan pasif (skill poin) yang dipilih oleh pemain berdasarkan stage dan juga musuh yang ada. Berdasarkan tiga faktor ini, pemain akan memiliki pengalaman bermain game Tower Defense yang bervariasi karena kombinasi yang berbeda akan memberikan hasil yang berbeda dengan sistem dan kesulitan dari permainan yang berbeda pula untuk setiap gameplaynya. Hasil dari penelitian ini adalah tingkat kesulitan yang dinamis dari game Tower Defense dan penyesuaian kesulitan yang

bersifat dinamis (DDA), dan output gameplay untuk kasus strategi yang baik, cukup, dan buruk.

4. Penelitian yang berjudul Mathematics learning opportunities when playing a Tower Defense Game, membahas tentang kesempatan belajar matematika pada permainan Tower Defense (Aura Hernández-Sabaté, Meritxell Joanpere, Núria Gorgorió and Lluís Albarracín 2015). Penelitian ini dijelaskan bahwa kesempatan belajar anak-anak yang berusia antara 10 sampai 12 tahun dapat dipahami sebagai momen mathematicisable dan melibatkan pembentukan hubungan antara permainan dan pemecahan masalah matematika. Berdasarkan analisis dari mathematicisable moment, dapat disimpulkan bahwa permainan khususnya Tower Defense dapat digunakan untuk proses pemecahan masalah dan memberikan kesempatan belajar yang dapat dikaitkan dengan materi matematika yang berbeda dalam kurikulum, dengan demikian permainan Tower Defense dapat dijadikan elemen yang dapat memfasilitasi proses dalam belajar matematika.

BAB III

DESAIN DAN IMPLEMENTASI

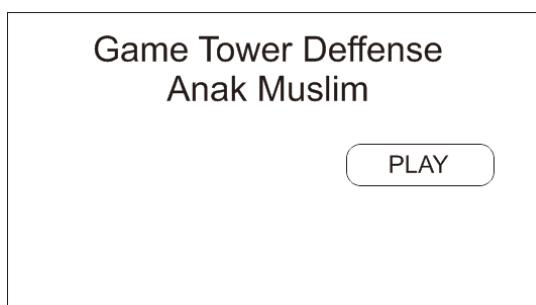
3.1 Desain Sistem

3.1.1 Keterangan Umum Game

Game yang dibangun adalah game single player yang berjenis game tower defense. Dalam permainan ini terdapat beberapa karakter baik yang menjaga area suci berbentuk halaman masjid dimainkan oleh player dan karakter jahat yang bertugas untuk melawan karakter baik dengan cara menyerang mendekati area masjid dimainkan oleh komputer. Objek penelitian dalam permainan game tower defense ini adalah cara kemunculan karakter jahat dalam memasuki area permainan yang dikendalikan oleh komputer.

3.1.2 Storyboard

Alur dari Game Tower Defense Anak Muslim adalah dengan mempertahankan suatu daerah dari serangan musuh. Player akan mengalahkan musuh satu persatu untuk bisa menuju ke tahap level map. Untuk *Game Storyboard* dapat dilihat pada Gambar 3.1 berikut



Gambar 3.1 *Storyboard* Menu Utama

Pada saat membuka game akan ditampilkan judul dan menu untuk memilih memulai game. Ketika memilih untuk memulai game maka akan diarahkan ke menu pemilihan map, bisa dilihat pada gambar 3.2



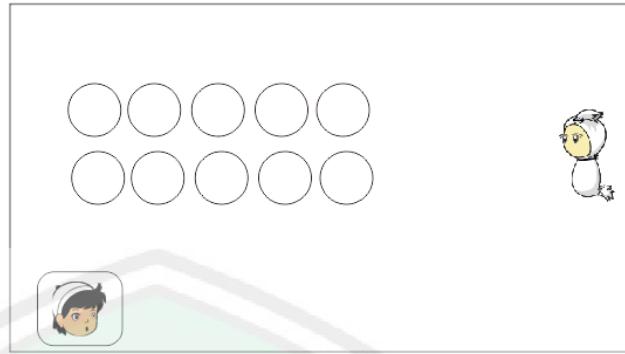
Gambar 3.2 Storyboard Menu Map

Berikutnya akan masuk ke map level 1 yang didalamnya berisi tentang tutorial cara memainkan game dengan jumlah Lane hanya 1 garis, bisa dilihat pada gambar 3.3



Gambar 3.3 Storyboard Map Level 1

Pada map *level 1* player diminta untuk menempatkan tower ke dalam salah satu kolom lane yang tersedia untuk menghadang musuh. Setelah player berhasil membunuh musuh di *level 1* maka akan dilanjutkan ke *level 2*. Untuk storyboard map *level 2* bisa dilihat pada gambar 3.4



Gambar 3.4 Storyboard Map Level 2

Pada map *level 2* ini player dihadapkan pada tingkat kesulitan lebih tinggi dari pada map *level 1*, yaitu jumlah *lane* menjadi dua baris. Player harus bisa mengalahkan semua musuh yang berjalan mendekati markas melalui dua jalur *lane* tersebut. Setelah player berhasil membunuh musuh di *level 2* maka akan dilanjutkan ke *level 3*. Untuk storyboard map *level 3* bisa dilihat pada gambar 3.5



Gambar 3.5 Storyboard Map Level 3

Pada map *level 3* player disediakan jumlah *lane* lebih banyak dari map *level 2* yaitu menjadi sebanyak tiga baris *lane*, dan masing-masing *lane* terdapat lima kolom yang bisa digunakan untuk menempatkan tower. Musuh yang muncul juga menjadi lebih banyak lagi melalui tiga jalur *lane* tersebut.

3.1.3 Storyline

Game tower defense ini merupakan game yang menggambarkan karakter utama yang dikendalikan oleh player untuk menjaga area masjid, yang harus dilakukan player adalah menempatkan posisi para karakter baik ini di dalam petak petak halaman masjid yang nantinya akan dilewati jalur karakter jahat untuk menerobos area masjid. Namun pemain tidak bisa menempatkan karakter baik dengan jumlah yang banyak, terbatas berapa jumlah koin yang dimiliki pemain, setiap karakter baik yang dipanggil akan mengurangi jumlah koin pemain. Dan untuk menambah jumlah koin tersebut, pemain harus melawan karakter jahat yang muncul secara acak ditentukan oleh komputer.

3.1.4 Rancangan FSM

State utama yang tersusun dalam FSM dapat digambarkan sebagai berikut:

- a. *Spawn*

Merupakan *state* posisi awal NPC musuh muncul

- b. Sampai Jangkauan Tembak

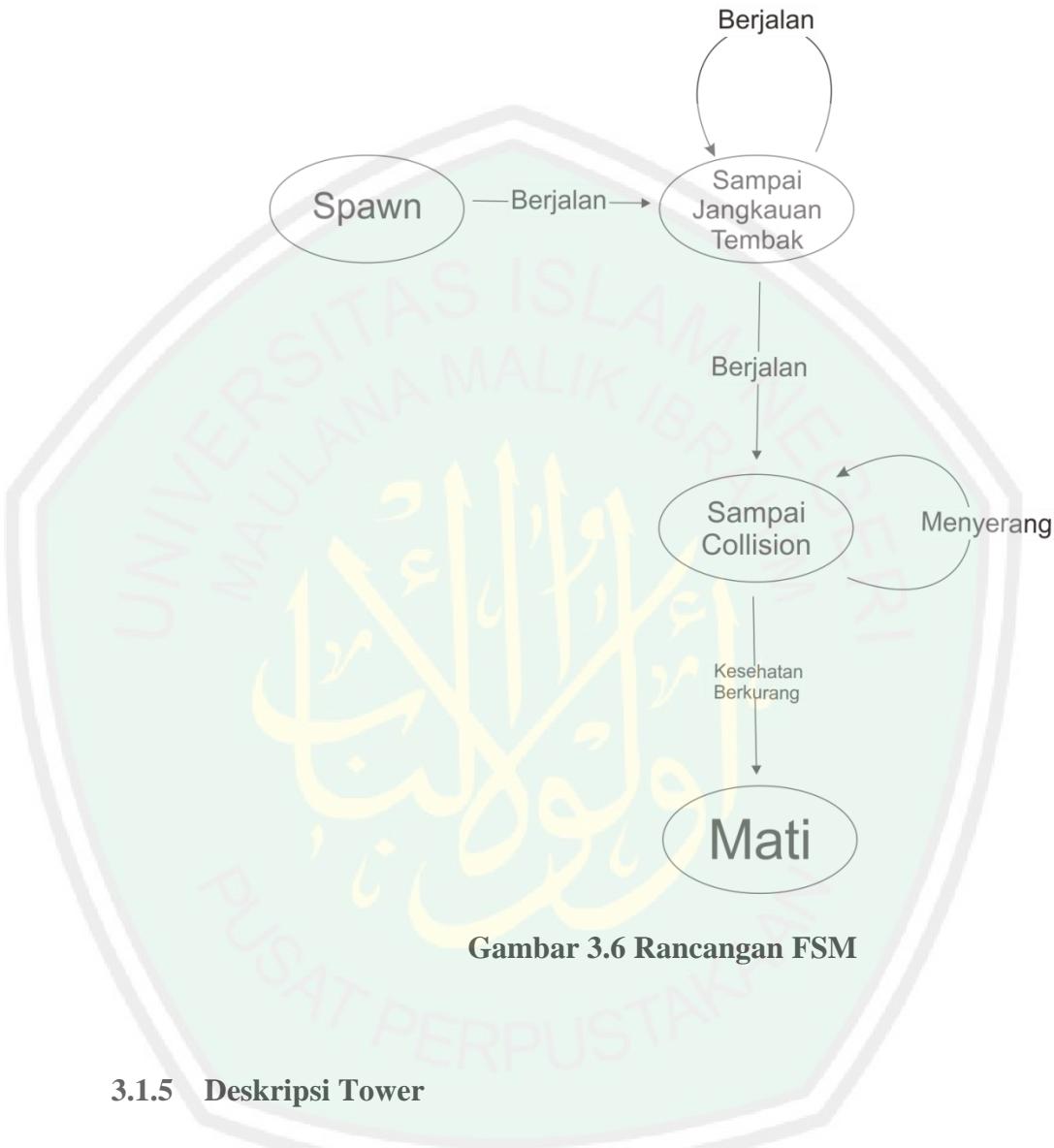
Musuh berjalan kearah markas tower melalui jalur *lane* yang ditentukan dan bertahan dari serangan yang diberikan tower.

- c. Sampai *Collision*

Musuh memberikan serangan saat posisi bertabrakan dengan tower.

d. Mati

Musuh mati ketika nilai kesehatan nol atau habis.



Gambar 3.6 Rancangan FSM

3.1.5 Deskripsi Tower

Dalam Game ini karakter baik sebagai Rony dibuat berupa anak muslim umur 10 tahun. Rony ini hanya bisa di tempatkan pada area *lane* dalam map. Player harus memiliki cukup koin untuk bisa mengambil Rony dan mempertahankan area dari serangan NPC Musuh. Rony memiliki 4 animasi pada saat mempertahankan area yaitu Dead, Idle, Reload, Shoot.

1. Rancangan Rony Dead

Rancangan Rony Dead yang memiliki pergerakan Mati dimodelkan seperti anak muslim yang tergemarkan di tanah karena tidak bisa bertahan dari serangan musuh. Model Rony dead dapat dilihat pada gambar 3.7 berikut :



Gambar 3.7 Karakter Rony Dead

2. Rancangan Rony Idle

Rancangan Rony Idle yang memiliki pergerakan Diam dimodelkan seperti anak muslim yang terdiam siaga menunggu kedatangan musuh. Model Rony Idle dapat dilihat pada gambar 3.8 berikut :



Gambar 3.8 Karakter Rony Idle

3. Rancangan Rony Reload

Rancangan Rony Reload yang memiliki pergerakan yang sama dengan karakter Tower Idle yaitu dimodelkan seperti anak

muslim yang terdiam siaga. Model Rony Reload dapat dilihat pada gambar 3.9 berikut :



Gambar 3.9 Karakter Rony Reload

4. Rancangan Rony Shoot

Rancangan Rony Shoot yang memiliki pergerakan menembakkan amunisi atau Shoot yang dimodelkan seperti anak muslim sedang mengeluarkan tenaga dalam ke arah musuh.

Model Rony Shoot dapat dilihat pada gambar 3.10 berikut :



Gambar 3.10 Karakter Rony Shoot

3.1.6 Deskripsi karakter jahat

Dalam game ini karakter jahat dibuat berupa karakter hantu lokal indonesia. Karakter jahat ini akan muncul secara acak dan

jumlahnya dikendalikan oleh komputer. Setiap permainan dimulai beberapa karakter jahat akan muncul dari arah kanan menuju ke arah kiri mendekati area karakter baik, hingga jalur yang dijaga karakter baik kosong atau karakter jahat telah habis terkena serangan dari karakter baik. Karakter jahat memiliki 4 animasi pada saat menyerang area suci yaitu Idle, Attack, Walk, dan Dead.

1. Rancangan Karakter Jahat Idle

Rancangan Karakter Jahat Idle yang memiliki pergerakan Idle atau diam dimodelkan seperti hantu yang sedang berdiam di tempat pada saat game akan dimulai. Model Karakter Jahat Idle dapat dilihat pada gambar 3.11 berikut :



Gambar 3.11 Karakter Jahat Idle

2. Rancangan Karakter Jahat Attack

Rancangan Karakter Jahat Attcak yang memiliki pergerakan Attack atau menyerang karakter player dimodelkan seperti hantu yang sedang menyerang menggunakan tangannya pada saat posisi terdekat dengan player. Model Karakter Jahat Attcak dapat dilihat pada gambar 3.12 berikut :



Gambar 3.12 Karakter Jahat Attack

3. Rancangan Karakter Jahat Walk

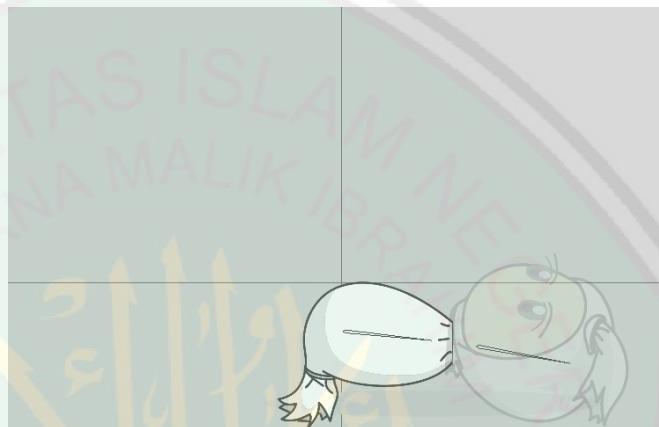
Rancangan Karakter Jahat Walk yang memiliki pergerakan Walk atau berjalan mendekati player dimodelkan seperti hantu yang sedang berjalan dari start awal munculnya sampai mendekati player dan sampai ke tempat tujuan akhir yaitu base player. Model Karakter Jahat Walk dapat dilihat pada gambar 3.13 berikut :



Gambar 3.13 Karakter Jahat Walk

4. Rancangan Karakter Jahat Dead

Rancangan Karakter Jahat Dead yang memiliki pergerakan Dead atau mati dimodelkan seperti hantu yang sedang tertidur atau tumbang ketika score lifenya telah habis. Model Karakter Jahat Walk dapat dilihat pada gambar 3.14 berikut :



Gambar 3.14 Karakter Jahat Dead

3.2 Algoritma Game

3.2.1. Skenario Perubahan Perilaku Karakter Jahat atau NPC

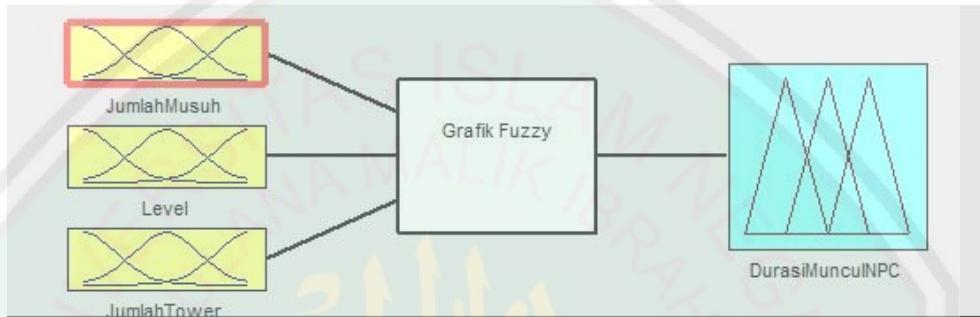
Karakter dalam game ini dibagi menjadi 2 yaitu karakter pemain berbentuk tower (player) dan karakter jahat (NPC) yang menjadi objek penelitian. Karakter jahat ini memiliki variabel yang mempengaruhi perubahan perilaku seperti pada Tabel 3.1 di bawah ini :

Tabel 3.1 Perubahan Kecepatan Durasi Kemunculan NPC

NPC	Variabel Input	Variabel Output
Karakter Jahat	Jumlah Musuh, Jumlah Tower, Level.	Durasi Sedikit, Durasi Sedang, Durasi Banyak.

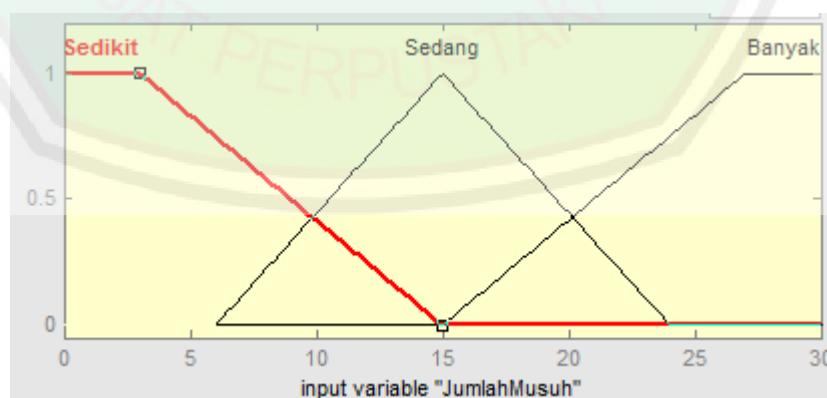
3.3 Desain Fuzzy Karakter Jahat atau NPC

Tiga Variabel digunakan untuk merancang perilaku Karakter Jahat atau NPC yaitu variabel “Jumlah Musuh”, “Jumlah Tower” dan variabel “Level”. Desain fuzzy untuk menghasilkan perilaku karakter jahat atau NPC dapat dilihat pada gambar 3.15 berikut.



Gambar 3.15 Desain Fuzzy

Untuk fungsi keanggotaan “JumlahMusuh”, variabel linguistik mempunyai nilai-nilai linguistik dalam interval numerik yang semantiknya didefinisi oleh fungsi keanggotaannya. Berikut gambar 3.16 yang menunjukkan derajat keanggotaan (membership degree) untuk input “JumlahMusuh” yang mempunyai nilai dalam interval antara 0 sampai 31.



Gambar 3.16 Derajat Keanggotaan Jumlah Musuh

Tabel 3.2 Variabel linguistik input jumlah musuh

Input jumlah musuh		
Variabel	Notasi	Nilai
S	Sedikit	0 - 15
SD	Sedang	6 – 24
B	Banyak	15 - 31

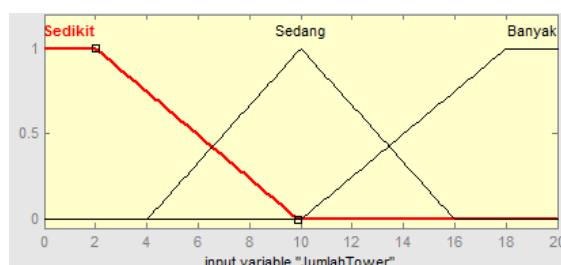
Perhitungan nilai fuzzifikasi variabel jumlah musuh didapatkan dari perhitungan fungsi yang digunakan pada 3 atribut jumlahmusuhsedikit, jumlahmusuhsedang dan jumlahmusuhbanyak secara manual sebagai berikut :

$$\mu_{jumlahmusuhsedikit}[x] = \begin{cases} 1; x \leq 3 \\ 0; x \geq 15 \\ \frac{15-x}{15-3}; 3 < x \leq 15 \end{cases} \dots \dots \dots (1)$$

$$\mu_{jumlahmusuhsedang}[x] = \begin{cases} 1; x \leq 6 \\ 0; x \geq 24 \\ \frac{24-x}{24-6}; 6 < x \leq 24 \end{cases} \dots \dots \dots (2)$$

$$\mu_{jumlahmusuhbanyak}[x] = \begin{cases} 1; x \leq 15 \\ 0; x \geq 27 \\ \frac{27-x}{27-15}; 15 < x \leq 31 \end{cases} \dots \dots \dots (3)$$

Untuk fungsi keanggotaan “Jumlah Tower” mempunyai beberapa variabel linguistik serta notasinya dengan interval nilai beragam, dalam interval 0 sampai 18 yang ditentukan seperti dalam Gambar 3.17 berikut.

**Gambar 3.17 Derajat keanggotaan jumlahtower**

Tabel 3.3 Variabel linguistik input jumlahtower

Input Jumlah Tower		
Variabel	Notasi	Nilai
S	Sedikit	0 - 10
SD	Sedang	4 – 16
B	Banyak	10 - 18

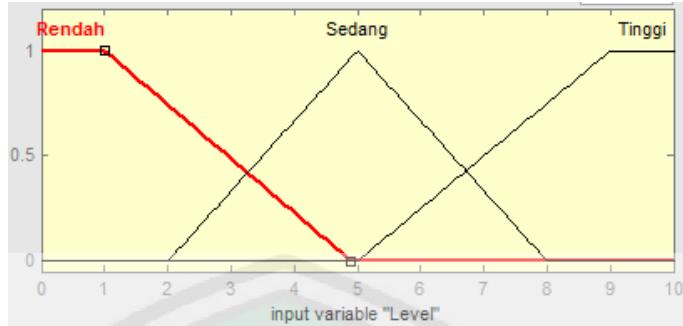
Perhitungan nilai fuzzifikasi variabel Jumlah tower didapatkan dari perhitungan fungsi yang digunakan pada 3 atribut jumlahtowersedikit, jumlahtowersedang dan jumlahtowerbanyak secara manual sebagai berikut :

$$\mu_{\text{jumlahtowersedikit}}[x] = \begin{cases} 1; x \leq 2 \\ 0; x \geq 10 \\ \frac{10-x}{10-2}; 2 < x \leq 10 \end{cases} \dots \dots \dots \quad (4)$$

$$\mu_{\text{jumlahtowersedang}}[x] = \begin{cases} 1; x \leq 4 \\ 0; x \geq 16 \\ \frac{16-x}{16-4}; 4 < x \leq 16 \end{cases} \dots \dots \dots \quad (5)$$

$$\mu_{\text{jumlahtowerbanyak}}[x] = \begin{cases} 1; x \leq 10 \\ 0; x \geq 18 \\ \frac{18-x}{18-10}; 10 < x \leq 16 \end{cases} \dots \dots \dots \quad (6)$$

Untuk fungsi keanggotaan “Level”, variabel linguistik mempunyai nilai-nilai linguistik dalam interval numerik yang semantiknya didefinisi oleh fungsi keanggotanya. Gambar 3.18 berikut menunjukkan derajat keanggotaan (membership degree) untuk input jumlah karakter baik yang mempunyai nilai dalam interval antara 1 sampai 11.



Gambar 3.18 Derajat keanggotaan level

Untuk fungsi keanggotaan “Level” mempunyai beberapa variabel linguistik serta notasinya dengan interval nilai beragam, dalam interval 1 sampai 11 yang ditentukan seperti dalam Tabel 3.4 berikut.

Tabel 3.4 Variabel linguistik input level

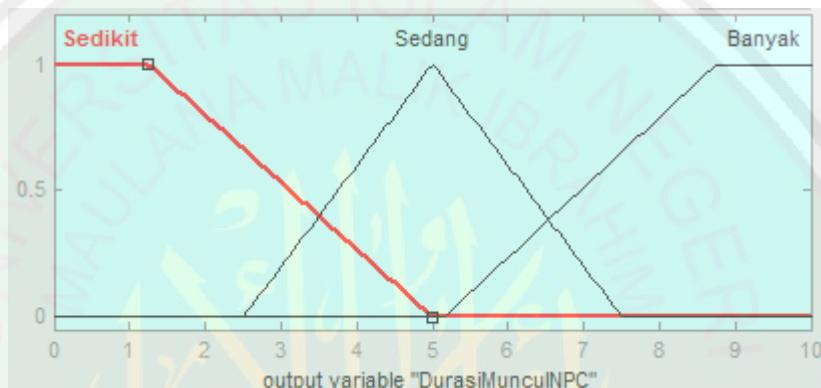
Input Level		
Variabel	Notasi	Nilai
R	Rendah	0 - 5
SD	Sedang	2 – 8
T	Tinggi	5 - 11

Perhitungan nilai fuzzifikasi variabel Jumlah tower didapatkan dari perhitungan fungsi yang digunakan pada 3 atribut levelrendah, levelsedang dan leveltinggi secara manual sebagai berikut :

$$\mu_{\text{levelrendah}}[x] = \begin{cases} 1; & x \leq 1 \\ 0; & x \geq 5 \\ \frac{5-x}{5-1}; & 1 < x \leq 5 \end{cases} \dots \dots \dots \quad (7)$$

$$\mu_{\text{levelsedang}}[x] = \begin{cases} 1; & x \leq 2 \\ 0; & x \geq 8 \\ \frac{8-x}{8-2}; & 2 < x \leq 8 \end{cases} \dots \dots \dots \quad (8)$$

Selanjutnya Gambar 3.19 menunjukkan keanggotaan untuk output pada Karakter Jahat atau NPC. Untuk output durasi kemunculan Karakter jahat atau NPC dijelaskan melalui Tabel 3.5, nilai linguistiknya dibagi menjadi 3, dengan notasi Banyak (B), Sedang (SD) dan Sedikit (S).



Gambar 3.19 Derajat Keanggotaan Durasi Spawn NPC

Tabel 3.5 Varianel linguistik output karakterjahat

Output Karakter Jahat atau NPC		
Variabel	Notasi	Nilai
S	Sedikit	1 - 5
SD	Sedang	3 - 7
B	Banyak	5 - 10

Selanjutnya aturan fuzzy (fuzzy rule) yang disusun dari formulasi 3 input yaitu Jumlah Musuh, Jumlah Tower dan Level untuk menghasilkan perilaku NPC musuh dijelaskan pada Tabel 3.6 berikut.

Tabel 3.6 Aturan Fuzzy Durasi Kemunculan NPC

Input			Output
Jumlah Musuh	Level	Jumlah Tower	Durasi Spawn NPC
Sedikit	Rendah	Sedikit	Sedikit
Sedikit	Rendah	Sedang	Sedikit
Sedikit	Rendah	Banyak	Sedang
Sedikit	Sedang	Sedikit	Sedikit
Sedikit	Sedang	Sedang	Sedang
Sedikit	Sedang	Banyak	Sedang
Sedikit	Tinggi	Sedikit	Sedang
Sedikit	Tinggi	Sedang	Sedang
Sedikit	Tinggi	Banyak	Banyak
Sedang	Rendah	Sedikit	Sedikit
Sedang	Rendah	Sedang	Sedang
Sedang	Rendah	Banyak	Sedikit
Sedang	Sedang	Sedikit	Sedikit
Sedang	Sedang	Sedang	Sedang
Sedang	Sedang	Banyak	Sedang
Banyak	Rendah	Sedikit	Sedang
Banyak	Rendah	Sedang	Sedang
Banyak	Rendah	Banyak	Banyak
Banyak	Sedang	Sedikit	Sedang

Banyak	Sedang	Sedang	Sedang
Banyak	Sedang	Banyak	Banyak
Banyak	Tinggi	Sedikit	Sedang
Banyak	Tinggi	Sedang	Sedang
Banyak	Tinggi	Banyak	Banyak

Dari tabel fuzzy rule , dapat diperoleh fuzzy rule IF/THEN yang menjelaskan hubungan antara input dan output variabel linguistik, yang tersusun sebagai berikut:

1. If (Jumlah Musuh is Sedikit) and (Level is Rendah) and (Jumlah Tower is Sedikit) Then (Jumlah Durasi Spawn NPC Sedikit)
2. If (Jumlah Musuh is Sedikit) and (Level is Rendah) and (Jumlah Tower is Sedang) Then (Jumlah Durasi Spawn NPC Sedikit)
3. If (Jumlah Musuh is Sedikit) and (Level is Rendah) and (Jumlah Tower is Banyak) Then (Jumlah Durasi Spawn NPC Sedang)
4. If (Jumlah Musuh is Sedikit) and (Level is Sedang) and (Jumlah Tower is Sedikit) Then (Jumlah Durasi Spawn NPC Sedikit)
5. If (Jumlah Musuh is Sedikit) and (Level is Sedang) and (Jumlah Tower is Sedang) Then (Jumlah Durasi Spawn NPC Sedang)
6. If (Jumlah Musuh is Sedikit) and (Level is Sedang) and (Jumlah Tower is Banyak) Then (Jumlah Durasi Spawn NPC Sedang)
7. If Jumlah Musuh is Sedikit) and (Level is Tinggi) and (Jumlah Tower is Sedikit) Then (Jumlah Durasi Spawn NPC Sedang)

8. If (Jumlah Musuh is Sedikit) and (Level is Tinggi) and (Jumlah Tower is Sedang) Then (Jumlah Durasi Spawn NPC Sedang)
9. If Jumlah Musuh is Sedikit) and (Level is Tinggi) and (Jumlah Tower is Banyak) Then (Jumlah Durasi Spawn NPC Banyak)
10. If (Jumlah Musuh is Sedang) and (Level is Rendah) and (Jumlah Tower is Sedikit) Then (Jumlah Durasi Spawn NPC Sedikit)
11. If (Jumlah Musuh is Sedang) and (Level is Rendah) and (Jumlah Tower is Sedang) Then (Jumlah Durasi Spawn NPC Sedang)
12. If (Jumlah Musuh is Sedang) and (Level is Rendah) and (Jumlah Tower is Banyak) Then (Jumlah Durasi Spawn NPC Sedang)
13. If (Jumlah Musuh is Sedang) and (Level is Sedang) and (Jumlah Tower is Sedikit) Then (Jumlah Durasi Spawn NPC Sedikit)
14. If (Jumlah Musuh is Sedang) and (Level is Sedang) and (Jumlah Tower is Sedang) Then (Jumlah Durasi Spawn NPC Sedang)
15. If (Jumlah Musuh is Sedang) and (Level is Sedang) and (Jumlah Tower is Banyak) Then (Jumlah Durasi Spawn NPC Sedang)
16. If (Jumlah Musuh is Banyak) and (Level is Rendah) and (Jumlah Tower is Sedikit) Then (Jumlah Durasi Spawn NPC Sedang)
17. If (Jumlah Musuh is Banyak) and (Level is Rendah) and (Jumlah Tower is Sedang) Then (Jumlah Durasi Spawn NPC Sedang)
18. If (Jumlah Musuh is Banyak) and (Level is Rendah) and (Jumlah Tower is Banyak) Then (Jumlah Durasi Spawn NPC Banyak)
19. If (Jumlah Musuh is Banyak) and (Level is Sedang) and (Jumlah Tower is Sedikit) Then (Jumlah Durasi Spawn NPC Sedang)

20. If (Jumlah Musuh is Banyak) and (Level is Sedang) and (Jumlah Tower is Sedang) Then (Jumlah Durasi Spawn NPC Sedang)
21. If (Jumlah Musuh is Banyak) and (Level is Sedang) and (Jumlah Tower is Banyak) Then (Jumlah Durasi Spawn NPC Banyak)
22. If (Jumlah Musuh is Banyak) and (Level is Tinggi) and (Jumlah Tower is Sedikit) Then (Jumlah Durasi Spawn NPC Sedang)
23. If (Jumlah Musuh is Banyak) and (Level is Tinggi) and (Jumlah Tower is Sedang) Then (Jumlah Durasi Spawn NPC Sedang)
24. If (Jumlah Musuh is Banyak) and (Level is Tinggi) and (Jumlah Tower is Banyak) Then (Jumlah Durasi Spawn NPC Banyak)

Sehingga dalam menentukan output atau keputusan dapat dilakukan dengan menggunakan rumus sebagai berikut :

1. Fuzzyifikasi

Menghitung nilai $\mu[x]$ sebagai derajat keanggotaan fuzzy untuk setiap variabel input, menggunakan rumus fungsi keanggotaan 1 – 9.

2. Implikasi

$$\alpha_n = \mu \text{ predikat } n = \min(\mu[x]; \mu[y]) \dots \dots \dots (10)$$

$(\mu[x]; \mu[y])$ = variabel input yang saling berhubungan di dalam rule base.

$$Z_n = Z_{\min} - \alpha_n (Z_{\max} - Z_{\min}) \dots \dots \dots (11)$$

Z_{\min} = nilai terkecil dari masing-masing atribut variabel output Jumlah Durasi Spawn NPC

Zmax = nilai terbesar dari masing-masing atribut variabel output Jumlah Durasi Spawn NPC

3. Defuzzyfikasi

Tsukamoto

Sugeno

Berikut ini adalah studi kasus fuzzy Tsukamoto dan Sugeno untuk Jumlah Durasi Spawn NPC : Contoh kasus, misal pemain pada level 6 jumlah musuh 10 dan telah memasang tower sebanyak 8, maka untuk mengetahui jumlah NPC yang akan muncul adalah sebagai berikut :

Diketahui :

Level = 4

Jumlah Musuh = 12

Jumlah Tower = 7

1. Fuzzyfikasi

Dengan fungsi keanggotaan Jumlah Musuh pada Gambar 3.16 :

$$\mu_{\text{jumlahmusuhsedikit}}[x] = \begin{cases} 1; x \leq 3 \\ 0; x \geq 15 \\ \frac{15-x}{15-3}; 3 < x \leq 15 \end{cases}$$

$$\mu_{\text{jumlahmusuhsedang}}[x] = \begin{cases} 1; x \leq 6 \\ 0; x \geq 24 \\ \frac{24-x}{24-6}; 6 < x \leq 24 \end{cases}$$

$$\mu_{\text{jumlahmusuhbanyak}}[x] = \begin{cases} 1; x \leq 15 \\ 0; x \geq 27 \\ \frac{27-x}{27-15}; 15 < x \leq 27 \end{cases}$$

Nilai Jumlah Musuh 12 termasuk kedalam himpunan fuzzy sedikit dan sedang dengan tingkat keanggotaan sesuai fungsi sebagai berikut:

$$\mu_{\text{jumlahmusuhsedikit}}[12] = \frac{15-12}{15-3} = 0,25 ; 3 < \text{jumlah musuh} \leq 15$$

$$\mu_{\text{jumlahmusuhsedang}}[12] = \frac{24-12}{24-6} = 0,66667 ; 6 < \text{jumlah musuh} \leq 24$$

$$\mu_{\text{jumlahmusuhbanyak}}[12] = 1 ; \text{jumlah musuh} \leq 15$$

Dengan fungsi keanggotaan jumlah tower pada Gambar 3.17:

$$\mu_{\text{jumlahtowersedikit}}[x] = \begin{cases} 1; x \leq 2 \\ 0; x \geq 10 \\ \frac{10-x}{10-2}; 2 < x \leq 10 \end{cases}$$

$$\mu_{\text{jumlahtowersedang}}[x] = \begin{cases} 1; x \leq 4 \\ 0; x \geq 13 \\ \frac{16-x}{16-4}; 4 < x \leq 13 \end{cases}$$

$$\mu_{\text{jumlahtowerbanyak}}[x] = \begin{cases} 1; x \leq 10 \\ 0; x \geq 16 \\ \frac{18-x}{18-10}; 10 < x \leq 16 \end{cases}$$

Nilai Jumlah Tower 7 termasuk kedalam himpunan fuzzy sedikit dan sedang dengan tingkat keanggotaan sesuai fungsi sebagai berikut:

$$\mu_{\text{jumlahtowersedikit}}[7] = \frac{10-7}{10-2} = 0,375 ; 2 < \text{jumlahtower} \leq 10$$

$$\mu_{\text{jumlahtowersedang}}[7] = \frac{13-7}{13-4} = 0,66667 ; 4 < \text{jumlahtower} \leq 13$$

$$\mu_{\text{jumlahtowerbanyak}}[7] = 1 ; \text{jumlahtower} \leq 10$$

Dengan fungsi keanggotaan level pada Gambar 3.18:

$$\mu_{\text{levelrendah}}[x] = \begin{cases} 1; x \leq 1 \\ 0; x \geq 5 \\ \frac{5-x}{5-1}; 1 < x \leq 5 \end{cases}$$

$$\mu_{\text{levelsedang}}[x] = \begin{cases} 1; x \leq 2 \\ 0; x \geq 8 \\ \frac{8-x}{8-2}; 2 < x \leq 8 \end{cases}$$

$$\mu_{\text{leveltinggi}}[x] = \begin{cases} 1; & x \leq 5 \\ 0; & x \geq 11 \\ \frac{9-x}{9-5}; & 5 < x \leq 11 \end{cases}$$

Nilai level 4 termasuk kedalam himpunan fuzzy sedang dan tinggi dengan tingkat keanggotaan sesuai fungsi sebagai berikut:

$$\mu_{\text{levelrendah}}[4] = \frac{5-4}{5-1} = 0,25 ; 1 < \text{level} \leq 5$$

$$\mu_{\text{levelsedang}}[4] = \frac{8-6}{8-2} = 0,6667 ; 2 < \text{level} \leq 8$$

$$\mu_{\text{leveltinggi}}[4] = 1; \text{level} \leq 5$$

2. Implikasi

Dari perhitungan pada fuzzifikasi di atas diperoleh:

$$\mu_{\text{jumlahmusuhsedikit}}[12] = 0,25, \mu_{\text{jumlahmusuhsedikit}}[12] = 0,6667 \text{ dan}$$

$$\mu_{\text{jumlahmusuhsedikit}}[12] = 1 \text{ untuk fungsi jumlahmusuh,}$$

$$\mu_{\text{jumlahtowersedikit}}[7] = 0,375, \mu_{\text{jumlahtowersedang}}[7] = 0,66667 \text{ dan}$$

$$\mu_{\text{jumlahtowerbanyak}}[7] = 1 \text{ untuk fungsi jumlahtower dan}$$

$$\mu_{\text{levelrendah}}[4] = 0,25, \mu_{\text{levelsedang}}[4] = 0,6667 \text{ dan } \mu_{\text{leveltinggi}}[4] =$$

$$1 \text{ untuk fungsi level}$$

Berdasarkan aturan – aturan yang sesuai dengan kondisi tersebut, dengan menggunakan rumus (10) dan (11) maka diperoleh:

1. $\alpha_1 = \min (0,25; 0,25; 0,375) = 0,25 ; z_1=8$
2. $\alpha_2 = \min (0,25; 0,25; 0,666667) = 0,25 ; z_2=4$
3. $\alpha_3 = \min (0,25; 0,25; 1) = 0,25 ; z_3=6$
4. $\alpha_4 = \min (0,25; 0,666667; 0,375) = 0,25 ; z_4=4$
5. $\alpha_5 = \min (0,25; 0,25; 0,666667) = 0,25 ; z_5=6$
6. $\alpha_6 = \min (0,25; 0,25; 1) = 0,25 ; z_6=6$
7. $\alpha_7 = \min (0,25; 1; 0,375) = 0,25 ; z_7=6$
8. $\alpha_8 = \min (0,25; 1; 0,666667) = 0,25 ; z_8=6$
9. $\alpha_9 = \min (0,25; 1; 1) = 0,25 ; z_9=8$
10. $\alpha_{10} = \min (0,666667; 0,25; 0,375) = 0,25 ; z_{10}=4$
11. $\alpha_{11} = \min (0,666667; 0,25; 0,666667) = 0,25 ; z_{11}=6$
12. $\alpha_{12} = \min (0,666667; 0,25; 1) = 0,25 ; z_{12}=6$
13. $\alpha_{13} = \min (0,666667; 0,666667; 0,375) = 0,375; z_{13}=3,5$
14. $\alpha_{14} = \min (0,666667; 0,666667; 0,666667) = 0,666667; z_{14}=4,33333$
15. $\alpha_{15} = \min (0,666667; 0,666667; 1) = 0,666667; z_{15}=4,33333$
16. $\alpha_{16} = \min (1; 0,25; 0,375) = 0,25 ; z_{16}=6$
17. $\alpha_{17} = \min (1; 0,25; 0,666667) = 0,25 ; z_{17}=6$
18. $\alpha_{18} = \min (1; 0,25; 1) = 0,25 ; z_{18}=8$

$$19. \alpha_{19} = \min(1; 0,66667; 0,375) = 0,375; z_{19}=5,5$$

$$20. \alpha_{20} = \min(1; 0,66667; 0,666667) = 0,66667; z_{20}=4,3333$$

$$21. \alpha_{21} = \min(1; 0,66667; 1) = 0,66667; z_{21}=6,3$$

$$22. \alpha_{22} = \min(1; 1; 0,375) = 0,375; z_{22}=5,5$$

$$23. \alpha_{23} = \min(1; 1; 0,666667) = 0,666667; z_{23}=4,33333$$

$$24. \alpha_{24} = \min(1; 1; 1) = 1; z_{24}=5$$

3. Defuzzyifikasi

Dengan menggunakan rumus (12) ditemukan nilai output :

$$Z = \frac{\sum \alpha_i z_i + \dots + \sum \alpha_n z_n}{\sum \alpha_1 + \dots + \sum \alpha_n}$$

$$\begin{aligned} \sum \alpha n \cdot z_n &= (0,25 * 8) + (0,25 * 4) + (0,25 * 6) + (0,25 * 4) + (0,25 \\ &\quad * 6) + (0,25 * 6) + (0,25 * 6) + (0,25 * 6) + (0,25 * 8) \\ &\quad + (0,25 * 4) + (0,25 * 6) + (0,25 * 6) + (0,375 * 3,5) + (\\ &\quad 0,66667 * 4,3333) + (0,6667 * 4,3333) + (0,25 * 6) + (0,25 \\ &\quad * 6) + (0,25 * 8) + (0,375 * 5,5) + (0,66667 * 4,3333) + \\ &\quad (0,66667 * 4,3333) + (0,375 * 5,5) + (0,666667 * 4,3333) + \\ &\quad (1 * 5) \\ &= 48,7152 \end{aligned}$$

$$\begin{aligned} \sum \alpha n &= 0,25 + 0,25 + 0,25 + 0,25 + 0,25 + 0,25 + 0,25 + 0,25 + 0,25 \\ &\quad + 0,25 + 0,25 + 0,25 + 0,375 + 0,6667 + 0,6667 + 0,25 + \\ &\quad 0,25 + 0,25 + 0,375 + 0,6667 + 0,6667 + 0,375 + 0,6667 + 1 \\ &= 9,20833333 \end{aligned}$$

$$Z = \frac{\sum \alpha n \cdot z_n}{\sum \alpha n} = \frac{27,388889}{4,833333} = 5,290346908$$

Diketahui nilai Z adalah 5,290346908. Jadi dengan metode Fuzzy Tsukamoto Jumlah Durasi Spawn NPC akan muncul sebanyak 5,290346908 pada saat Pemain mencapai Level 4, Jumlah Musuh 12 dan Jumlah Tower 7.

Dengan menggunakan rumus (13) ditemukan nilai output :

$$k = \max (\alpha_1, \dots, \alpha_n)$$

$$k = \max (0,25; 0,25; 0,25; 0,25; 0,25; 0,25; 0,25; 0,25; 0,25; 0,25; 0,25; 0,25; 0,375; 0,6667; 0,6667; 0,25; 0,25; 0,25; 0,375; 0,6667; 0,6667; 0,375; 0,6667; 1)$$

$$k = 1$$

Diketahui indek tertinggi adalah 1. Jadi dengan Metode Fuzzy Sugeno Durasi Spawn NPC sebanyak 1 pada saat Pemain mencapai Level 4, Jumlah Musuh 12 dan Jumlah Tower 7

BAB IV

UJI COBA DAN PEMBAHASAN

4.1. Game Perspektif Islam

Dalam Islam Game Tower Defense ini diidentikkan dengan bentuk perlawanan manusia terhadap godaan-godaan setan baik berupa bisikan-bisikan ataupun godaan secara fisik yang bisa merusak Iman manusia. (Ustadz Abu Muslim Al Atsari, 2010) Menurut Imam Ibnu Qayyim, jihad melawan setan, ada dua tingkatan. Pertama, menolak syubhat dan keraguan yang dilemparkan setan. Kedua, menolak syahwat dan kehendak-kehendak rusak yang dilemparkan setan. Jihad yang pertama akan diakhiri dengan keyakinan, sedangkan jihad yang kedua akan diakhiri dengan kesabaran. Allah Subhanahu wa Ta'ala berfirman :

وَجَعْلَنَا مِنْهُمْ أَئِمَّةً يَهُدُونَ بِأَمْرِنَا لَمَّا صَبَرُوا وَكَانُوا بِتَائِيَاتِنَا يُوقْنُونَ

“Dan Kami jadikan di antara mereka itu pemimpin-pemimpin yang memberi petunjuk dengan perintah Kami ketika mereka sabar. Dan adalah mereka meyakini ayat-ayat Kami”.[As Sajdah : 24].

Sesungguhnya seluruh kekuatan, kekuasaan, kesempurnaan hanyalah milik Allah Pencipta alam. Oleh karena itu, seorang hamba yang ditolong dan dilindungi Allah, maka tidak ada seorangpun yang mampu mencelakakannya. Sehingga senjata pertama dan terutama bagi seorang mukmin untuk

menghadapi setan, ialah dengan beriman secara benar kepada Allah, beribadah dengan ikhlas kepadaNya, bertawakkal hanya kepadaNya, dan beramal shalih menurut aturanNya, lewat Sunnah RasulNya.

Allah Subhanahu wa Ta'ala telah memberitakan, setan tidak memiliki kekuasaan terhadap hamba-hambaNya yang beriman dan mentauhidkanNya. Allah berfirman :

إِنَّهُ لَيْسَ لَهُ سُلْطَانٌ عَلَى الَّذِينَ ءَامَنُوا وَعَلَى رَبِّهِمْ يَتَوَكَّلُونَ إِنَّمَا سُلْطَانُهُ عَلَى الَّذِينَ يَتَوَلَّهُ وَالَّذِينَ هُمْ بِهِ مُشْرِكُونَ

“Sesungguhnya setan itu tidak ada kekuasannya atas orang-orang yang beriman dan bertawakkal kepada Rabb-nya. Sesungguhnya kekuasaannya (setan) hanyalah atas orang-orang yang mengambilnya jadi pemimpin dan atas orang-orang yang mempersekutukannya dengan Allah”.[An Nahl : 99, 100].

Ibnul Qayyim menjelaskan, ketika Iblis mengetahui bahwa dia tidak memiliki jalan (untuk menguasai) orang-orang yang ikhlas, maka dia mengecualikan mereka dari sumpahnya yang bersyarat untuk menyesatkan dan membinasakan (manusia). Disebutkan dalam Al Qur'an, Iblis mengatakan.

قَالَ فَبِعِزْرِكَ لَا غُوَيْنَهُمْ أَجْمَعِينَ إِلَّا عِبَادَكَ مِنْهُمْ الْمُخَلَّصِينَ

“Demi kekuasaanMu, aku akan menyesatkan mereka semuanya, kecuali hamba-hambaMu yang mukhlas di antara mereka”.[Shad : 82, 83]

Allah Subhanahu wa Ta'ala berfirman :

إِنَّ عَبْدِي لَيْسَ لَكَ عِنْهُمْ سُلْطَانٌ إِلَّا مَنِ اتَّبَعَكَ مِنَ الْغَاوِيْنَ

“Sesungguhnya hamba-hambaku tidak ada kekuasaan bagimu (Iblis) terhadap mereka, kecuali orang-orang yang mengikutimu, yaitu orang-orang yang sesat” [Al Hijr : 42]

Berpegang Teguh Kepada Al Kitab Dan As Sunnah Sesuai Dengan Pemahaman Salafush Shalih. Ketika Allah menurunkan manusia di muka bumi, sesungguhnya Dia menyertakan petunjuk untuk mereka. Sehingga manusia hidup di dunia ini tidak dibiarkan begitu saja tanpa bimbingan, atau tanpa perintah dan tanpa larangan. Bahkan Allah menurunkan kitab suci dan mengutus para rasul yang membawa peringatan, penjelasan dan bukti-bukti. Barangsiapa berpaling dari peringatan Allah, maka dia akan menjadi mangsa setan dan dijerumuskan ke dalam kecelakaan abadi. Allah Subhanahu wa Ta'ala berfirman :

وَمَنْ يَعْشُ عَنْ ذِكْرِ الرَّحْمَنِ نُقَيْضُنَ لَهُ شَيْطَانًا فَهُوَ لَهُ قَرِيبٌ

“Barangsiapa yang berpaling dari pengajaran (Rabb) Yang Maha Pemurah (Al Qur`an), Kami adakan baginya setan (yang menyesatkan), maka setan itulah yang menjadi teman yang selalu menyertainya” [Az Zukhruf : 36].

Nabi Muhammad Shallallahu ‘alaihi wa sallam telah menjelaskan bahwa jalan Allah, jalan kebenaran, hanya satu. Menyimpang darinya, berarti mengikuti jalan-jalan setan.

عَنْ جَابِرٍ قَالَ كُنَّا جُلُوسًا عِنْدَ النَّبِيِّ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ فَخَطَّ خَطًّا هَكَذَا أَمَامَهُ فَقَالَ هَذَا سَبِيلُ اللَّهِ عَزَّ وَجَلَّ وَخَطَّنِي عَنْ يَمِينِهِ وَخَطَّنِي عَنْ شِمَائِلِهِ قَالَ هَذِهِ سَبِيلُ الشَّيْطَانِ ثُمَّ وَضَعَ يَدَهُ فِي الْخَطِّ الْأَسْوَدِ ثُمَّ تَلَاهُ هَذِهِ الْآيَةُ (وَأَنَّ هَذَا صِرَاطِي مُسْتَقِيمًا فَاتَّبِعُوهُ وَلَا تَنْتَهُوا السُّبُلَ فَتَرَقَ بِكُمْ عَنْ سَبِيلِهِ ذَلِكُمْ وَصَاعِدُوهُ)
 (لَعَلَّكُمْ تَنْتَهُونَ)

“Dari Jabir, dia berkata,”Kami duduk di dekat Nabi Shallallahu ‘alaihi wa sallam, kemudian beliau membuat sebuah garis di depan beliau, lalu beliau mengatakan: ‘Ini jalan Allah Azza wa Jalla. (Kemudian) beliau Shallallahu ‘alaihi wa sallam membuat dua buah garis di kanannya, dan dua garis di kirinya. Beliau bersabda: ‘Ini jalan-jalan setan’. Kemudian beliau meletakkan tangannya pada garis yang hitam (tengah, Pen), lalu membaca ayat ini.” (Dan bahwa (yang kami perintahkan ini adalah jalan-Ku yang lurus, maka ikutilah dia ; dan janganlah kamu mengikuti jalan-jalan kamu mengikuti jalan-jalan (yang lain), karena jalan-jalan itu mencerai-beraikan kamu dari jalan-Nya. Yang demikian itu diperintahkan Allah kepadamu agar kamu bertakwa” [Al An'aam : 153]

Dari kutipan Al-Qur'an dan Hadits di atas, Manusia yang ingin selamat, ia harus berjihad melawan setan dengan bersenjatakan ilmu dan mentazkiyah jiwanya. Pada Game Tower Defense ini menggunakan tower sebagai bentuk cerminan manusia yang seakan-akan sedang mempertahankan area dari serangan setan

4.2. Implementasi Antarmuka

Pada sub bab implementasi antarmuka ini akan dijelaskan komponen-komponen yang ada pada Game Tower Deffense Anak Muslim

4.2.1. Main Menu

Adalah menu utama sebelum melakukan permainan.



Gambar 4.1 Menu Utama

4.2.2. Game

Pada tampilan game ini menampilkan beberapa item diantaranya:



Gambar 4.2 Tampilan Game Tower Deffense Anak Muslim

1. Gold : Menunjukkan berapa banyak perolehan Gold yang didapat selama permainan dijalankan.
2. Pause : Tombol untuk menghentikan permainan
3. Max Death : Menunjukan Jumlah Maksimal kekalahan oleh Karakter baik yang dibangun oleh Player
4. Tower Builder : Tombol untuk membangun Karakter baik yang dapat ditempatkan pada petak-petak map.

4.3. Implementasi Sistem Game

Sistem game yang akan di implementasikan pada game Tower Defense Anak Muslim ini menggunakan bahasa pemrograman C# dan javascript sebagai core dari game, dan menggunakan editor monodevelop. Sedangkan desain visual dan interface menggunakan SpriterPro7.

4.3.1. Pengaturan Lane



Gambar 4.3 Tampilan Lane

Pada bagian ini membahas mengenai untuk mengatur lane yang digunakan dalam game, fungsi lane ini adalah sebagai jalur pertahanan karakter baik atau tower untuk menjaga agar NPC musuh tidak bisa menerobos, tower ini bertahan dari serangan NPC musuh dengan cara diam di tempat menunggu NPC musuh mendekat dan menyerangnya, sekaligus jalur lane ini sebagai lintasan NPC musuh untuk menyerang

tower hingga menerobos ke belakang tower. Berikut script untuk mengatur Lane dalam kelas LaneController.cs:

```

using UnityEngine;
using System.Collections;
using System.Collections.Generic;

public class LaneController : MonoBehaviour
{
    // Use this for initialization
    public GameObject unitsPositionRoot;
    public int maxLane = 3;
    public int maxPositionInLane = 7;
    public Dictionary<int, Transform> positionOfLane = new Dictionary<int, Transform>();

    void Awake()
    {
        if (Master.Lane == null)
        {
            Master.Lane = this;
        }
        // SetLane(maxLane);
    }

    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {

    }

    public void SetUnitAtPosition(GameObject position, GameObject unit)
    {
        listUnitAtPosition[position] = unit;
    }

    public void RemoveUnitAtPosition(GameObject position)
    {
        listUnitAtPosition[position] = null;
    }

    public void ShowUnitPositionsAvailable()
    {
        foreach (GameObject position in listUnitAtPosition.Keys)
        {
            position.SetActive(true);
            if (listUnitAtPosition[position] != null)
            {
                position.SetActive(false);
            }
        }
    }
}

```

```

        }

    public void ChangeColorPosition()
    {
        foreach (GameObject gameObjectAtMouse in Master.
Touch.listGameObjectsAt.mousePosition)
        {
            foreach (GameObject position in listUnitAtPo
sition.Keys)
            {
                GameObject icon = position.transform.Fin
d("Icon").gameObject;

                if (gameObjectAtMouse == position)
                {
                    icon.GetComponent<SpriteRenderer>().
color = new Color(0, 1, 1);
                }
                else
                {
                    icon.GetComponent<SpriteRenderer>().
color = new Color(1, 1, 1);
                }
            }
        }
    }

    public void HideUnitPosition()
    {
        foreach (GameObject position in listUnitAtPositi
on.Keys)
        {
            position.SetActive(false);
        }
    }

    public void SetCharacterAtLane(GameObject character,
int lane)
{
    listLanes[lane].Add(character);
}

public List<GameObject> GetCharactersInLane(int lane
)
{
    return listLanes[lane];
}

public List<GameObject> GetCharactersInLaneByTag(int
lane, string tag)
{
    List<GameObject> characters = new List<GameObjec
t>();

    foreach (GameObject character in listLanes[lane]
)
    {
        if (character != null)
    }
}

```

```

        {
            if (character.tag == tag)
            {
                characters.Add(character);
            }
        }
        return characters;
    }

    public void RemoveCharacterAtLane(int lane, GameObject obj)
    {
        listLanes[lane].Remove(obj);
    }

    public bool IsExistCharacterByTagInLane(int lane, string tag)
    {
        foreach (GameObject character in listLanes[lane])
        {
            if (character != null)
            {
                if (character.tag == tag)
                {
                    return true;
                }
            }
        }
        return false;
    }

    public bool IsExistCharacterByTagInAllLane(string tag)
    {
        for (int i = 1; i <= Master.Level.currentLevelData.NumberOfLanes; i++)
        {
            foreach (GameObject character in listLanes[i])
            {
                if (character != null)
                {
                    if (character.tag == tag)
                    {
                        return true;
                    }
                }
            }
        }
        return false;
    }
}

```

Kode Sumber 4.1 Pengaturan Lane

4.3.2. Pengaturan Gold



Gambar 4.4 Tampilan Gold

Pada bagian ini membahas mengenai untuk mengatur Gold yang diperoleh oleh pemain selama permainan berlangsung. Setiap kali NPC musuh yang tumbang akan menjatuhkan Coin yang bernilai 10 Gold, dan Gold ini nantinya dapat digunakan untuk membangun Tower lebih banyak lagi. Dan setiap kali player membangun Tower maka akan mengurangi nilai Gold sebanyak 40 Point. Berikut script untuk mengatur Gold dalam kelas GoldDrop.cs:

```
using UnityEngine;
using System.Collections;
using DG.Tweening;
public class GoldDrop : MonoBehaviour
{
    // Use this for initialization
    private Rigidbody2D rgBody2d;
    private float firstY;
    public int goldDiambil;

    void Start()
    {
        gameObject.layer = LayerMask.NameToLayer("UI");
        gameObject.transform.parent = Master.UIGameplay.uiRoot.transform;
        rgBody2d = GetComponent<Rigidbody2D>();
        firstY = transform.position.y;
        SetMove();
    }

    // Update is called once per frame
    void Update()
    {
        if (transform.position.y < firstY)
        {
            Destroy(rgBody2d);
        }
    }

    void SetMove()
```

```

    {
        int[] randomX = new int[] { -20, -18, -16, 16, 18, 20 };
        rgBody2d.AddForce(new Vector3(randomX[Random.Range(0, randomX.Length)], Random.Range(60, 70)));
    }

    public void OnTouchIn()
    {

        Master.Audio.PlaySound("snd_getGold");
        transform.DOMove(Master.UIGameplay.totalGoldLabel.transform.position, 0.7f).OnComplete(() =>
        {
            Master.Gameplay.gold += StatsController.GoldPerCoin;
            Destroy(gameObject);
        });
    }
}

```

Kode Sumber 4.2 Pengaturan Gold

4.4.Implementasi Fuzzy Tsukamoto dan Fuzzy Sugeno Pada Durasi Spawn

NPC

Proses Implementasi adalah proses pembangunan komponen-komponen pokok suatu sistem yang didasarkan pada desain dan rancangan yang telah dibuat sebelumnya. Implementasi perancangan Artificial Intelligence pada penelitian ini diterapkan pada pengaturan Durasi Spawn NPC Musuh dengan menggunakan metode Algoritma Fuzzy Tsukamoto dan dibandingkan dengan Fuzzy Sugeno.

Pada bagian ini membahas mengenai implementasi Fuzzy Tsukamoto maupun Fuzzy Sugeno untuk menentukan Durasi Spawn NPC Musuh sehingga dihasilkan output Spawn NPC Musuh yang lebih dinamis. Algoritma Fuzzy bekerja pada saat game baru dimulai dengan inputan awal jumlah musuh = 0, jumlah tower = 0, dan level = 0, sehingga Durasi Spawn dari masing-masing NPC musuh akan bervariasi tergantung kondisi 3 inputan jumlah musuh, jumlah tower dan level. Berikut ini adalah script untuk

implementasi Fuzzy pada Durasi Spawn NPC Musuh dalam kelas LevelController.cs:

```

using UnityEngine;
using System.Collections;

public class LevelController : MonoBehaviour
{
    public int level = 0;
    public int counter = 0;
    public int jumlahMonster = 0;
    [HideInInspector]
    public bool isLastWave;
    [HideInInspector]
    public bool isLastSequence;
    bool isCheckingEnemyExistToStartNewWave;
    void Start()
    {
        if (Master.Level == null)
        {
            Master.Level = this;
        }
        SetLevelInfo();
    }

    // Update is called once per frame
    void Update()
    {

    }

    void OnDestroy()
    {
        StopAllCoroutines();
        CancelInvoke();
    }

    public void SetLevelInfo()
    {
        int currentLevel = 3;
        currentLevelData = Master.LevelData.GetLevelDataByIndex(Master.LevelData.currentLevel);
        totalWaves = 100;
        totalSequences = 100;
        //get total sequence

        totalSequenceIndex = 0;
        Master.Lane.SetLane(currentLevelData.NumberOfLanes, currentLevelData.NumberOfPositionsCanBuildUnitInLane);
        Master.Gameplay.gold = currentLevelData.InitialGold;
    }

    public void StartInitEnemny()
    {
        InitEnemy(0, 0);
    }
}

```

```

private void InitEnemy(int waveIndex, int sequenceIndexOfWave)
{
    //Fuzzy
    UnitSelect us = new UnitSelect();
    float inferensiJMS; float inferensiJMSD;
    float inferensiJMB; float inferensiJTS;
    float inferensiJTSD; float inferensiJTB;
    float inferensiLVR; float inferensiLVSD;
    float inferensiLVT;
    //inferensi Jumlah Musuh Sedikit
    if (jumlahMonster <= 3) {
        inferensiJMS = 1;
    } else if (jumlahMonster >= 15) {
        inferensiJMS = 0;
    } else {
        inferensiJMS = (15 - jumlahMonster) / (15 - 3);
    }
    //inferensi Jumlah Musuh Sedang
    if (jumlahMonster <= 6) {
        inferensiJMSD = 1;
    } else if (jumlahMonster >= 24) {
        inferensiJMSD = 0;
    } else {
        inferensiJMSD = (24 - jumlahMonster) / (24 - 6);
    }
    //inferensi Jumlah Musuh Banyak
    if (jumlahMonster <= 15) {
        inferensiJMB = 1;
    } else if (jumlahMonster >= 27) {
        inferensiJMB = 0;
    } else {
        inferensiJMB = (27 - jumlahMonster) / (27 - 15);
    }

    //inferensi Level Rendah
    if( level <= 1){
        inferensiLVR = 1 ;
    } else if ( level >= 5){
        inferensiLVR = 0 ;
    }else{
        inferensiLVR = (5 - level) / (5 - 1);
    }
    //inferensi Level Sedang
    if( level <= 2){
        inferensiLVSD = 1 ;
    } else if ( level >= 8){
        inferensiLVSD = 0 ;
    }else{
        inferensiLVSD = (8 - level) / (8 - 2);
    }
    //inferensi Level Tinggi
    if( level <= 5){
        inferensiLVT = 1 ;
    } else if ( level >= 9){
        inferensiLVT = 0 ;
    }else{
        inferensiLVT = (9 - level) / (9 - 5);
    }
}

```

```

    }

    //inferensi Jumlah Tower Sedikit
    if( us.jumlahTower <= 2){
        inferensiJTS = 1 ;
    } else if ( us.jumlahTower >= 10){
        inferensiJTS = 0 ;
    }else{
        inferensiJTS = (10 - level) / (10 - 2);
    }
    //inferensi Jumlah Tower Sedang
    if( us.jumlahTower <= 4){
        inferensiJTSD = 1 ;
    } else if ( us.jumlahTower >= 16){
        inferensiJTSD = 0 ;
    }else{
        inferensiJTSD = (16 - level) / (16 - 4);
    }
    //inferensi Jumlah Tower Banyak
    if( us.jumlahTower <= 10){
        inferensiJTB = 1 ;
    } else if ( us.jumlahTower >= 18){
        inferensiJTB = 0 ;
    }else{
        inferensiJTB = (18 - level)/(18-10);
    }

    // alpha predikat tiap rule
    float a1; float a2; float a3; float a4; float a5;
    float a6; float a7; float a8; float a9; float a10;
    float a11; float a12; float a13; float a14; float a15;
    float a16; float a17; float a18; float a19; float a20;
    float a21; float a22; float a23; float a24;

    a1 = Mathf.Min(inferensiJMS,inferensiLVR,inferensiJTS);
;
    a2 = Mathf.Min(inferensiJMS,inferensiLVR,inferensiJTSD);
);
    a3 = Mathf.Min(inferensiJMS, inferensiLVR, inferensiJT
B);
    a4 = Mathf.Min(inferensiJMS,inferensiLVSD,inferensiJTS
);
    a5 = Mathf.Min(inferensiJMS, inferensiLVR,inferensiJTS
D);
    a6 = Mathf.Min(inferensiJMS, inferensiLVR, inferensiJT
B);
    a7 = Mathf.Min(inferensiJMS, inferensiLVT, inferensiJT
S);
    a8 = Mathf.Min(inferensiJMS, inferensiLVT,inferensiJTS
D);
    a9 = Mathf.Min(inferensiJMS, inferensiLVT, inferensiJT
B);
    a10 = Mathf.Min(inferensiJMSD,inferensiLVR,inferensiJT
S);
    a11= Mathf.Min(inferensiJMSD,inferensiLVR,inferensiJTS
D);
    a12 = Mathf.Min(inferensiJMSD,inferensiLVR,inferensiJT
B);
    a13= Mathf.Min(inferensiJMSD,inferensiLVSD,inferensiJT
S);

```

```

        a14=Mathf.Min(inferensiJMSD,inferensiLVSD,inferensiJTS
D);
        a15= Mathf.Min(inferensiJMSD,inferensiLVSD,inferensiJT
B);
        a16= Mathf.Min(inferensiJMB, inferensiLVR, inferensiJT
S);
        a17= Mathf.Min(inferensiJMB,inferensiLVR, inferensiJTS
D);
        a18 = Mathf.Min(inferensiJMB,inferensiLVR, inferensiJT
B);
        a19 = Mathf.Min(inferensiJMB,inferensiLVSD,inferensiJT
S);
        a20= Mathf.Min(inferensiJMB,inferensiLVSD,inferensiJTS
D);
        a21= Mathf.Min(inferensiJMB,inferensiLVSD, inferensiJT
B);
        a22 = Mathf.Min(inferensiJMB,inferensiLVT, inferensiJT
S);
        a23 = Mathf.Min(inferensiJMB,inferensiLVT,inferensiJTS
D);
        a24 = Mathf.Min(inferensiJMB,inferensiLVT, inferensiJT
B);
        // zn dari tiap rule
float z1; float z2; float z3; float z4; float z5;
float z6; float z7; float z8; float z9; float z10;
float z11; float z12; float z13; float z14; float z15;
float z16; float z17; float z18; float z19; float z20;
float z21; float z22; float z23; float z24;

z1 = 9 - (a1 * (9 - 5)); z2 = 5 - (a2 * (5 - 1));
z3 = 7 - (a3 * (7 - 3)); z4 = 5 - (a4 * (5 - 1));
z5 = 7 - (a5 * (7 - 3)); z6 = 7 - (a6 * (7 - 3));
z7 = 7 - (a7 * (7 - 3)); z8 = 7 - (a8 * (7 - 3));
z9 = 9 - (a9 * (9 - 5)); z10 = 5 - (a10 * (5 - 1));
z11 = 7 - (a11 * (7 - 3)); z12 = 7 - (a12 * (7 - 3));
z13 = 5 - (a13 * (5 - 1)); z14 = 7 - (a14 * (7 - 3));
z15 = 7 - (a15 * (7 - 3)); z16 = 7 - (a16 * (7 - 3));
z17 = 7 - (a17 * (7 - 3)); z18 = 9 - (a18 * (9 - 5));
z19 = 7 - (a19 * (7 - 3)); z20 = 7 - (a20 * (7 - 3));
z21 = 9 - (a21 * (9 - 5)); z22 = 7 - (a22 * (7 - 3));
z23 = 7 - (a23 * (7 - 3)); z24 = 9 - (a24 * (9 - 5));

//defuzzyifikasi tsukamoto
float z;
z = ((z1*a1)+(z2*a2)+(z3*a3)+(z4*a4)+(z5*a5)+(z6*a6)+(
z7*a7)+(z8*a8)+(z9*a9)+(z10*a10)+(z11*a11)+(z12*a12)+(z13*a13) +
(z14*a14)+(z15*a15)+(z16*a16)+(z17*a17)+(z18*a18)+(z19*a19)+(
z20*a20)+(z21*a21)+(z22*a22)+(z23*a23)+(z24*a24)) / (a1+a2+a3+a4
+a5+a6+a7+a8+a9+a10+a11+a12+a13+a14+a15+a16+a17+a18+a19+a20+a2
1+a22+a23+a24);
//defuzzyifikasi sugeno
double k;
k = Mathf.Max(a1, a2, a3, a4, a5, a6, a7, a8, a9, a10,
a11, a12, a13, a14, a15, a16, a17, a18, a19, a20, a21, a22, a
23, a24);
Debug.Log("Ini Z = "+z);
Master.WaitAndDo(20, () =>
{
    counter++;
}

```

```

        int asd = Random.Range(1, 4);
        GameObject pf_enemy = Master.GetEnemyPrefabByID("0
1");
        GameObject go_enemy = NGUITools.AddChild(Master.Ga
meplay.enemiesRoot, pf_enemy);
        go_enemy.transform.localPosition = new Vector3(Mas
ter.Gameplay.outOfScreenPos[2].localPosition.x, Master.Lane.po
sitionOfLane[asd].localPosition.y - 30, 0);
        go_enemy.GetComponent<EnemyController>().status.Cu
rrentLane = asd;
        Master.Lane.SetCharacterAtLane(go_enemy, asd);
        if(counter == 5){
            level += 1;
            Debug.Log("Level = "+level);
            counter = 0;
        }
        sequenceIndexOfWave++;
        if (sequenceIndexOfWave >= 10)
        {
            if (!isCheckingEnemyExistToStartNewWave)
            {
                CheckEnemyExistAndStartNewWave(waveIndex,
sequenceIndexOfWave);
            }
        }
        else
        {
            InitEnemy(waveIndex, sequenceIndexOfWave);
            jumlahMonster += 1;
            Debug.Log("Jumlah Monster Cinta = "+jumlah
Monster);
        }
    }, this);
}
void CheckEnemyExistAndStartNewWave(int waveIndex, int seq
uenceIndex)
{
    isCheckingEnemyExistToStartNewWave = true;
    Master.WaitAndDo(0.5f, () =>
    {
        bool isExistEnemy = Master.Lane.isExistCharacterBy
TagInAllLane("Enemy");
        if (!isExistEnemy)
        {
            sequenceIndex = 0;
            waveIndex++;
            InitEnemy(waveIndex, sequenceIndex);
        }
        else
        {
            CheckEnemyExistAndStartNewWave(waveIndex, sequ
enceIndex);
        }
    }, this);
}
}

```

Kode Sumber 4.3 Implementasi Fuzzy

4.4.1. Durasi Spawn NPC Tsukamoto

Pada bagian ini membahas mengenai untuk pengaturan Durasi Spawn NPC Musuh dengan metode Fuzzy Tsukamoto yaitu Durasi Banyak, Durasi Sedang dan Durasi Sedikit. Jeda Spawn NPC Musuh akan semakin lambat atau menjadi lebih lama dikarenakan durasi setelah kemuncul NPC musuh yang pertama dengan kemunculan NPC Musuh berikutnya adalah banyak dan sebaliknya, Jeda Spawn NPC Musuh akan semakin cepat atau menjadi lebih padat dikarenakan durasi setelah kemuncul NPC musuh yang pertama dengan kemunculan NPC Musuh berikutnya adalah sedikit. Berikut ini adalah script untuk Durasi Spawn NPC Musuh Tsukamoto dalam kelas LevelController.cs:

```

int waktuSpawntsukamoto;
    //lovemoto
    if (z < 4) {
        waktuSpawntsukamoto = 20;
    } else if (z < 5) {
        waktuSpawntsukamoto = 17;
    } else if(z < 6){
        waktuSpawntsukamoto = 14;
    }else if ( z < 7 ){
        waktuSpawntsukamoto = 11;
    }else if ( z < 8 ){
        waktuSpawntsukamoto = 8;
    }else if ( z < 9 ){
        waktuSpawntsukamoto = 5;
    }
Master.WaitAndDo(waktuSpawntsukamoto, () =>
{
    counter++;
    int asd = Random.Range(1,4);
    GameObject pf_enemy = Master.GetEnemyPrefabB
yID("01");
    GameObject go_enemy = NGUITools.AddChild(Mas
ter.Gameplay.enemiesRoot, pf_enemy);
    go_enemy.transform.localPosition = new Vecto
r3(Master.Gameplay.outOfScreenPos[2].localPosition.x, Ma
ster.Lane.positionOfLane[asd].localPosition.y - 30, 0);
    go_enemy.GetComponent<EnemyController>().sta
tus.CurrentLane = asd;
}

```

```

        Master.Lane.SetCharacterAtLane(go_enemy, asd
);
        if(counter == 5){
            level += 1;
            Debug.Log("Level = "+level);
            counter = 0;
        }
        sequenceIndexOfWave++;
        if (sequenceIndexOfWave >= 10)
        {
            if (!isCheckingEnemyExistToStartNewWave)
            {
                CheckEnemyExistAndStartNewWave(waveI
ndex, sequenceIndexOfWave);
            }
        }
        else
        {
            InitEnemy(waveIndex, sequenceIndexOfWave
);
            jumlahMonster += 1;
            Debug.Log("Jumlah Monster Cinta = "+
jumlahMonster);
        }
    }
}

```

Kode Sumber 4.4 Durasi Spawn NPC Musuh Tsukamoto

4.4.2. Durasi Spawn NPC Sugeno

Pada bagian ini membahas mengenai untuk pengaturan Durasi Spawn NPC Musuh dengan metode Fuzzy Sugeno yaitu Durasi Banyak, Durasi Sedang dan Durasi Sedikit. JedaSpawn NPC Musuh akan semakin lambat atau menjadi lebih lama dikarenakan durasi setelah kemuncul NPC musuh yang pertama dengan kemunculan NPC Musuh berikutnya adalah banyak dan sebaliknya, JedaSpawn NPC Musuh akan semakin cepat atau menjadi lebih padat dikarenakan durasi setelah kemuncul NPC musuh yang pertama dengan kemunculan NPC Musuh berikutnya adalah sedikit. Berikut ini adalah script untuk Durasi Spawn NPC Musuh Sugeno dalam kelas LevelController.cs:

```

int waktuspaawnsugeno;
    //hategeno
    if (k > 0) {
        waktuspaawnsugeno = 20;
    } else if (k > 0.1) {
        waktuspaawnsugeno = 17;
    } else if (k > 0.4) {
        waktuspaawnsugeno = 15;
    } else if (k > 0.5) {
        waktuspaawnsugeno = 13;
    }else if ( k > 0.6 ){
        waktuspaawnsugeno = 11;
    }else if ( k > 0.7 ){
        waktuspaawnsugeno = 9;
    }else if ( k > 0.8 ){
        waktuspaawnsugeno = 7;
    }else if ( k > 0.9 ){
        waktuspaawnsugeno = 5;
    }
}
Master.WaitAndDo(waktuspaawnsukamoto, () =>
{
    counter++;
    int asd = Random.Range(1,4);
    GameObject pf_enemy = Master.GetEnemyPrefabB
yID("01");
    GameObject go_enemy = NGUITools.AddChild(Mas
ter.Gameplay.enemiesRoot, pf_enemy);
    go_enemy.transform.localPosition = new Vecto
r3(Master.Gameplay.outOfScreenPos[2].localPosition.x, Ma
ster.Lane.positionOfLane[asd].localPosition.y - 30, 0);
    go_enemy.GetComponent<EnemyController>().sta
tus.CurrentLane = asd;
    Master.Lane.SetCharacterAtLane(go_enemy, asd
);
    if(counter == 5){
        level += 1;
        Debug.Log("Level = "+level);
        counter = 0;
    }
    sequenceIndexOfWave++;
    if (sequenceIndexOfWave >= 10)
    {
        if (!isCheckingEnemyExistToStartNewWave)
        {
            CheckEnemyExistAndStartNewWave (waveI
ndex, sequenceIndexOfWave);
        }
    }
    else
    {
        InitEnemy(waveIndex, sequenceIndexOfWave
);
        jumlahMonster += 1;
        Debug.Log("Jumlah Monster Cinta = "+
jumlahMonster);
    }
}
}

```

Kode Sumber 4.5 Durasi Spawn NPC Musuh Sugeno

4.5. Uji Coba

Untuk mengetahui sejauh mana implementasi algoritma terhadap Durasi Spawn NPC Musuh, maka perlu dilakukan pengujian. Pengujian dilakukan beberapa kali dengan studi kasus yang berbeda dan di jalankan pada computer dengan spesifikasi sebagai berikut:

- e. Prosesor intel Core i5-3437U CPU@1.90GHz 2.40GHz
- f. HardDisk 120 GB
- g. RAM 4GB
- h. VGA IntelHD Graphics
- i. Keyboard
- j. Monitor 14”

4.6. Hasil Pengujian Durasi Spawn NPC Musuh pada Game Tower Deffense

Dari variasi variable masukan yang digambarkan pada perhitungan sebelumnya dapat diperoleh keluaran yang variatif, dimana selanjutnya dikelompokkan menjadi 3 tipe durasi Spawn NPC Musuh yaitu sedikit, sedang dan banyak. Respon keluaran fuzzy NPC Musuh terhadap variasi masing-masing input.

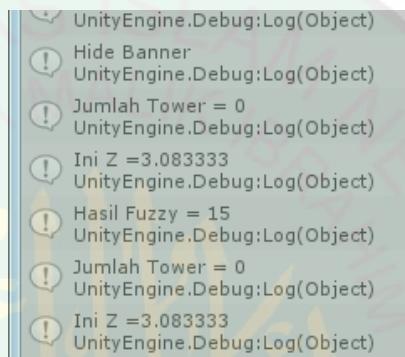
Tabel 4.1 Pengujian Fuzzy Tsukamoto dan Fuzzy Sugeno pada Durasi**Spawn NPC musuh dalam perhitungan manual excel**

No	Nilai Input			Output			
	Jumlah Musuh	Level	Jumlah Tower	Fuzzy Tsukamoto	Fuzzy Sugeno	Stopwatch Fuzzy Tsukamoto	Stopwatch Fuzzy Sugeno
1	0	0	0	15	15	15:23	15:12
2	4	1	3	15	15	15:23	15:12
3	7	2	4	15	15	15:23	15:12
4	9	3	6	12	15	12:23	15:12
5	12	4	7	9	15	9:12	15:12
6	15	5	9	9	15	9:12	15:12
7	18	6	12	6	9	6:10	9:12
8	21	7	15	3	3	3:9	3:9
9	23	7	15	3	3	3:9	3:9
10	24	8	15	3	3	3:9	3:9
11	27	9	15	3	3	3:9	3:9
12	30	10	15	1	1	1:02	1:02

Dari tabel 4.1 dapat dilihat bahwa semua output sudah sesuai dengan rule yang telah ditentukan. Durasi yang dihasilkan dari output tsukamoto tersebut adalah durasi spawn NPC musuh sedikit 35.00%, durasi spawn NPC musuh sedang 30.00% dan durasi spawn NPC musuh banyak 35.00% dari 12 data yang ada. Sedangkan Durasi yang dihasilkan dari output Sugeno tersebut adalah durasi

spawn NPC musuh sedikit 30.00%, durasi spawn NPC musuh sedang 20.00% dan durasi spawn NPC musuh banyak 50.00% dari 12 data yang ada.

Hasil pengujian perubahan durasi spawn NPC musuh tsukamoto juga dilakukan dengan menjalankan game. Perubahan durasi spawn NPC musuh ditampilkan pada console game engine dalam debug mode. Berikut beberapa hasil perubahan durasi spawn NPC musuh Tsukamoto:



```

UnityEngine.Debug.Log(Object)
Hide Banner
UnityEngine.Debug.Log(Object)
Jumlah Tower = 0
UnityEngine.Debug.Log(Object)
Ini Z =3.083333
UnityEngine.Debug.Log(Object)
Hasil Fuzzy = 15
UnityEngine.Debug.Log(Object)
Jumlah Tower = 0
UnityEngine.Debug.Log(Object)
Ini Z =3.083333
UnityEngine.Debug.Log(Object)

```

Gambar 4.5 Hasil Perubahan Durasi Spawn NPC Tsukamoto 15

detik

Pada gambar 4.5 menunjukkan durasi spawn NPC musuh Tsukamoto dengan nilai input Jumlah Tower = 0, Jumlah Musuh = 0 dan level = 0 adalah 15 detik dari nilai defuzzyifikasi $Z = 3.083333$.



```

Level = 2
UnityEngine.Debug.Log(Object)
Jumlah Tower = 6
UnityEngine.Debug.Log(Object)
Ini Z =4.190938
UnityEngine.Debug.Log(Object)
Hasil Fuzzy = 12
UnityEngine.Debug.Log(Object)
Jumlah Tower = 6
UnityEngine.Debug.Log(Object)
Ini Z =4.190938
UnityEngine.Debug.Log(Object)
Jumlah Monster = 6
UnityEngine.Debug.Log(Object)

```

Gambar 4.6 Hasil Perubahan Durasi Spawn NPC Tsukamoto 12

detik

Pada gambar 4.6 menunjukkan durasi spawn NPC musuh Tsukamoto dengan nilai input Jumlah Tower = 6, Jumlah Musuh = 6 dan level = 2 adalah 12 detik dari nilai defuzzyifikasi $Z = 4,190938$

```

    ✓ UnityEngine.Debug:Log(Object)
    ! Ini Z =4.845011
    UnityEngine.Debug:Log(Object)
    ! Jumlah Monster = 9
    UnityEngine.Debug:Log(Object)
    ! Jumlah Tower = 8
    UnityEngine.Debug:Log(Object)
    ! Jumlah Tower = 8
    UnityEngine.Debug:Log(Object)
    ! Ini Z =5.009523
    UnityEngine.Debug:Log(Object)
    ! Hasil Fuzzy = 9
    UnityEngine.Debug:Log(Object)
  
```

Gambar 4.7 Hasil Perubahan Durasi Spawn NPC Tsukamoto 9

detik

Pada gambar 4.7 menunjukkan durasi spawn NPC musuh Tsukamoto dengan nilai input Jumlah Tower = 8, Jumlah Musuh = 9 dan level = 3 adalah 9 detik dari nilai defuzzyifikasi $Z = 5,009523$

```

    ! Jumlah Tower = 12
    UnityEngine.Debug:Log(Object)
    ! Jumlah Tower = 12
    UnityEngine.Debug:Log(Object)
    ! Ini Z =6.016548
    UnityEngine.Debug:Log(Object)
    ! Hasil Fuzzy = 6
    UnityEngine.Debug:Log(Object)
    ! Jumlah Tower = 12
    UnityEngine.Debug:Log(Object)
    ! Ini Z =6.016548
    UnityEngine.Debug:Log(Object)
    ! Jumlah Monster = 17
    UnityEngine.Debug:Log(Object)
  
```

Gambar 4.8 Hasil Perubahan Durasi Spawn NPC Tsukamoto 6

detik

Pada gambar 4.8 menunjukkan durasi spawn NPC musuh Tsukamoto dengan nilai input Jumlah Tower = 12, Jumlah Musuh = 17 dan level = 5 adalah 6 detik dari nilai defuzzyifikasi $Z = 6,016548$

Hasil pengujian perubahan durasi spawn NPC musuh Sugeno juga dilakukan dengan menjalankan game. Perubahan durasi spawn NPC musuh ditampilkan pada console game engine dalam debug mode. Berikut beberapa hasil perubahan durasi spawn NPC musuh Sugeno:

```
Jumlah Tower = 0
UnityEngine.Debug:Log(Object)
Ini k =1
UnityEngine.Debug:Log(Object)
Hasil Fuzzy = 15
UnityEngine.Debug:Log(Object)
Jumlah Tower = 0
UnityEngine.Debug:Log(Object)
Ini k =1
UnityEngine.Debug:Log(Object)
Jumlah Tower = 1
UnityEngine.Debug:Log(Object)
Jumlah Tower = 2
```

Gambar 4.9 Hasil Perubahan Durasi Spawn NPC Sugeno 15 detik

Pada gambar 4.9 menunjukkan durasi spawn NPC musuh Sugeno dengan nilai input Jumlah Tower = 0, Jumlah Musuh = 0 dan level = 0 adalah 15 detik dari nilai defuzzyifikasi $K = 1$.

```
UnityEngine.Debug:Log(Object)
Hasil Fuzzy = 15
UnityEngine.Debug:Log(Object)
Jumlah Tower = 5
UnityEngine.Debug:Log(Object)
Ini k =1
UnityEngine.Debug:Log(Object)
Jumlah Monster = 5
UnityEngine.Debug:Log(Object)
Level = 2
UnityEngine.Debug:Log(Object)
Jumlah Tower = 5
UnityEngine.Debug:Log(Object)
```

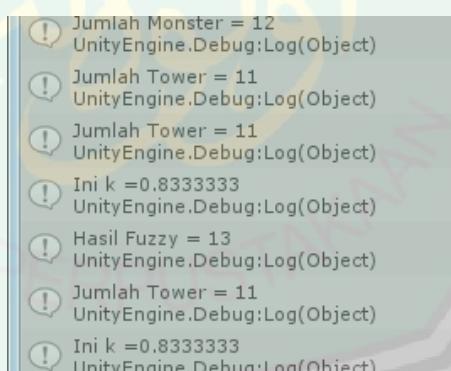
Gambar 4.10 Hasil Perubahan Durasi Spawn NPC Sugeno 15 detik

Pada gambar 4.10 menunjukkan durasi spawn NPC musuh Sugeno dengan nilai input Jumlah Tower = 5, Jumlah Musuh = 5 dan level = 2 adalah 15 detik dari nilai defuzzyifikasi K = 1



Gambar 4.11 Hasil Perubahan Durasi Spawn NPC Sugeno 15 detik

Pada gambar 4.11 menunjukkan durasi spawn NPC musuh Sugeno dengan nilai input Jumlah Tower = 8, Jumlah Musuh = 8 dan level = 3 adalah 15 detik dari nilai defuzzyifikasi K = 1



Gambar 4.12 Hasil Perubahan Durasi Spawn NPC Sugeno 13 detik

Pada gambar 4.12 menunjukkan durasi spawn NPC musuh Sugeno dengan nilai input Jumlah Tower = 11, Jumlah Musuh = 12 dan level = 4 adalah 13 detik dari nilai defuzzyifikasi K = 0,833333

```
UnityEngine.Debug.Log(Object)
Level = 5
UnityEngine.Debug.Log(Object)
Jumlah Tower = 12
UnityEngine.Debug.Log(Object)
Ini k =0.6666667
UnityEngine.Debug.Log(Object)
Hasil Fuzzy = 9
UnityEngine.Debug.Log(Object)
Jumlah Tower = 12
UnityEngine.Debug.Log(Object)
Ini k =0.6666667
UnityEngine.Debug.Log(Object)
```

Gambar 4.13 Hasil Perubahan Durasi Spawn NPC Sugeno 9 detik

Pada gambar 4.13 menunjukkan durasi spawn NPC musuh Sugeno dengan nilai input Jumlah Tower = 12, Jumlah Musuh = 15 dan level = 5 adalah 13 detik dari nilai defuzzyifikasi K = 0,66667

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Berdasarkan game yang telah dibuat dan uji coba yang telah dilakukan, maka dapat ditarik kesimpulan yaitu Algoritma Fuzzy Tsukamoto memiliki nilai yang lebih bervariatif dan efektif dibandingkan dengan Algoritma Fuzzy Sugeno yang monoton pada NPC musuh untuk menentukan Durasi Spawn NPC musuh. Berdasarkan hasil uji coba dari 12 data yang berbeda, durasi yang dihasilkan dari menggunakan Fuzzy Tsukamoto tersebut adalah durasi spawn NPC musuh sedikit 35.00%, durasi spawn NPC musuh sedang 30.00% dan durasi spawn NPC musuh banyak 35.00% dari 12 data yang ada. Sedangkan durasi yang dihasilkan dari menggunakan Fuzzy Sugeno tersebut adalah durasi spawn NPC musuh sedikit 30.00%, durasi spawn NPC musuh sedang 20.00% dan durasi spawn NPC musuh banyak 50.00% dari 12 data yang ada.

5.2. Saran

Berdasarkan kesimpulan diatas, terdapat beberapa saran untuk pengembangan game ini selanjutnya:

1. Penambahan Inputan lebih banyak lagi agar game berjalan optimal
2. Penambahan Jumlah Lane pada Game sehingga dapat menampung lebih banyak Spawn enemy dan meningkatkan tingkat kesulitan bermain game.

3. Bisa menggunakan metode lainnya yang lebih stabil dalam menentukan waktu kemunculan musuh karena metode Fuzzy Sugeno sangat tidak seimbang berdasarkan peningkatan level permainan.



DAFTAR PUSTAKA

Sumber Buku

- Anggani Sudono, 2000. *Sumber Belajar dan Alat Permainan*. Jakarta: Grasindo
- Ardiansyah, Firdan. 2011. *Pengenalan Dasar Android Programming*. Jakarta: Biraynara
- Annisa, Kuswardayan, I. & Sunaryono, D., 2013. "Rancang Bangun Chomical : Game Tower Defense dengan Menerapkan Eksperimen Kimia". Surabaya: Jurnal Teknik ITS
- Arif S Sadiman. 1993. *Media Pendidikan: Pengertian, Pengembangan, dan Pemanfaatannya*. RajaGrafindo Persada. Jakarta.
- Avery, P., Togelius, J., Alistar, E. & van Leeuwen, R. P., 2011. Computational intelligence and Tower defence games. 2011 IEEE Congress of Evolutionary Computation (CEC), pp. 1084-1091.
- Craig, W., Reynolds. 1999. "Steering Behaviors For Autonomous Characters." Boulevard, California , n.d., 763-782.
- Danim, Sudirman, 2008, *Media Komunikasi Pendidikan; Pelayanan Profesional Pembelajaran dan Mutu Hasil Belajar [Proses Belajar Mengajar di Perguruan Tinggi]*, Jakarta: Bumi Aksara.
- Gansala, Fauzadin, 2016, *Perilaku Non Playable Character (NPC) Musuh Pada Game Sepeda Menggunakan Fuzzy State Machine (Fusm)*, Malang :

Jurusan Teknik Informatika Fakultas Sains Dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim

Henry, Samuel. (2010). Cerdas dengan Game: Panduan Praktis Bagi Orangtua dalam Mendampingi Anak Bermain Game. Jakarta: PT. Gramedia Pustaka Utama.

Irshad, S. & Rohaya Bt Awang Rambli, D., 2014. User experience of mobile augmented reality: A review of studies. 2014 3rd International Conference on User Science and Engineering (i-USer), pp. 125-130.

Joanpere, M., Gorgorio, N. & Albarracin, L. 2015. "Mathematics Learning Opportunities When Playing a Tower Defense Game". Spanyol: Universitat Autònoma de Barcelona

Kusumadewi, S., Purnomo, H. 2010. *Aplikasi Logika Fuzzy untuk Pendukung Keputusan Edisi 2*. Graha Ilmu. Yogyakarta.

Rona G P. 2013. Perancangan Dan Pembuatan Animasi 2d "Kerusakan Lingkungan" Dengan Teknik Masking. Jurnal Ilmiah DASI. 14(4): 54-57.

Sudarwanto, Budianto, A. & Yohannes, Y. n.d., 2013. "Rancang Bangun Aplikasi Edu Game Selamatkan Hutan Indonesia Berbasis UNITY 3D". Palembang: Program Studi Informatika STMIK GI MDP Palembang

Sutoyo, R., Winata, D., Oliviani, K. & Martadinata, D. 2015. Dynamic Difficult Adjustment in Tower Defence. 2015 ICCSCI International Conference on Computer Science and Computational Intelligence, pp. 435 – 444

Sumber Internet

Dugan, GAMASUTRA. 2007. Internet

https://web.archive.org/web/20130804175159/http://www.gamasutra.com/view/feature/1728/slamdance_postcolumbine__.php. Diakses 28 Mei 2019.

Kompasiana. Dampak - Dampak Positif dan Negatif dari Bermain Video Game . 2015. Internet.

<https://www.kompasiana.com/martinhn/55f69fe8957a618109258af6/dampak-dampak-positif-dan-negatif-dari-bermain-video-game>. Diakses pada 28 Mei 2019.

Muslim Al Atsari, Abu. Jihad Melawan Setan. 2010 . Internet

<https://almanhaj.or.id/2721-jihad-melawan-setan.html>. Diakses 28 Mei 2019

Nacke, L., THE ACAGAMIC. 2014 . Internet

<http://www.acagamic.com/courses/infr1330-2014/the-formal-Systems-of-games-and-game-design-atoms>. Diakses 28 Mei 2019.

Rutkoff, Strategy Game Pits Players Against Dekstop Invasion - WSJ. 2007.

Internet

<http://www.wsj.com/articles/SB117987060189311315> Diakses 28 Mei 2019.

