

**IMPLEMENTASI METODE HIERARCHICAL FINITE STATE  
MACHINE PADA SISTEM ENEMY UNTUK PERMAINAN  
"VOID ASTRALE"**

**SKRIPSI NON REGULER**

*Diajukan memenuhi salah satu syarat mencapai derajat Sarjana*

*Program Studi Teknologi Informasi*



Disusun oleh

**ANDREY ADHYASTA**

**21.60.0124**

Kepada

**FAKULTAS ILMU KOMPUTER  
UNIVERSITAS AMIKOM YOGYAKARTA  
YOGYAKARTA**

**2025**

**IMPLEMENTASI METODE HIERARCHICAL FINITE STATE  
MACHINE PADA SISTEM ENEMY UNTUK PERMAINAN  
"VOID ASTRALE"**

**SKRIPSI NON REGULER**

*Diajukan memenuhi salah satu syarat mencapai derajat Sarjana  
Program Studi Teknologi Informasi*



Disusun oleh  
**ANDREY ADHYASTA**  
**21.60.0124**

Kepada  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS AMIKOM YOGYAKARTA**  
**YOGYAKARTA**  
**2025**

**HALAMAN PERSETUJUAN  
SKRIPSI NON REGULER**

**IMPLEMENTASI METODE HIERARCHICAL FINITE STATE MACHINE PADA  
SISTEM ENEMY UNTUK PERMAINAN "VOID ASTRALE"**

yang disusun dan diajukan oleh

**ANDREY ADHYASTA**

**21.60.0124**

telah disetujui oleh Dosen Pembimbing Skripsi  
pada tanggal 30 Desember 2024

**Dosen Pembimbing**



**Muhammad Fairul Filza, M.Kom.**

**NIK. 190302332**

**HALAMAN PENGESAHAN  
SKRIPSI NON REGULER**

**IMPLEMENTASI METODE HIERARCHICAL FINITE STATE  
MACHINE PADA SISTEM ENEMY UNTUK PERMAINAN  
"VOID ASTRALE"**

yang disusun dan diajukan oleh

**ANDREY ADHYASTA**

**21.60.0124**

Telah dipertahankan di depan Dewan Penguji  
pada tanggal 30 Desember 2024

**Susunan Dewan Penguji**

**Nama Penguji**

**Tanda Tangan**

**Bhanu Sri Nugraha, S.Kom., M.Kom.**  
**NIK. 190302164**

**Raditya Wardhana, S.Kom., M.Kom.**  
**NIK. 190302208**

**Muhammad Fairul Filza, S.Kom., M.Kom.**  
**NIK. 190302332**



Skripsi ini telah diterima sebagai salah satu persyaratan  
untuk memperoleh gelar Sarjana Komputer  
Tanggal, 30 Desember 2024

**DEKAN FAKULTAS ILMU KOMPUTER**



**Hanif Al Fatta, S.Kom., M.Kom., Ph.D.**  
**NIK. 190302096**

## HALAMAN PERNYATAAN KEASLIAN SKRIPSI

Yang bertandatangan di bawah ini,

Nama mahasiswa : **Andrey Adhyasta**

NIM : **21.60.0124**

Menyatakan bahwa Skripsi dengan judul berikut:

**IMPLEMENTASI METODE HIERARCHICAL FINITE STATE MACHINE PADA SISTEM ENEMY UNTUK PERMAINAN "VOID ASTRALE"**

Dosen Pembimbing : **Muhammad Fairul Filza, S.Kom., M.Kom.**

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya.
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Dosen Pembimbing.
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini.
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta.
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang sudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi.

Yogyakarta, 30 Desember 2024

Yang Menyatakan,



Andrey Adhyasta

## KATA PENGANTAR

Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa, atas rahmat, kasih sayang, serta kesempatan yang diberikan-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul Implementasi Metode *Hierarchical Finite State Machine* pada Sistem *Enemy* untuk Permainan "VOID Astrale". Tidak luput dari dukungan berbagai pihak yang telah memberikan bantuan baik secara langsung maupun tidak langsung dalam proses penyusunan skripsi ini. Ucapan terima kasih penulis haturkan kepada:

1. Seluruh anggota keluarga peneliti yang telah memberikan segala dukungan dan membimbing dalam segala keputusan yang penulis ambil.
2. Seluruh anggota Power of Two Studio yang selalu tekun dalam proses pengerjaan produk.
3. Seluruh anggota Retartos yang telah menemani dan membantu perkembangan penulis dari masih kecil hingga saat ini.
4. Kelas 21BCIT yang selalu bersama dari awal perkuliahan hingga akhir.
5. Teman-teman yang telah memberikan dukungan dan semangat kepada penulis.
6. Bapak Fairul Filza sebagai dosen pembimbing skripsi yang telah membimbing dari mata kuliah konsentrasi hingga penyusunan skripsi sehingga banyak ilmu yang berguna dan bermanfaat.

Semoga dengan adanya skripsi ini dapat memberikan manfaat dan sebagai informasi maupun pengetahuan bagi pembaca. Penulis menyadari adanya kekurangan dalam penulisan. Kritik dan saran dapat diberikan sebagai masukan untuk penyempurnaan dalam penulisan di masa yang akan datang.

Yogyakarta, 30 Desember 2024

Andrey Adhyasta

## DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PERSETUJUAN.....	ii
HALAMAN PENGESAHAN.....	iii
HALAMAN PERNYATAAN KEASLIAN SKRIPSI .....	iv
KATA PENGANTAR.....	v
DAFTAR ISI.....	vi
DAFTAR GAMBAR .....	viii
DAFTAR TABEL.....	x
DAFTAR LAMPIRAN.....	xi
DAFTAR LAMBANG DAN SINGKATAN.....	xii
DAFTAR ISTILAH .....	xiii
INTISARI.....	xiv
ABSTRACT .....	xv
BAB I PENDAHULUAN .....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah.....	2
1.3. Batasan Masalah .....	2
1.4. Tujuan Penelitian .....	2
BAB II TEORI DAN PERANCANGAN .....	2
2.1. Dasar Teori.....	2
2.1.1. Game.....	2
2.1.2. Video Game .....	2
2.1.3. Genre .....	3
2.1.4. Game Dua-Dimensi (2D).....	3
2.1.5. Game Engine .....	4
2.1.6. Unity .....	4
2.1.7. C-Sharp.....	4
<b>2.1.7.1. Object Oriented Programming (OOP).....</b>	<b>4</b>
2.1.8. Visual Studio .....	7
2.1.9. Event Function.....	7
2.1.10. Mono-Behaviour.....	9
2.1.11. Scriptable-Object.....	9

2.1.12.	Artificial-Intelligence (AI)	9
2.1.13.	Hierarchical Finite State Machine (HFMSM)	11
2.2.	Teori Analisis Kebutuhan	12
2.2.1.	Brief Produksi	12
2.2.2.	Referensi	12
2.2.2.1.	<b>Monster Hunter Series</b>	<b>13</b>
2.2.2.2.	<b>Granblue Fantasy</b>	<b>14</b>
2.2.2.3.	<b>Blasphemous</b>	<b>15</b>
2.2.2.4.	<b>Touhou Project</b>	<b>15</b>
2.2.3.	Kebutuhan Fungsional	16
2.2.4.	Kebutuhan Non Fungsional	17
2.3.	Analisis Aspek Produksi	18
2.3.1.	Aspek Kreatif	18
2.3.2.	Aspek Teknis	20
2.4.	Pra-Produksi	22
2.4.1.	Game Design Document (GDD)	23
2.4.2.	Rancangan Teknis Sistem	23
<b>BAB III IMPLEMENTASI</b>		<b>28</b>
3.1.	Produksi	28
3.1.1.	Pembuatan Event Listener	28
3.1.2.	Pembuatan Enemy Status	33
3.1.3.	Pembuatan States dan Substates	39
3.1.4.	Pembuatan Condition	44
3.1.5.	Pembuatan Enemy Behaviour	45
3.1.6.	Pembuatan Enemy Scanner	48
3.1.7.	Pembuatan Attack Script	50
3.2.	Pasca-Produksi	51
3.2.1.	Debug And Testing	51
3.3.	Evaluasi	53
<b>BAB IV PEMBAHASAN</b>		<b>57</b>
4.1.	Kesimpulan	57
4.2.	Saran	57
<b>DAFTAR PUSTAKA</b>		<b>58</b>
<b>LAMPIRAN</b>		<b>59</b>



## DAFTAR GAMBAR

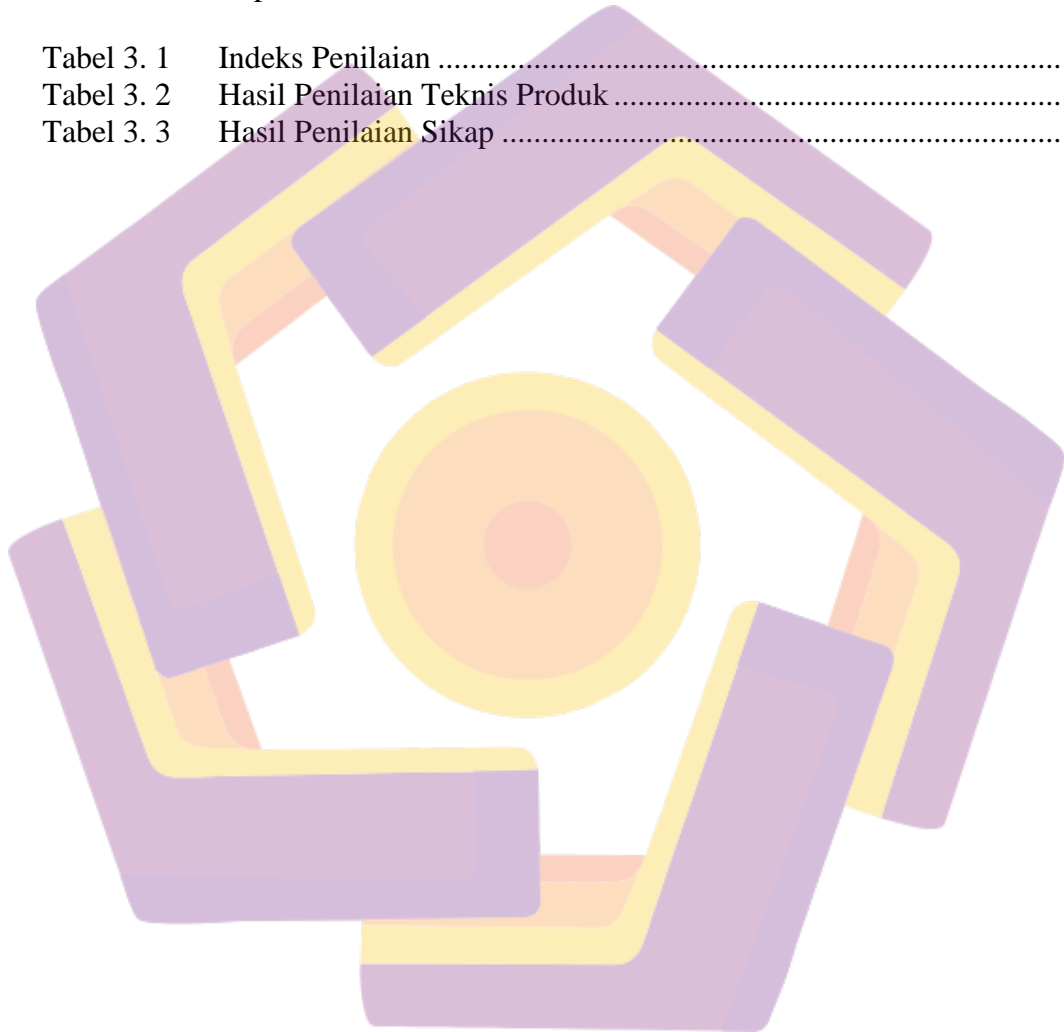
Gambar 2. 1	Diagram <i>Class</i> dan <i>Object</i> .....	5
Gambar 2. 2	Diagram <i>Encapsulation</i> .....	5
Gambar 2. 3	Diagram <i>Inheritance</i> .....	6
Gambar 2. 4	Diagram <i>Polymorphism</i> .....	6
Gambar 2. 5	Diagram <i>Abstraction</i> .....	7
Gambar 2. 6	<i>Flowchart execution order</i> dalam <i>Unity</i> .....	8
Gambar 2. 7	Alur sederhana <i>state machine</i> .....	11
Gambar 2. 8	Perbedaan diagram FSM dengan HFSM.....	12
Gambar 2. 9	<i>Gameplay</i> dari game “Monster Hunter Generation Ultimate” ....	13
Gambar 2. 10	<i>Gameplay</i> dari game “Granblue Fantasy”.....	14
Gambar 2. 11	<i>Mode Bar</i> pada game “Granblue Fantasy.”.....	14
Gambar 2. 12	<i>Gameplay</i> pada game “Blasphemous”.....	15
Gambar 2. 13	<i>Gameplay</i> pada game “Touhou 7 : Perfect Cherry Blossom”.....	15
Gambar 2. 14	Lapisan darah pada salah satu bos bernama Yukari Yakumo ....	16
Gambar 2. 15	Diagram <i>sequence</i> dari <i>Event Listener</i> .....	23
Gambar 2. 16	Diagram <i>class inheritance</i> pada <i>Enemy Status</i> .....	24
Gambar 2. 17	Diagram aktifitas <i>Enemy Status</i> sebagai <i>Hub</i> antar komponen...	24
Gambar 2. 18	Diagram HFSM pada sistem <i>Enemy Behaviour</i> .....	26
Gambar 2. 19	Diagram State pada <i>Main Action State</i> .....	26
Gambar 2. 20	Struktur dari <i>Enemy Scanner</i> .....	27
Gambar 3. 1	GDD terkait hal teknis pada <i>enemy</i> .....	28
Gambar 3. 2	Pembuatan <i>class</i> yang berfungsi sebagai <i>event</i> .....	29
Gambar 3. 3	Pembuatan <i>wrapper</i> dalam bentuk <i>class</i> untuk <i>event</i> .....	29
Gambar 3. 4	Struktur <i>SO_VoidGameEvent</i> dan <i>SO_ParameterGameEvent</i> .....	30
Gambar 3. 5	Tampilan <i>SO_VoidGameEvent</i> dan <i>SO_ParameterGameEvent</i> .....	31
Gambar 3. 6	Isi dari <i>EventListenerComponent</i> .....	32
Gambar 3. 7	Tampilan <i>EventListenerComponent</i> pada <i>inspector</i> .....	32
Gambar 3. 8	<i>CustomVriable</i> , <i>Interface INumericVariable</i> dan <i>IBoolVariable</i> .....	33
Gambar 3. 9	Pembuatan <i>class</i> yang menyimpan nilai tipe data sederhana.....	34
Gambar 3. 10	Tampilan variabel yang diturunkan dari <i>BaseStatus</i> .....	35
Gambar 3. 11	Variabel yang dapat dimodifikasi.....	35
Gambar 3. 12	Variabel yang tidak dapat dimodifikasi ( <i>read-only</i> ).....	36
Gambar 3. 13	Fungsi yang mengirimkan sinyal melalui <i>event</i> .....	37
Gambar 3. 14	Fungsi yang mengurus <i>rage mode</i> pada <i>enemy</i> .....	38
Gambar 3. 15	Tampilan <i>Enemy Status Effect Container</i> pada <i>inspector</i> .....	38
Gambar 3. 16	Fungsi dari pemberian sebuah efek kepada <i>enemy</i> .....	39
Gambar 3. 17	Struktur dan fungsi dari efek <i>rage</i> .....	39
Gambar 3. 18	Struktur <i>states</i> pada <i>enemy</i> .....	40
Gambar 3. 19	Struktur <i>class</i> abstrak pada <i>substate</i> untuk <i>enemy</i> .....	41
Gambar 3. 20	Struktur <i>FixedState</i> dan <i>SO_DynamicState</i> .....	41
Gambar 3. 21	Tampilan <i>SO_DynamicStates</i> dan <i>Condition</i> pada <i>inspector</i> .....	42

Gambar 3. 22	Struktur <i>SO_Base_Attack_Fixed</i> .	43
Gambar 3. 23	Tampilan <i>substate</i> dari <i>SO_Base_Attack_Fixed</i> .	43
Gambar 3. 24	Struktur serangan yang memiliki kondisi.	44
Gambar 3. 25	Tampilan <i>Condition</i> pada <i>inspector</i> .	44
Gambar 3. 26	Fungsi dari <i>condition</i> yang membandingkan dua nilai.	45
Gambar 3. 27	<i>Flowchart Sequence State</i> pada <i>enemy</i> .	46
Gambar 3. 28	Fungsi yang menjalankan alur <i>Main Action State</i> .	46
Gambar 3. 29	Fungsi yang mengeksekusi dan mengganti <i>substate</i> .	47
Gambar 3. 30	Fungsi pengecekan kondisi <i>SO_DynamicState</i> .	47
Gambar 3. 31	Respon berbagai macam kondisi internal.	48
Gambar 3. 32	Contoh tampilan <i>EnemyBehaviour</i> pada <i>inspector</i> .	48
Gambar 3. 33	Tampilan <i>Scanner</i> pada <i>inspector</i> .	49
Gambar 3. 34	Proses <i>scanning</i> yang terjadi pada <i>LateUpdate()</i> .	49
Gambar 3. 35	Fungsi perhitungan waktu toleransi dan pengiriman sinyal.	50
Gambar 3. 36	<i>Interface</i> untuk objek yang dapat terkena <i>damage</i> .	50
Gambar 3. 37	Penggunaan <i>Interface IDamageable</i> pada suatu komponen.	50
Gambar 3. 38	Struktur dan fungsi dari <i>AttackHandler</i> .	51
Gambar 3. 39	<i>Bug</i> dimana <i>enemy</i> hanya terdiam.	52
Gambar 3. 40	Menambahkan pengecekan <i>case</i> pada <i>OnDamage()</i> .	52
Gambar 3. 41	Respon dari sinyal <i>OnHealthChange</i> .	53



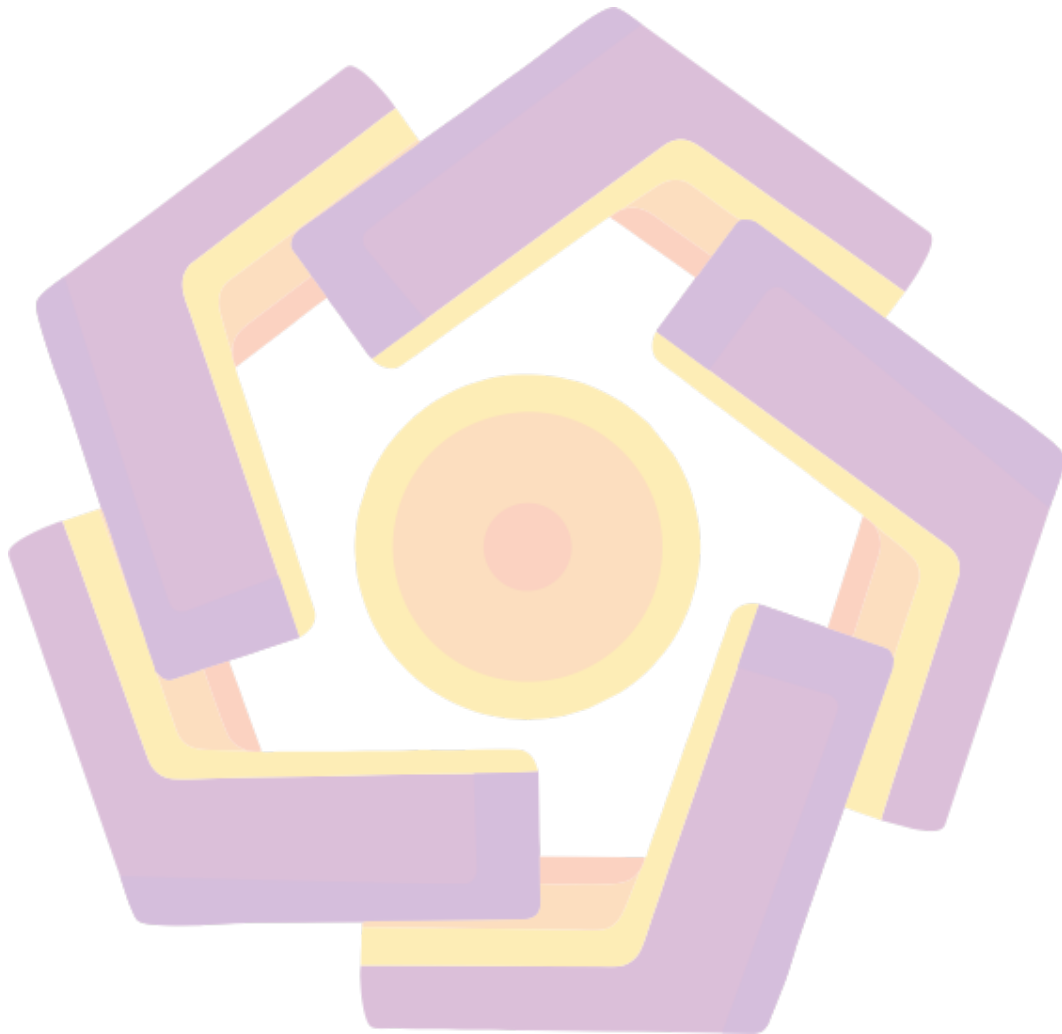
## DAFTAR TABEL

Tabel 2. 1	Spesifikasi <i>hardware</i> untuk pengembangan <i>game</i> .....	17
Tabel 2. 2	Spesifikasi <i>software</i> untuk pengembangan <i>game</i> .....	18
Tabel 2. 3	Aspek Kreatif .....	19
Tabel 2. 4	Aspek Teknis.....	20
Tabel 3. 1	Indeks Penilaian .....	54
Tabel 3. 2	Hasil Penilaian Teknis Produk .....	55
Tabel 3. 3	Hasil Penilaian Sikap .....	56



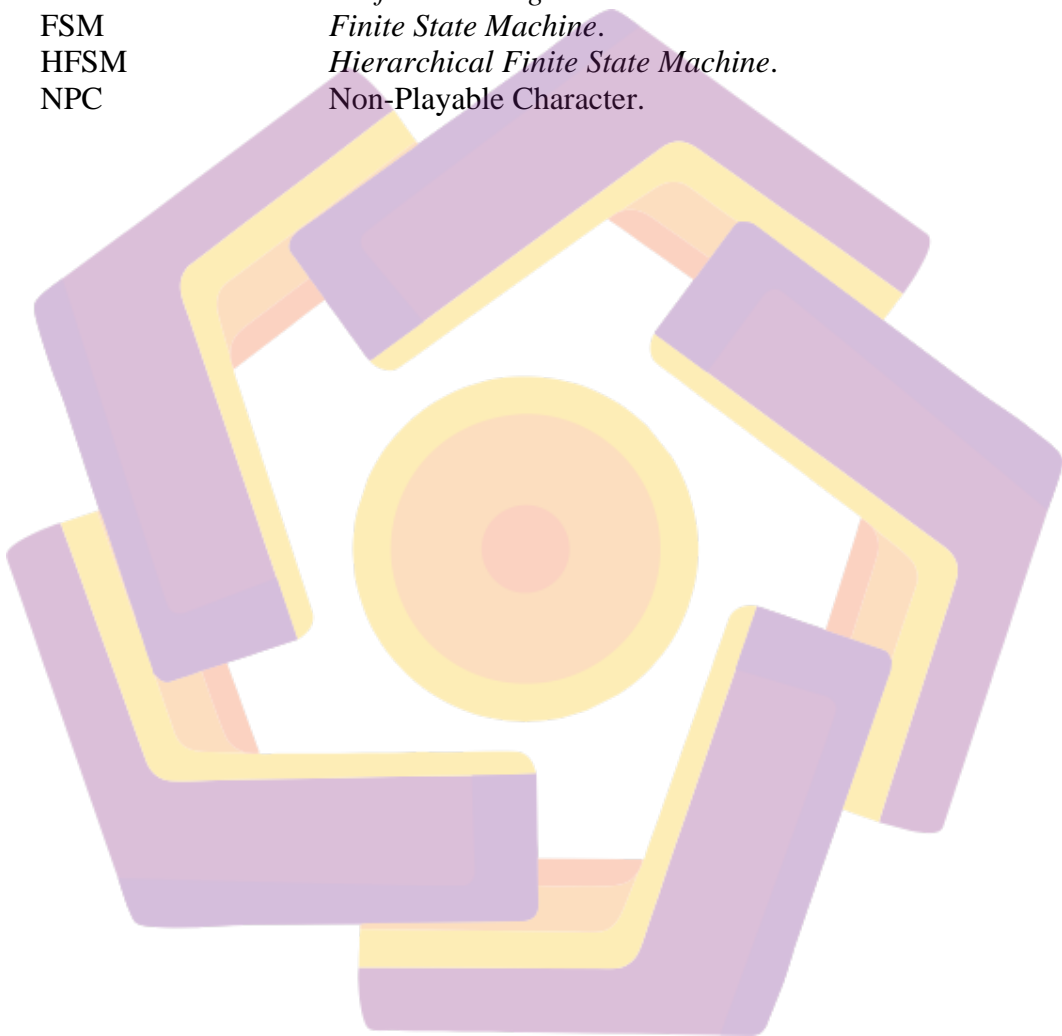
## DAFTAR LAMPIRAN

Lampiran 1	Hasil Penilaian Produk dan Kegiatan Pandawa.....	59
Lampiran 2	<i>Game Design Document</i> (GDD).....	60
Lampiran 3	<i>Screenshot gameplay</i> “VOID Astrale” .....	79



## DAFTAR LAMBANG DAN SINGKATAN

<i>.jpeg</i>	<i>Format gambar Joint Photographic Expert Group.</i>
<i>.png</i>	<i>Format gambar Portable Network Graphics.</i>
2D	Dua dimensi.
3D	Tiga dimensi.
AI	<i>Artificial Intelligence.</i>
FSM	<i>Finite State Machine.</i>
HFSM	<i>Hierarchical Finite State Machine.</i>
NPC	Non-Playable Character.



## DAFTAR ISTILAH

<i>Argument</i>	Nilai <i>parameter</i> yang digunakan dalam sebuah fungsi.
<i>Asset</i>	Kebutuhan yang dipakai dalam proyek.
<i>Bug</i>	Masalah dalam kode yang membuat instruksi memberi hasil yang abnormal atau tidak sesuai yang diharapkan.
<i>C#</i>	Bahasa pemrograman C-Sharp.
<i>C++</i>	Bahasa pemrograman C-Plus-plus.
<i>Callback</i>	Fungsi yang dipanggil dalam sebuah <i>argument</i> .
<i>Class</i>	Cetakan atau <i>blueprint</i> dari suatu objek.
<i>Drag and Drop</i>	Proses menahan klik pada objek lalu melepas klik.
<i>Event</i>	Peristiwa yang terjadi dalam program yang dapat dipicu oleh berbagai hal.
<i>File</i>	Kumpulan data yang disimpan di komputer.
<i>Folder</i>	Tempat menyimpan berbagai macam kelompok file.
<i>GameObject</i>	Entitas dasar yang merepresentasikan sebuah objek.
<i>Input</i>	Proses masuknya data atau informasi ke dalam sistem.
<i>Inspector</i>	Panel untuk menampilkan dan mengedit nilai properti dari objek atau komponen.
<i>Instance</i>	Representasi objek dari sebuah <i>class</i>
<i>Instantiation</i>	Proses pembuatan sebuah objek ke dalam <i>scene</i> .
<i>Interface</i>	Kontrak yang harus diimplementasikan oleh penggunanya.
<i>Java</i>	Bahasa pemrograman Java.
<i>Komponen</i>	Bagian-bagian modular yang ditambahkan ke <i>GameObject</i> yang memberi fungsi tertentu.
<i>Koordinat</i>	Bilangan lokasi suatu titik dalam ruang.
<i>Library</i>	Kumpulan kode yang sudah ditulis sebelum pengembangan.
<i>Lifecycle</i>	Siklus hidup dari suatu sistem.
<i>Method</i>	Fungsi dari sebuah kode.
<i>Parameter</i>	Variabel yang dipakai dalam sebuah fungsi.
<i>Physic</i>	Proses perhitungan fisika seperti gravitasi dan gaya.
<i>Prefab</i>	Cetakan atau <i>blueprint</i> dari satu atau kumpulan <i>GameObject</i> .
<i>Program</i>	Kumpulan instruksi atau kode yang ditulis untuk menyelesaikan tugas tertentu yang dieksekusi oleh komputer.
<i>Rendering</i>	Proses menghasilkan gambar atau video.
<i>Scene</i>	Tempat menyusun dan mengatur <i>GameObject</i> .
<i>Script</i>	Kumpulan instruksi yang ditulis dalam bahasa pemrograman.
<i>Sprite</i>	Objek grafis dua-dimensi yang digunakan dalam <i>video game</i> .
<i>Wrapper</i>	Fungsi yang membungkus komponen atau <i>instance</i> lainnya.

## INTISARI

Dalam sebuah *game* atau permainan, terdapat aturan dan tujuan dalam permainan yang dibuat untuk para pemain yang akan menentukan para pemain dalam posisi kalah atau menang. *Game* sudah berkembang pesat hingga saat ini terdapat *game* dalam media digital yaitu *video game*. Umumnya dalam sebuah *video game* terdapat tantangan tambahan atau *obstacle* yang berperan sebagai penghalang bagi pemain dalam meraih kemenangan.

Tantangan dalam *video game* yang berjudul “VOID Astrale” berupa NPC (*Non-Playable Character*) yang berperan sebagai musuh. Untuk menciptakan pengalaman permainan yang imersif, NPC perlu memiliki sistem kecerdasan buatan atau AI (*Artificial Intelligence*) yang berperan sebagai aturan dasar dalam mengambil keputusan-keputusan tertentu. Perilaku kecerdasan buatan ini dapat diaplikasikan dengan metode *Hierarchical Finite State Machine* (HFSM).

Dengan memanfaatkan struktur HFSM, perilaku musuh diorganisasikan ke dalam beberapa tingkatan *state*. Metode ini memastikan musuh dapat merespons secara adaptif terhadap berbagai skenario dalam permainan, sehingga perilaku dari NPC dapat meningkatkan pengalaman permainan. Penelitian ini mencakup perancangan, implementasi, improvisasi dan pengujian sistem musuh berbasis HFSM. Penelitian ini bertujuan untuk memberi pengetahuan tentang bagaimana penerapan metode HFSM diimplementasikan di dalam *game* “VOID Astrale.”

**Kata kunci:** *Hierarchical Finite State Machine*, Permainan, Kecerdasan Buatan, *Unity*, *Scriptable Object*

## ABSTRACT

*In a game, there are rules and objectives that are designed for players to pursue and experience which determine whether they are winning or losing. Games have evolved significantly leading to the creation of digital media-based games known as video games. Generally, in video games, there are additional challenges or obstacles that serve as barriers for players in achieving victory.*

*Challenges in the video game titled "VOID Astrale" take the form of NPCs (Non-Playable Characters) that act as their enemies. To create an immersive gameplay experience, NPCs tend to have an artificial intelligence (AI) system that serves as their rule set for making decisions. This artificial intelligence behavior can be replicated using the Hierarchical Finite State Machine (HFSM) method.*

*By utilizing the HFSM structure, enemy behaviors could be organized into multiple state levels or state hierarchy. This method ensures that enemies can adaptively respond to various scenarios within the game, enhancing the overall player experience. This research covers the system design, implementation, improvisation, and testing of HFSM-based enemy systems. The aim of this research is to provide insights into how the HFSM method is applied in the game "VOID Astrale."*

**Keyword:** Hierarchical Finite State Machine, Game, Artificial Intelligence, Unity, Scriptable Object