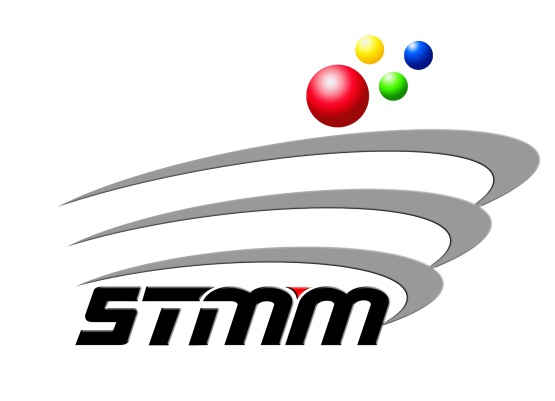
**LAPORAN PROJECT**

**UJIAN AKHIR SEMESTER (UAS)**

**GAME ARTIFICIAL INTELLEGENT (AI)**

**Disusun Sebagai Laporan Kegiatan**

**Mata Kuliah Game Artificial Intellegent (AI)**



Oleh :

**Ahmad Nuhisya Adillaumam**

NIM 02120147309

**Maulana Arba'i Dwi Putranto**

NIM 02120147285

**Revi Nur Ridwan**

NIM 02120147304

SEKOLAH TINGGI MULTI MEDIA MMTC

2023

**KATA PENGANTAR**

Puji syukur kami haturkan  kehadirat Allah Swt. yang telah melimpahkan rahmat dan hidayah-Nya sehingga kami bisa menyelesaikan Laporan kegiatan ujian akhir semester mata kuliah Game Artificial intelligent (AI).

Tidak lupa juga kami mengucapkan terima kasih kepada semua pihak yang telah turut memberikan kontribusi dalam penyusunan laporan pertanggung jawaban . Tentunya, tidak akan bisa maksimal jika tidak mendapat dukungan dari berbagai pihak.

Sebagai penyusun, kami menyadari bahwa masih terdapat kekurangan, baik dari penyusunan maupun tata bahasa penyampaian dalam laporan pertanggung jawaban ini. Oleh karena itu, kami dengan rendah hati menerima saran dan kritik dari pembaca agar kami dapat memperbaiki karya ilmiah ini.

Kami berharap semoga karya ilmiah yang kami susun ini memberikan manfaat dan juga inspirasi untuk pembaca.

Yogyakarta, November 2023

Penulis

**BAB I**

**PENDAHULUAN**

1. **Latar Belakang**

Game AI, atau kecerdasan buatan dalam game, mengacu pada penerapan perilaku cerdas dan proses pengambilan keputusan dalam video game. Game AI bertanggung jawab untuk mengendalikan karakter non-pemain (NPC), menciptakan pengalaman gameplay yang dinamis dan menantang, dan meningkatkan perendaman pemain secara keseluruhan. Berikut adalah beberapa aspek kunci dari Game Ai.

Game Yang akan di buat ini merupakan game populer yang biasa dikenal sebagai "tactical shooter". "Tactical shooter adalah genre permainan video yang menekankan strategi, kerjasama tim, dan taktik dalam penggunaan senjata untuk mencapai tujuan permainan. Pengembangan game ini terinspirasi dari game Counter Strike dimana player menjadi counter terorist yang menyelamatkan sandera. Beberapa bertimbangan kami memilih game ini sebagai berikut:

* Gameplay lebih umum/mudah dipahami dibanding game lain
* Melibatkan jumlah NPC yang banyak dan aktif bergerak

1. **Konsep**

| **Elemen** | **Deskripsi** |
| --- | --- |
| Goals | Menyelamatkan sandra dan membawanya ke markas pemain. |
| Interaction | Third person control (WASD & Mouse). |
| Obstacle | Terorist akan berpatroli dan mengumpulkan sandera ke markasnya. Terroist menyerang pemain jika pemain berada didekatnya. |
| Rules | Pemain harus menyelamatkan seluruh sandera. |

**BAB II**

**INSTALASI SOFTWARE PENGEMBANG**

* 1. Teknologi yang Digunakan
* [**Unity**](https://unity.com/), Merupakan lingkungan pengembangan perangkat lunak dan game engine yang memungkinkan pembuat game untuk membuat, mengelola, dan merancang aplikasi berbasis 2D maupun 3D.
* [**MonsterLove StateMachine**](https://github.com/thefuntastic/Unity3d-Finite-State-Machine)**,** Merupakan plugin untuk Unity yang menyediakan Finite State Machine (FSM) untuk membantu pengembang mengatur logika perpindahan keadaan dalam game dengan lebih terstruktur.
* [**Behavior Bricks**](https://assetstore.unity.com/packages/tools/visual-scripting/behavior-bricks-74816), Merupakan alat visual scripting di Unity yang memfasilitasi pengembangan game dengan memungkinkan pengembang untuk merancang logika gameplay menggunakan blok-blok visual dan menghubungkannya secara intuitif.
* [**Starter Assets - Third Person Character Controller**](https://assetstore.unity.com/packages/essentials/starter-assets-third-person-character-controller-urp-196526)**,** Adalah sebuah paket asset di Unity yang menyediakan karakter pengendali pihak ketiga (third-person) sebagai dasar untuk pengembangan game, termasuk kontrol karakter, animasi, dan fitur-fitur dasar lainnya.
* **FLOWCHART**. Flowchart adalah representasi grafis dari suatu proses atau alur kerja yang menggunakan simbol-simbol grafis untuk menggambarkan langkah-langkah dalam urutan tertentu. Tujuan utama dari flowchart adalah memberikan gambaran visual yang jelas dan mudah dimengerti tentang bagaimana suatu proses atau sistem bekerja. menggunakan simbol-simbol standar seperti persegi panjang (untuk merepresentasikan langkah-langkah atau aktivitas), oval (untuk merepresentasikan awal atau akhir dari suatu proses),
  1. Behaviour hostage (FSM)
* Ketika berada di dekat markas pemain, mereka akan menuju markas pemain.
* Ketika berada di dekat markas teroris, mereka akan menuju markas teroris.
* Jika berada di dekat teroris, mereka akan mengikuti teroris.
* Jika berada di dekat pemain, mereka akan mengikuti pemain.
* Jika tidak ada pemain atau teroris di sekitar mereka, mereka akan menuju area perlindungan terdekat.

Perilaku sandera dikendalikan oleh sebuah finite state machine atau FSM yang diimplementasikan menggunakan library **MonsterLove.StateMachine** di Unity. FSM mengelola berbagai keadaan seperti inisialisasi, idle, dan mengikuti target.

Mari kita bahas komponen-komponen kunci dari kode tersebut:

1. **Serialized Fields:**

[SerializeField] private NavMeshAgent agent;

[SerializeField] private float maxFollowDistance = 10f;

Bidang ini diserialkan dan diekspos di Unity Inspector. NavMeshAgent adalah komponen untuk pencarian jalur dan navigasi pada permukaan NavMesh. maxFollowDistance adalah jarak maksimum di dalamnya sandera akan mengikuti target.

1. **Private Fields:**

private Transform target;

private bool isAtHQ;

target adalah referensi ke target saat ini yang sedang diikuti sandera, dan isAtHQ adalah bendera boolean yang menunjukkan apakah sandera berada di markas besar.

1. **Nested Class**Driver**:**

public class Driver {

public StateEvent Update;

public StateEvent<collider> OnTriggerEnter;

}

Kelas bersarang ini mendefinisikan peristiwa (Update dan OnTriggerEnter) yang akan digunakan oleh state machine untuk berpindah antar keadaan.

1. **Enum**States**:**

public enum States

{

Init,

Idle,

FollowTarget

}

Enum ini mendefinisikan keadaan-keadaan yang mungkin dari state machine terbatas: Init, Idle, dan FollowTarget.

1. **Inisialisasi State Machine:**

StateMachine<States, Driver> fsm;

private void Awake()

{

fsm = new StateMachine<States, Driver>(this);

fsm.ChangeState(States.Init);

}

State machine diinisialisasi di dalam method Awake dengan keadaan awal diatur ke Init.

1. **Method Update:**

private void Update()

{

fsm.Driver.Update.Invoke();

}

Method Update dari state machine dipanggil di dalam lingkaran Update Unity.

1. **Method OnTriggerEnter:**

private void OnTriggerEnter(Collider collider) {

// ... (Lihat kondisi if untuk peralihan keadaan)

fsm.Driver.OnTriggerEnter.Invoke(collider);

}

Method ini dipanggil ketika Collider sandera berinteraksi dengan collider lain. Bergantung pada tag collider, state machine berpindah ke keadaan yang berbeda.

1. **Method FollowTarget\_Enter dan FollowTarget\_Update:**

private void FollowTarget\_Enter() {

// ... (Lihat kondisi untuk masuk ke keadaan FollowTarget)

}

private void FollowTarget\_Update() {

// ... (Lihat kondisi untuk memperbarui keadaan FollowTarget)

}

Method ini khusus untuk keadaan FollowTarget. FollowTarget\_Enter dipanggil saat memasuki keadaan, dan FollowTarget\_Update dipanggil selama keadaan untuk menangani perilaku tertentu, seperti menetapkan tujuan untuk NavMeshAgent dan memeriksa apakah target berada dalam jarak tertentu.

1. **Method Idle\_Enter:**

private void Idle\_Enter() {

target = null;

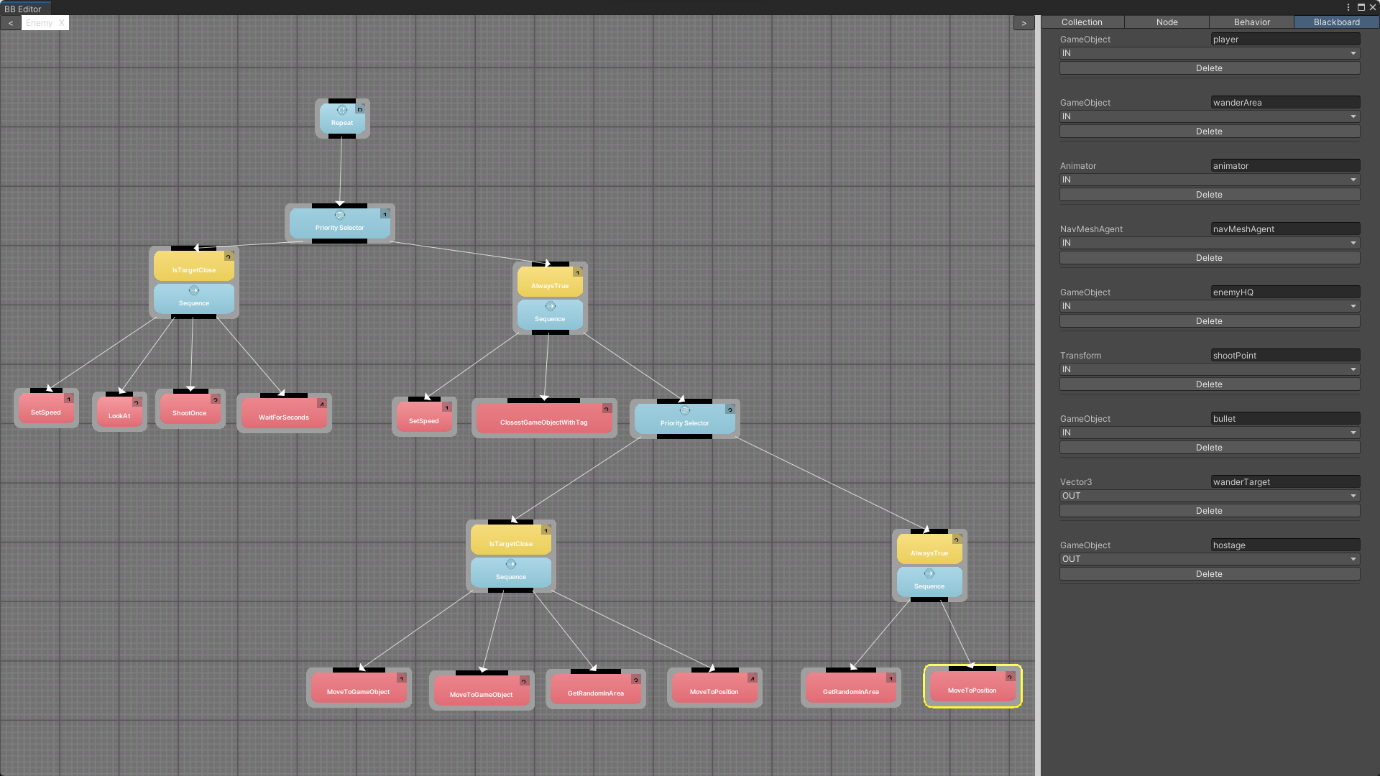
agent.ResetPath();

}

Method ini dipanggil saat memasuki keadaan Idle. Ini mengatur kembali target menjadi null dan membersihkan jalur NavMeshAgent.

Secara keseluruhan, skrip ini menunjukkan penggunaan state machine terbatas untuk mengontrol perilaku objek permainan sandera di Unity. Sandera dapat berada dalam keadaan awal, keadaan idle, atau mengikuti target berdasarkan interaksi dengan objek permainan lain

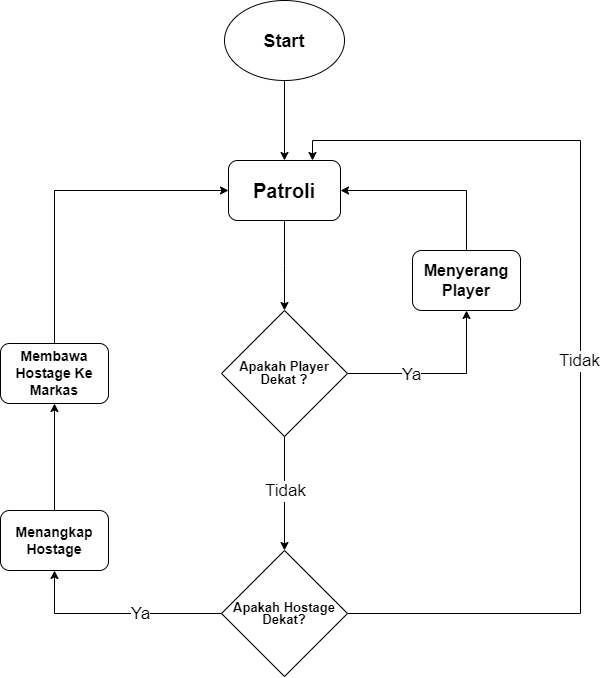
**C Behaviour Terrorist( Behaviour Tree )**



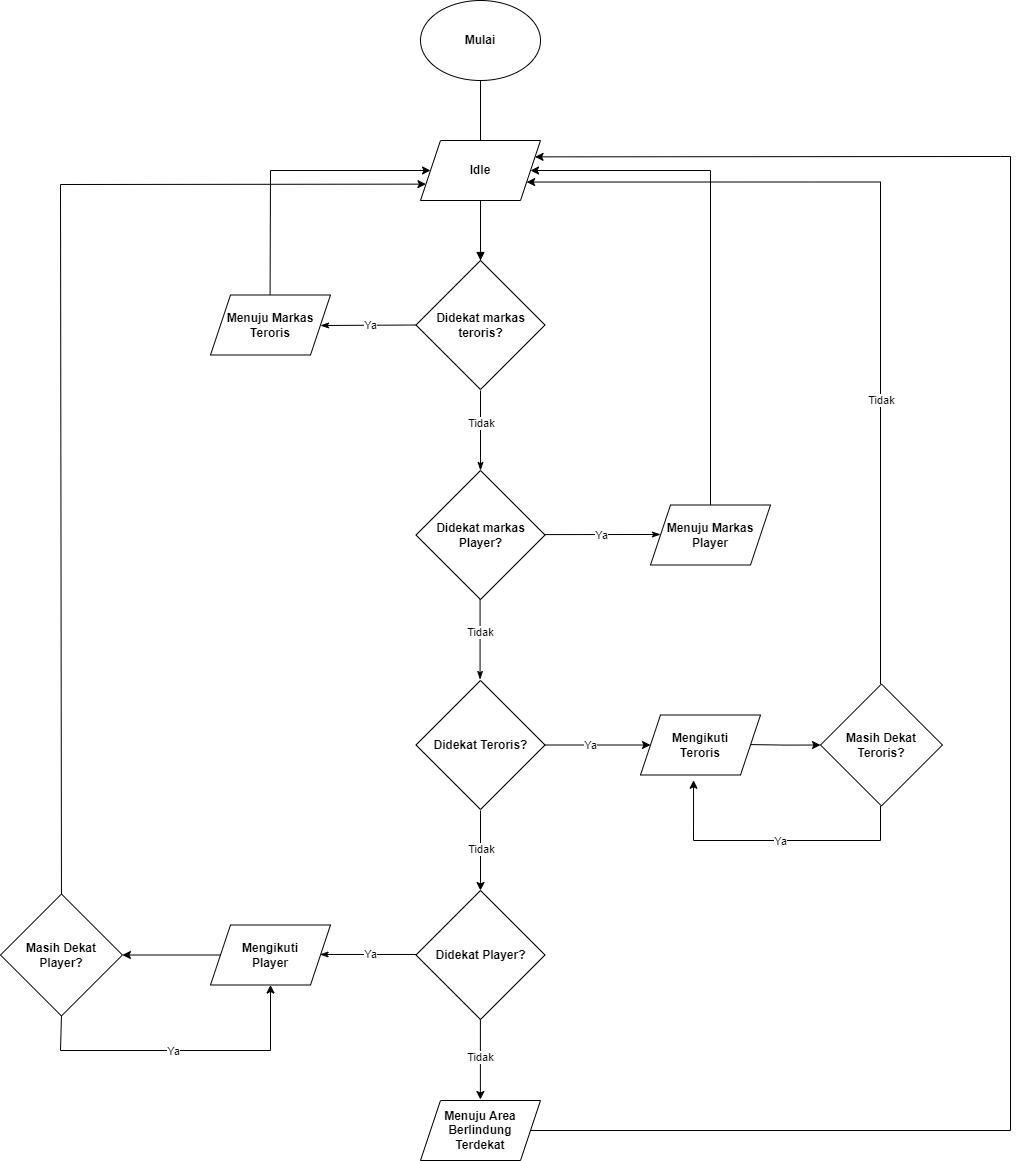
* Mereka akan menyerang pemain jika berada di dekat pemain.
* Jika berada di dekat sandera, mereka akan menculik sandera dan membawanya kembali ke markas teroris.
* Jika sudah mencapai markas teroris, mereka akan melakukan patroli.

D, FlowChart

* + 1. Flowchars Ai Terrrorist



* + 1. Flowchart Ai Polisi



**BAB III**

**PENUTUP**

**A,Kesimpulan**

Kesimpulan dari project AI yang kami kerjakan adalala kami di sini lebih menonjolkan mekani Ai yang di miliki oleh Hostage Dan Terrorist . Mungkin akan berkembang

**Daftar Pustaka**

[Apa Itu Artificial Intelligence (AI)? Ini Pengertian dan Contohnya! (niagahoster.co.id)](https://www.niagahoster.co.id/blog/artificial-intelligence-adalah/)

[Flowchart Adalah: Fungsi, Jenis, Simbol, dan Contohnya - Dicoding Blog](https://www.dicoding.com/blog/flowchart-adalah/)

[Finite State Machine : Pengertian, Karakteristik dan Cara Kerja (kantinit.com)](https://kantinit.com/kecerdasan-buatan/finite-state-machine-pengertian-karakteristik-dan-cara-kerja/)