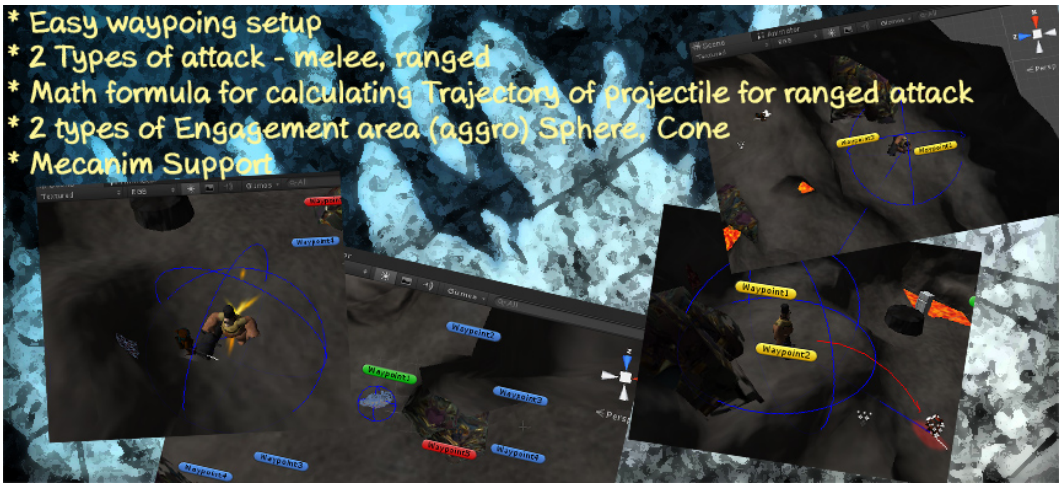
EnemyAI Package



Developed by: Anastasios Chouliaropoulos & Argiro Georgopoulou

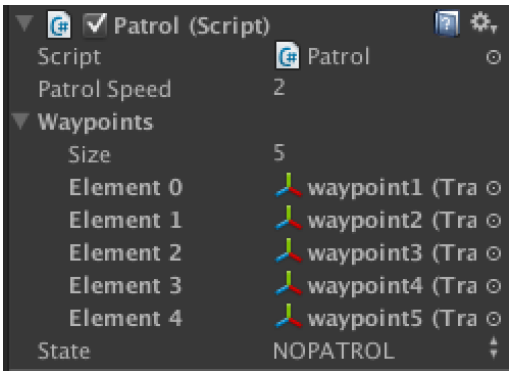
for

unity

The Enemy package consists of the Patrol, Aggro, Attack and Health scripts. They were designed in a very abstract way in order to be able to work separately - in case one of them is not needed by the **user.** They can be easily triggered as they were designed using a STATE logic. Also Mecanim animation is supported with an example script, model, animation and state machine.

Patrol

The enemy moves around predefined locations, the waypoints. Those can be placed according to the **user's** preference and then can be referenced in the inspector under **Waypoints.** Also the Patrol speed can be tweaked through the inspector.



To enable the script functionality the **user** just needs to change the state to **PATROL** or to **NOPATROL** to stop patrolling and go back to a sort of idle state.

Aggro

This script is the engagement area around the enemy. It consists of 2 levels.

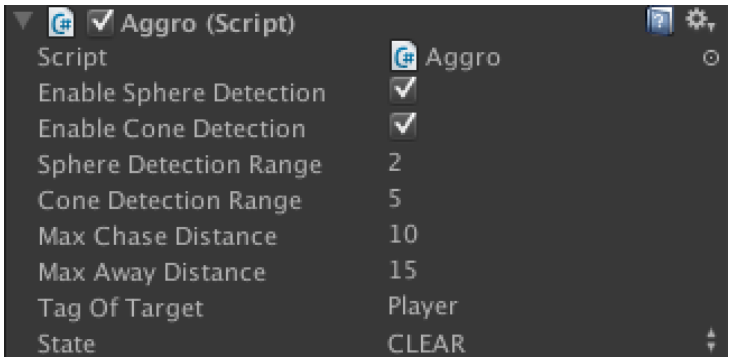
- **SPHERE :** a sphere around the enemy. When a target enters its radius, the SPHERE state is enabled and the **Target** variable is set with the target's GameObject.
- **CONE** : a cone in front of the enemy, within its field of view. When a target enters this area, the CONE state is enabled and a **Target** (GameObject) is set.
- **CLEAR** : this state is active when the **Target** is null, which means nothing triggered the other 2 states.

The **user** can enable or disable SPHERE and CONE detection for the Enemy by toggling the appropriate bool values. Also the range of the 2 states can be adjusted by tweaking **SphereDetectionRange** and **ConeDetectionRange** values.

Also there are 2 more variables that can be tweaked through the inspector.

- **Max Chase Distance** : Is the maximum distance between the Enemy and the target. If it's greater, the State changes back to CLEAR.
- **Max Away Distance** : This is the maximum distance that the enemy can walk away from its initial position. If it's greater, the State changes back to CLEAR.

Finally, the field **Tag of Target** represents the tag of the targets that the enemy Aggro/Engagement responds (e.g. "Player").



Attack

The attack script is responsible for the attack of the enemy and consists of 2 types of attack, **MELEE** and **RANGED**. This can be selected in the inspector in the **Type** field and defines the attack AI of the enemy. The attack functionality can be enabled by changing the **State** from **NOATTACK** to **ATTACK** and depending on the Type, the appropriate functionality will be triggered (melee or ranged).

MELEE

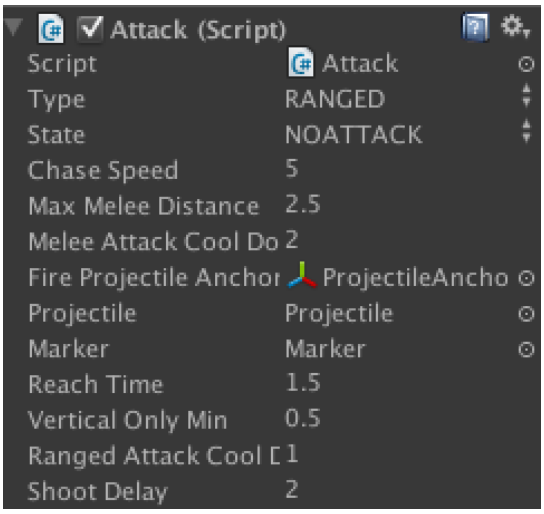
- **Chase Speed :** Enemy speed while chasing can be tweaked through the inspector.
- **Max Melee Distance** : is the max distance from the Target when the enemy starts melee attacking.
    - First it chases the target moving towards it
    - When it approaches, it checks if the target is in front of the enemy
    - and starts attacking -if the melee attack is not on cooldown (or 0)

- **Melee Attack Cooldown** : limits the melee attacks to perform every few seconds. User can tweak this value in the inspector.

RANGED

The ranged attack is similar to a cannon attack - done by computing the trajectory of projectile.

- **Fire Projectile Anchor :** The **user** can define through the inspector the anchor of the projectile, which is the position of the enemy model where the bullet is fired from.
- **Reach Time :** Defines after how much time the bullet will land on the target which also affects the angle of the trajectory.
- **Shoot Delay :** The enemy first “marks” the target and shoots the bullet on that spot after **“Shoot Delay”** seconds. Should be set to 0, if the ranged attack should have no delay after marking the target.
- **Ranged Attack Cooldown** : **User** can tweak this value to adjust how fast every ranged attack performs.
- **Projectile :** is the bullet prefab e.g. cannon ball
- **Marker :** an indication regarding to where the projectile will land.

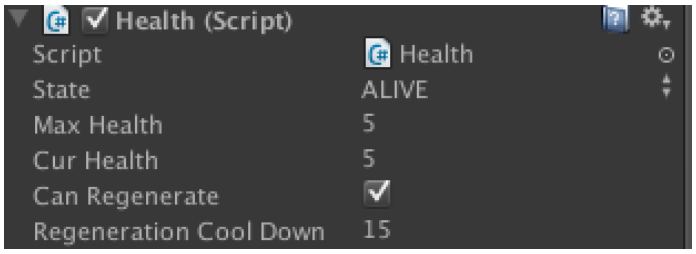


Health

The **State** can be called to check if the enemy is **DEAD** or **ALIVE**.

The user can tweak values through the inspector:

- **Max Health :** the maximum health that the enemy can have.
- **Cur Health :** is the current health of the enemy during the game
- **Can Regenerate :** toggles the ability for the enemy to regenerate health and recover back at **Max Health** when its not receiving any hits.
- **Regeneration Cooldown :** is the time after the last attack to the Enemy that can trigger the Health Regeneration.



Enemy Animation

Controls the enemy animations and rotations. Actual movement is handled by the NavMeshAgent.

The animations are for demonstration only -taken from the free unity tutorials. That's why they don't look quite right. This was added, so you can have a reference how to implement mecanim animations in your enemy.

EnemyController

It's an example that shows how the **User** could trigger the different scripts that were explained above and create a sort of AI using the different states.