

Emerald AI 3.1

Documentation



Online Documentation

Emerald AI's documentation is too large to fit into a PDF so it has move to the [Emerald AI Wiki](#). This wiki has tons of tutorials, guides, and API to help you get the most out of Emerald AI. Below you can find a basic guide on getting starting, **but it is highly recommended users use the Emerald AI Wiki.**

Table of Contents

Example Scenes

[Passive AI Example](#)

[Territorial AI Example](#)

[Companion AI Example](#)

[Aggressive AI Example](#)

[Pet AI Example](#)

[Line of Sight Example](#)

[Combat Text & Health bar Examples](#)

[Setting Up Example](#)

[Fleeing AI Example](#)

Setting Up your Own AI

Setting Up Waypoints

Setting Up an AI Destination

Setting Up an AI's Line of Sight

Setting Up an AI's Alert Mode

Setting Up an AI's UI

Setting Up Ranged Combat

Setting Up an AI's Optimization Settings

Setting Up an AI's Animations

[Applying and Updating Animations Through the Emerald Editor](#)

Setting Up an AI to be used with Object Pooling

Understanding an AI's Temperament

Behavior Types

[Passive](#)

[Cautious](#)

[Companion](#)

[Aggressive](#)

[Pet](#)

Confidence Level

Adding Footstep Sounds Timed to Custom Animations

Understanding an AI's Target Picking Method

[First Detected](#)

[Closest](#)

Example Scenes

Emerald AI has several example scenes to help get you started with using Emerald. It is recommended that you test these scenes prior to creating your own AI so you can get a grasp of the Editor and how it works. While testing the example scenes, feel free to tweak the settings.

Passive AI Example

Demonstrates an AI that roams around passively. A passive AI will only react to its target if it is damaged. Its actions are based on its Confidence Level. If an AI is set to Coward, it will flee. If an AI is set to Brave it will fight until its health is low. If an AI is set to Foolhardy, it will continue to fight until death.

Territorial AI Example

Demonstrates an AI that is territorial. A Cautious AI will watch a target before it attacks. It will play a warning animation before it is about to attack. If the target does not leave its radius soon, the AI will initialize an attack. If the target exits the AI's radius, it will resume wander.

Companion AI Example

Demonstrates an AI that acts as a companion. Companion AI will follow around an object with the appropriate tag, in this case, the Player tag. The companion AI will fight for the player and resume following the player when it is finished with combat.

Aggressive AI Example

Demonstrates an AI that acts as hostile towards any target with the appropriate tag.

Pet AI Example

Demonstrates an AI that acts as a non-combat pet. Pet AI cannot attack or be apart of combat. They are purely for cosmetic purposes.

Line of Sight Example

Demonstrates an AI that uses Line of Sight to detect the player. The angle for this is about 200°.

Combat Text & Health bar Examples

Demonstrates the Combat Text and Health Bar features on AI.

Setting Up Example

An example scene that's intended to be used for follow along with the Setting up your Own AI guide below.

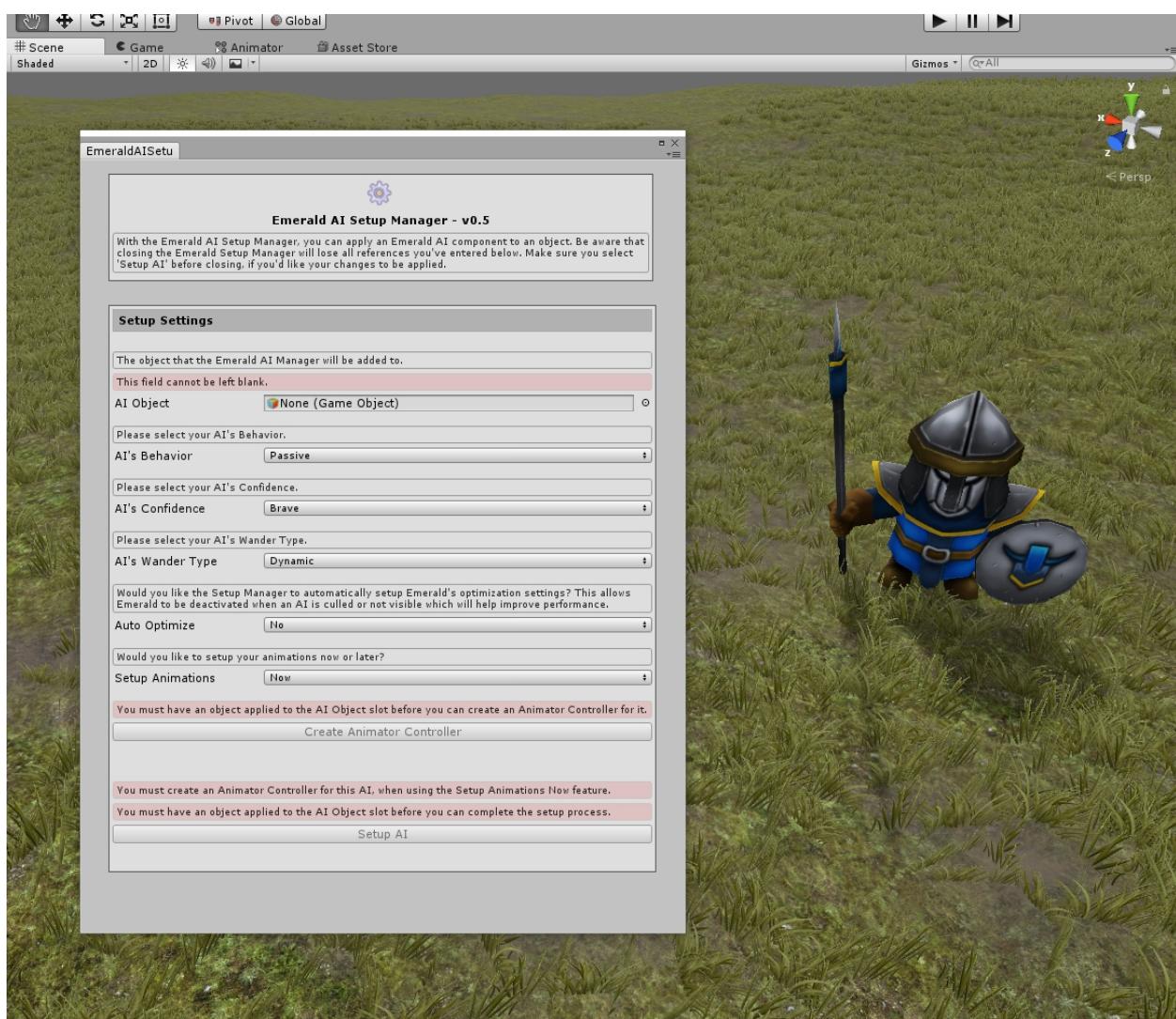
Fleeing AI Example

An example scene that demonstrates an AI fleeing from the player.

Setting Up your Own AI

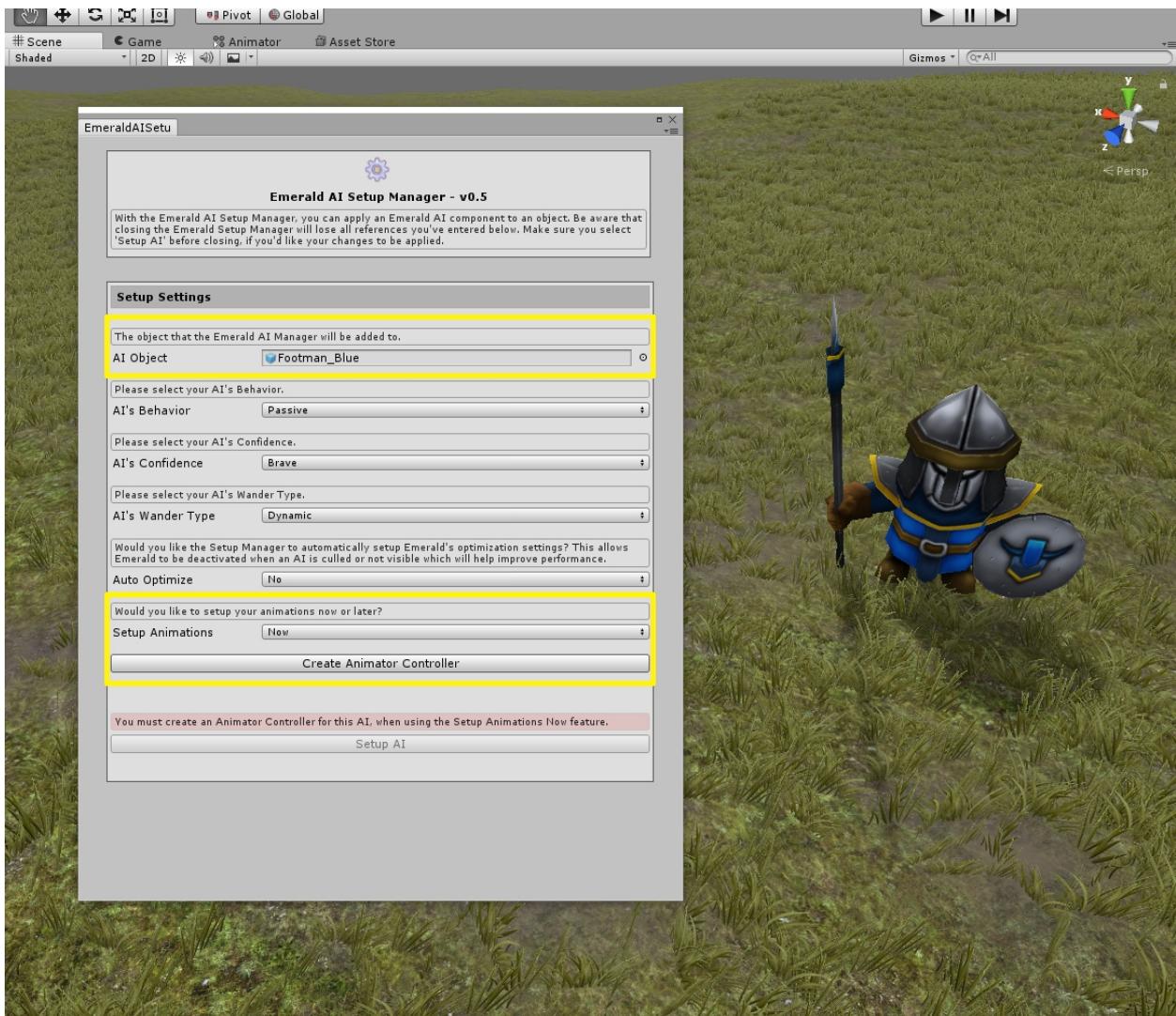
When you are ready to create your own AI, you will need to open up the Emerald AI Setup Manager. This can be done by pressing Ctrl+Shift+E or by going to Window>Emerald AI>Emerald AI Setup Manager. If you'd like to follow along with the tutorial, you can do so by using the Setting Up scene included with Emerald.

Once you've done this, you will see the Setup Manager like below. Ensure that you have an object that you'd like to set up as an AI ready. Note: When an AI is created, it will break the prefab instance.



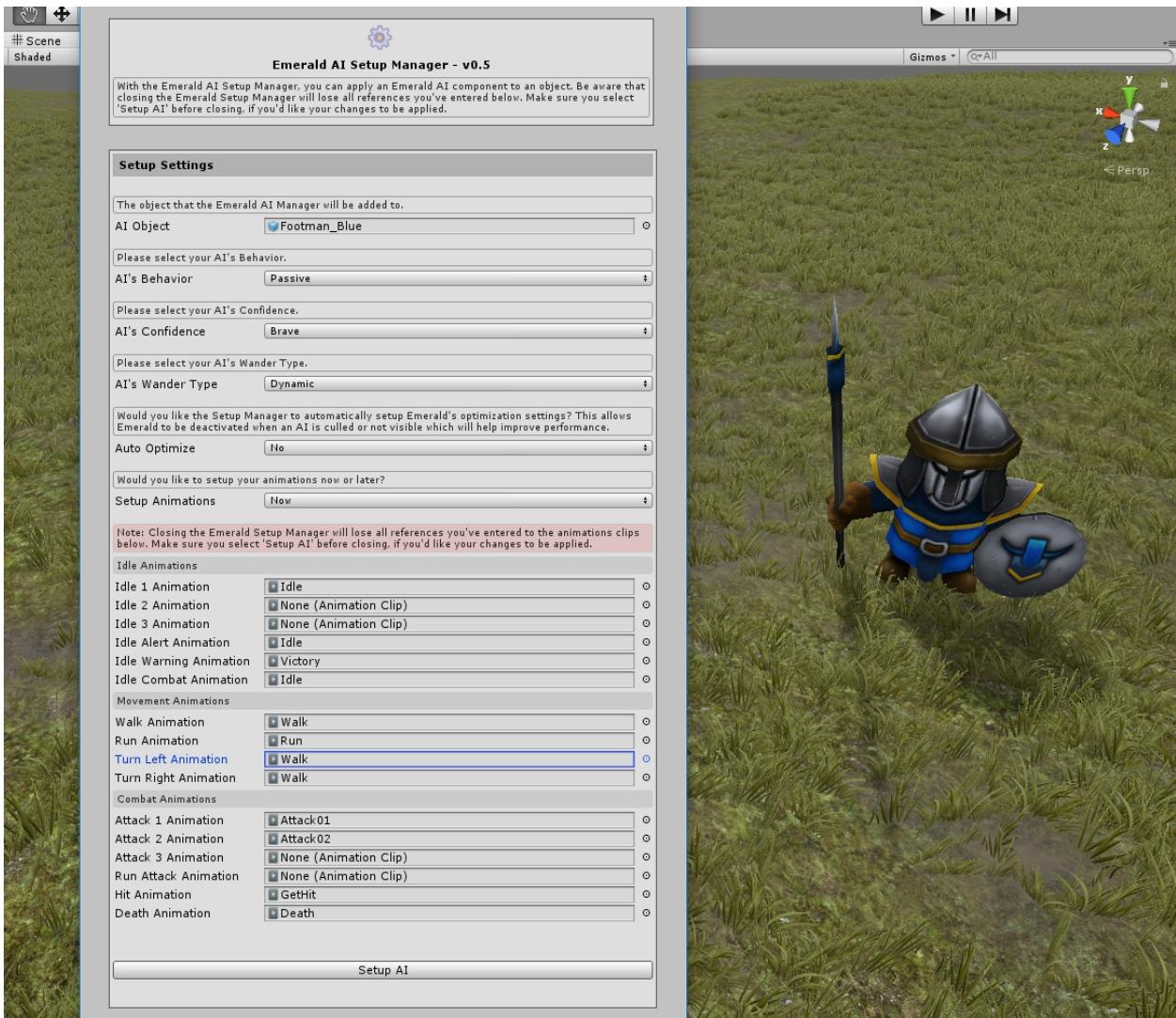
The Setup Manager allows you to easily create an AI. The Setup Manager will apply all needed components and apply an Emerald AI system to the object in the AI Object slot. You can change preset options and apply animations which Emerald will then apply to the created AI.

In order for the Setup Manager to allow you to create an AI, you will need to have an object applied to the AI Object slot. Once you've done this, the "Setup AI" button will become available. Setting up an AI's animation first with the Setup manager is highly recommended before creating your AI. This can be done by pressing the "Create Animator Controller" button. Once you've done this, a save menu will open. Here you will need to pick a location and a name for your Animator Controller to be saved to.

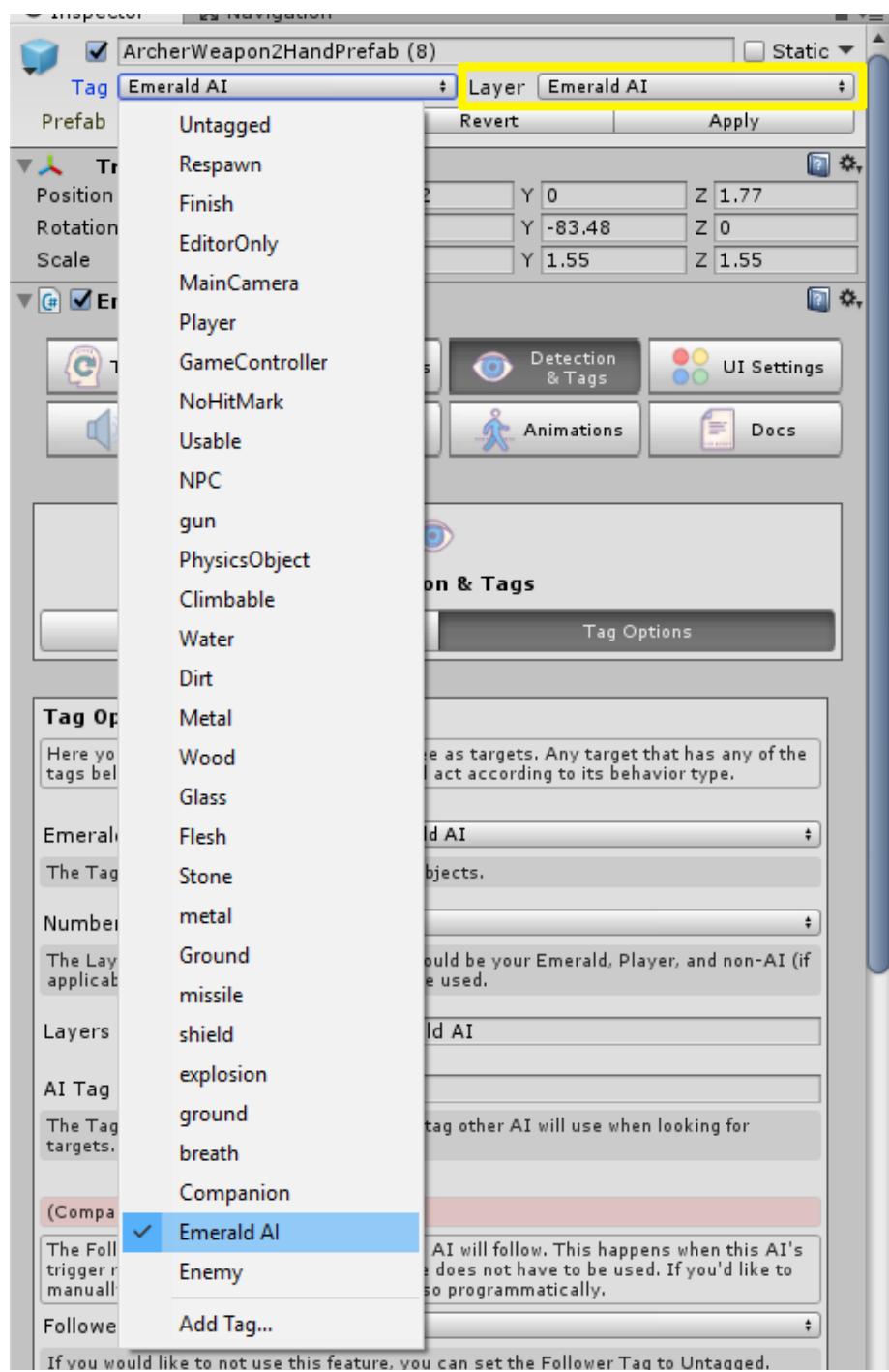


Once you've saved an Animator Controller, a new one will be created for your AI. This will also open up additional options that will allow you to set animations for your AI. It is recommended that an animation is filled for every slot. Duplicated animations can be used, if desired. If you don't have a turn animation, a walk animation will suffice. Emerald will automatically adjust your AI's animation settings, if you only have 1 idle and 1 attack animation.

When you are finished with applying your animations, press the "Setup AI" button to finish the setup process. Everything you've entered in the Setup Manager will be applied to the AI you just created.

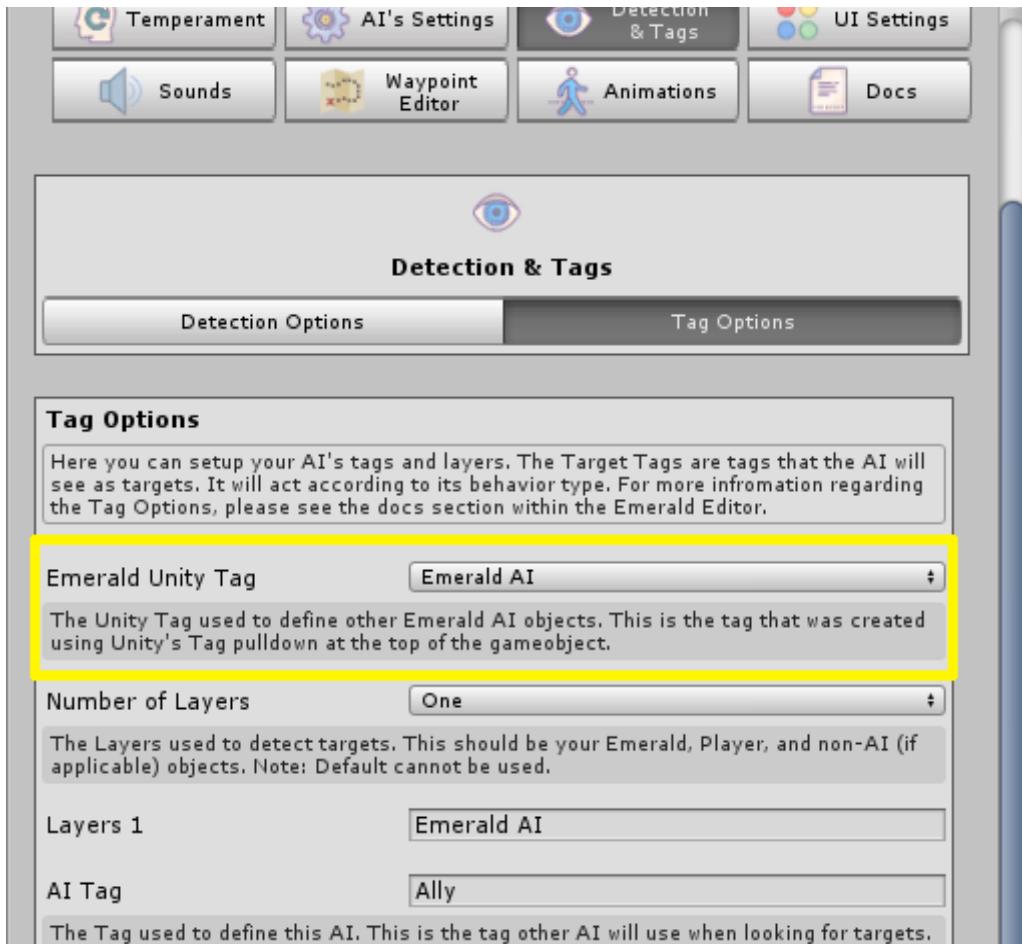


Before you can test your newly created AI, you must setup your AI's Unity Tag and Layer. This is done by selecting your AI object and going selecting the Tag and Layer pull down menus. An AI's tag is used to help other AI identify what is an appropriate target and what isn't. This will vary depending on your project's needs, but for this guide, we will use the Emerald AI tag and Emerald AI layer. You will have to create this yourself by going to the Add Tag and Add Layer options at the bottom of the pulldown menus.

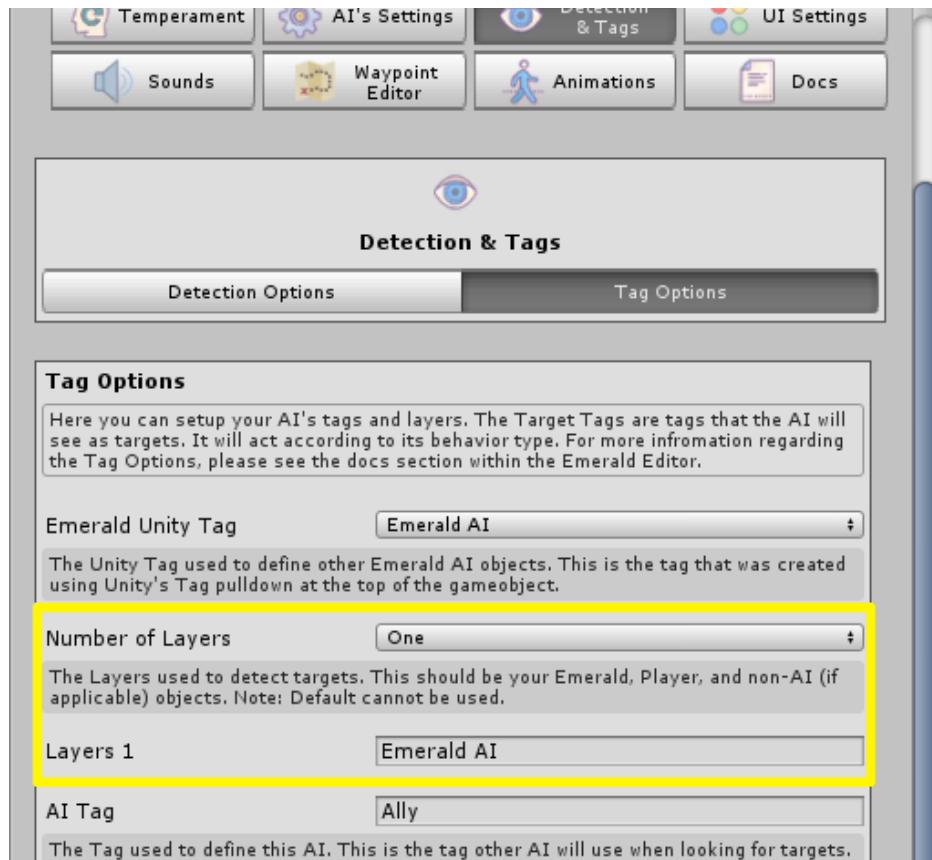


This next part will be covered in 4 steps because it's important and can be a little tricky. You will need to tell the Emerald Editor the Unity Tag and Unity Layer you set above. This is done to help increase performance as it allows AI to only look for objects with the appropriate Unity Tag and Layer.

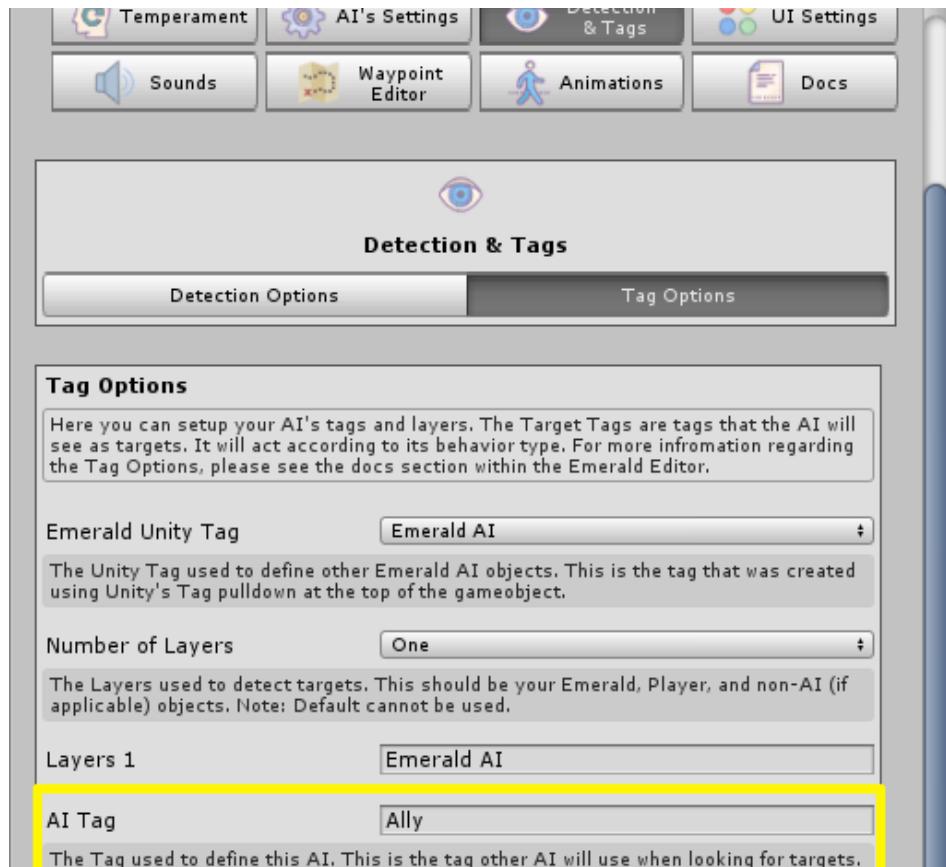
Step 1) Go to the Detection & Tags and select the Tag Options section. Next, set your Emerald Unity Tag to the tag you set above, which in this case is Emerald AI.



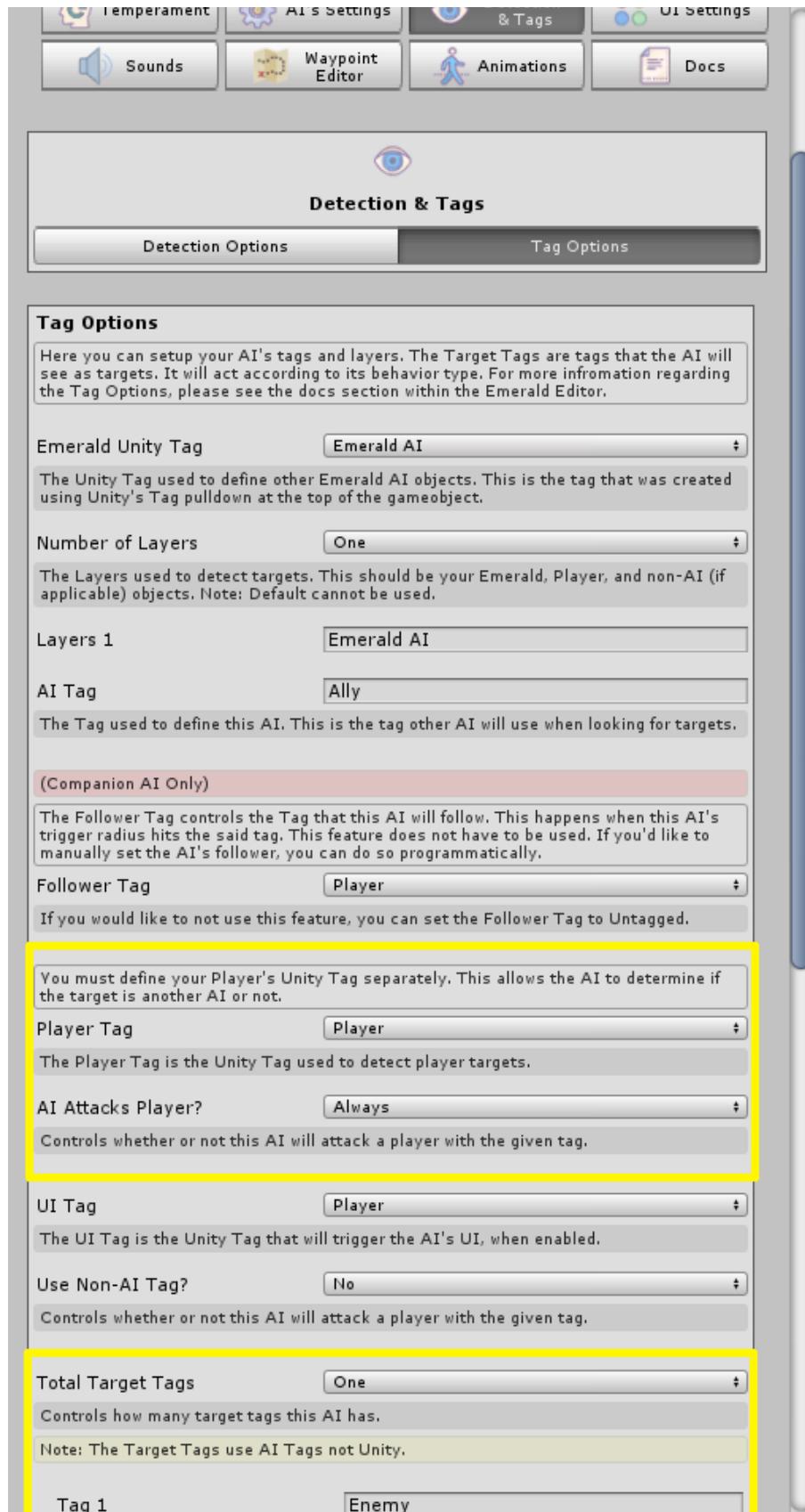
Step 2) In the same section, set the number of layers you will be using. If your player uses a different layer, you will need to set its layer here. With this tutorial, we will be setting the Emerald AI layer we set above here.



Step 3) Now, you will need to set your AI Tag. This is the tag that defines this AI and the tag that other AI will use when looking for targets. This can be anything you'd like something more vague like Prey or Ally; or something more specific such as Rabbit or Human.



Step 4) Lastly, you will need to define your AI's target tags. These should match up with other AI's AI Tags you've set. You can also set your Player tag and whether or not an AI will attack a player. For this guide, this AI will be attack our player. Set the Player Tag to Player and set the AI Attacks Player to Always. Remember, the tags used can be anything desired. The Emerald AI and other tags are used to help get you started.



Now, try testing the scene out. If you've setup everything correctly, your AI should react to your player and other objects of the appropriate tag. If for some reason your AI is not working properly,

repeat the Setting Up your Own AI steps to ensure none were missed.



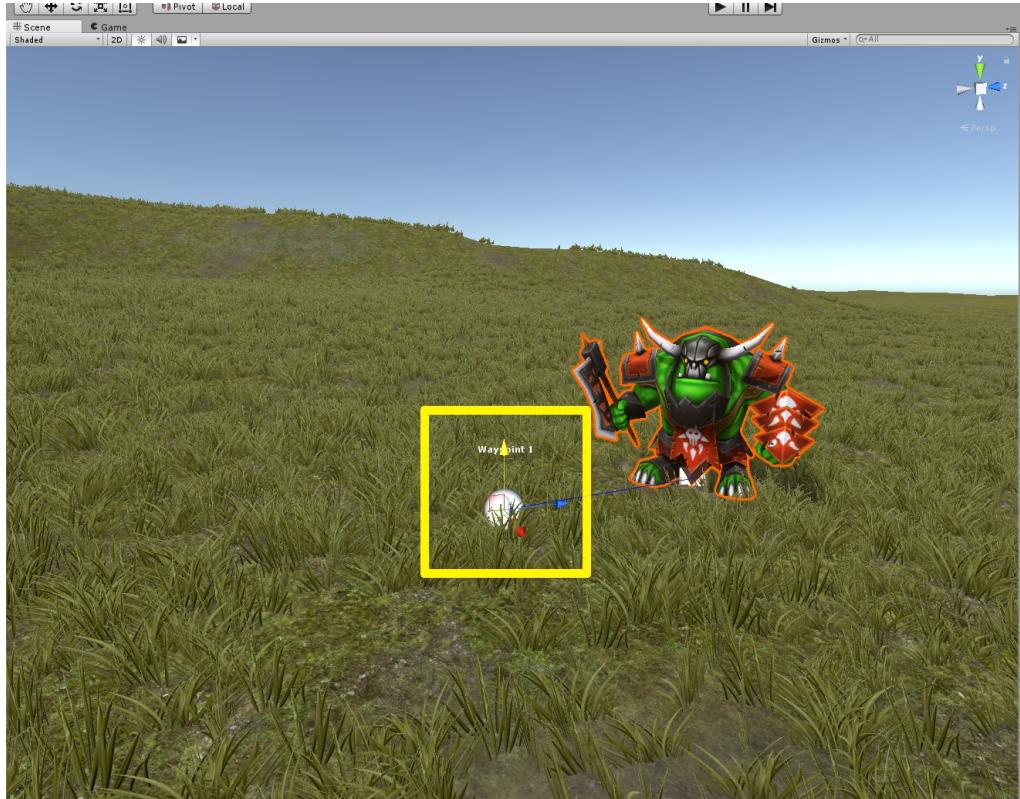
Setting Up Waypoints

Waypoints are a powerful new addition to Emerald AI. They allow you to create a series of destinations for your AI to move between. AI that are using waypoints will still react to targets according to their Behavior Type. Once a target has been killed, or successfully fled, the AI will resume moving between its waypoints.

If you'd like to create your own waypoint system for your AI, you can do so by going to the Waypoint Editor tab located in the Emerald Editor. By default, the Wander Type is set to Dynamic (which randomly generates waypoints). When you open the Waypoint Editor tab, you will see a button to enable waypoints. Once you've done this, you will see the waypoint options appear. You can now create a waypoint by pressing the "Add Waypoint" button.

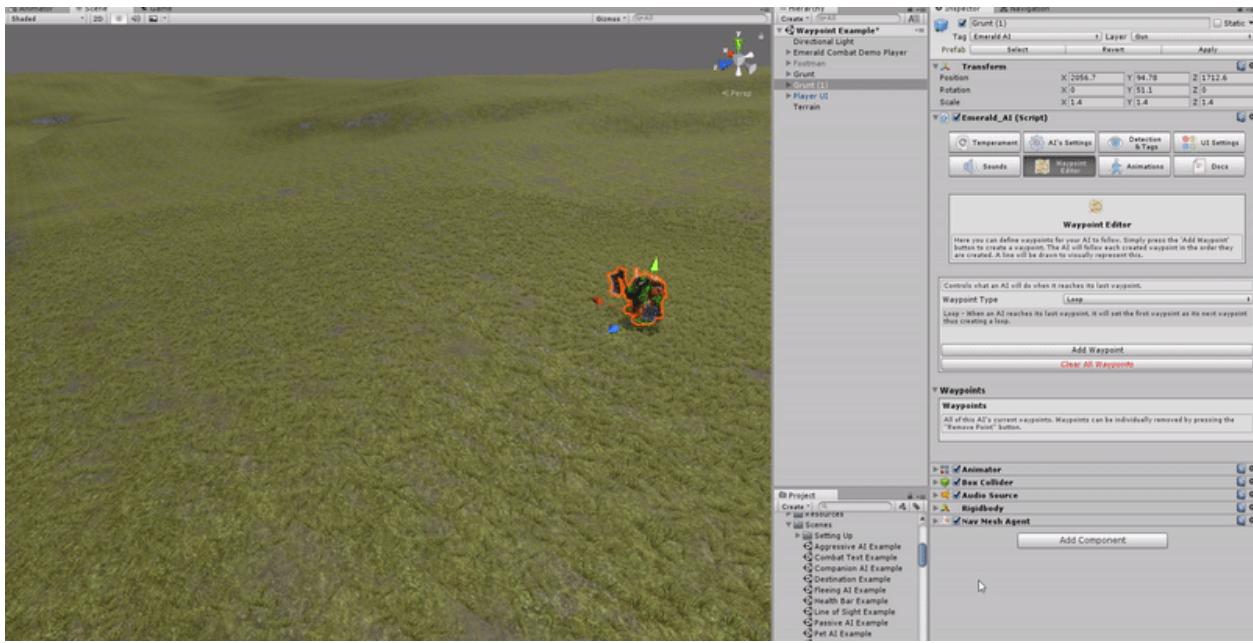


Pressing the Add Waypoint button will create your AI's first waypoint. You must have at least 2 waypoints for your AI to move between. Waypoints are only visible while the Waypoint Editor tab is active for each AI. Multi-Object Editing is not available for the Waypoints tab.

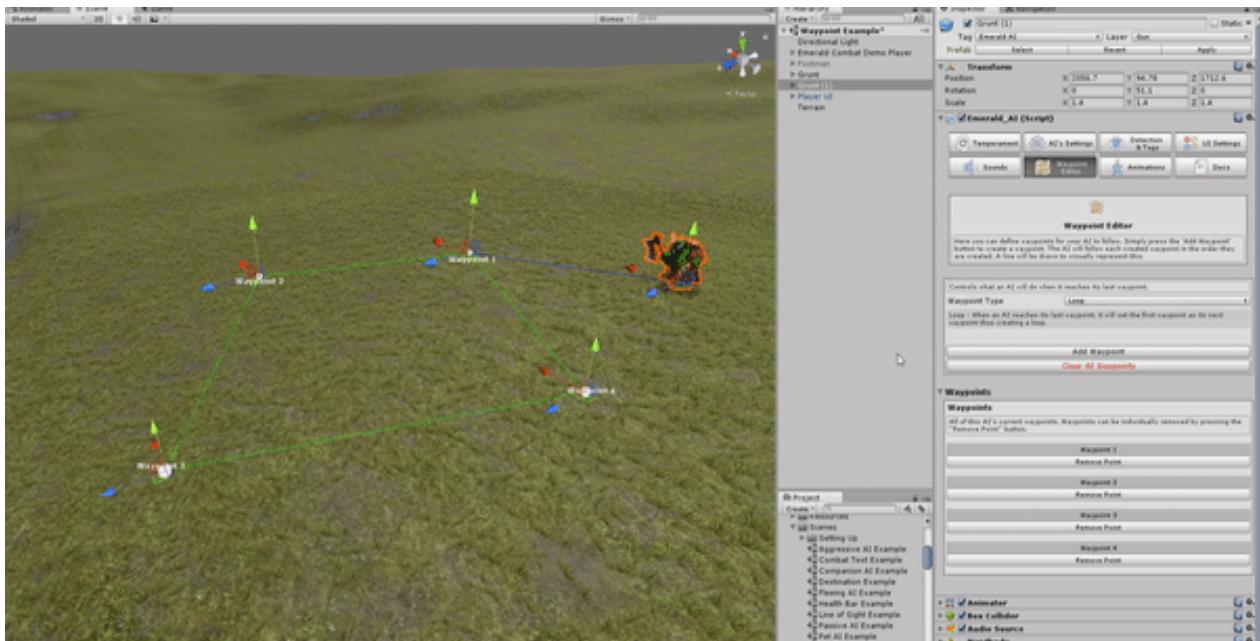


Each time that you press the Add Waypoint button, a new waypoint will be created. Each newly waypoint will be positioned 1 unit on the Z axis to avoid being placed in the same position as the previous waypoint. This also makes it easier for customizing your AI's route. The blue line you see that goes from your AI to your first waypoint indicates the AI's direction to the first point. You have the option to loop your waypoints, which will allow your AI to continue to loop through the waypoints after it has reached the end, or Reverse, which will have your AI walk the same path but in the opposite direction.

There is no cap on waypoints that can be created. When you test your scene, your AI will immediately start following its waypoint, given that it is currently in an active state.



If you happen make a mistake, and need to remove a waypoint that's located in the middle of 15 points, you can remove that specific waypoint and Emerald will resize itself so you don't have to start over. If you'd like to completely start over, you also have the option to clear all your waypoints. This can done by pressing the "Clear All Waypoints" button. Keep in mind that both of these processes cannot be undone.



Setting Up an AI Destination

Destinations are different than waypoints. Destinations allow you to set a point anywhere on the NavMesh and the AI will move to that destination. AI using destinations will not follow any other waypoints and will take the quickest route to get to their set destination. AI that have arrived at their destination will have a public variable called “AI Reached Destination” set to true. This can be useful for triggering other events. At anytime, you can change the AI’s destination programmatically by calling the `SetDestination(Transform Destination)` function. This feature can even make it possible for AI to even have schedules through code or a time of day system such as [UniStorm](#) (Emerald AI schedule integration with UniStorm is coming soon. This will allow users to set schedules via the Emerald Editor and not just through code).

Unlike Waypoints, destinations are set through the Temperament tab. To create a destination for your AI, go to the Temperament tab. Once here, change the Wander Type to Destination. This will automatically create a destination point for your AI.

Temperament Options

The Temperament Options allow you to control an AI's behavior, how it reacts to certain situations, and how it chooses to wander or move to a destination.

Behavior

The Behavior Type allows you to define the behavior your AI will have. All AI will only react to targets that have targets' tags set within the AI's Tags List.

Behavior Type: Aggressive

Aggressive - Aggressive AI will attack any target that comes within their trigger radius.

Confidence

The Confidence Level gives you more control over how your AI reacts with their Behavior Type.

Confidence Level: Foolhardy

Foolhardy - Foolhardy Aggressive AI will fight any target on sight or detection and will never flee. They will continue to fight until dead or the target has escaped the AI.

(Brave Confidence Level Only)

Health Level to Flee: 10

Controls the percent of health loss that's needed to trigger a flee attempt.

Wander Type

Controls the type of wandering technique this AI will use. AI will react to targets according to their Behavior Type.

Wander Type: Destination

Destination - Allows an AI to travel to a single destination. Once it reaches its destination, it will stay stationary.

Combat Style

(Companion AI Only)

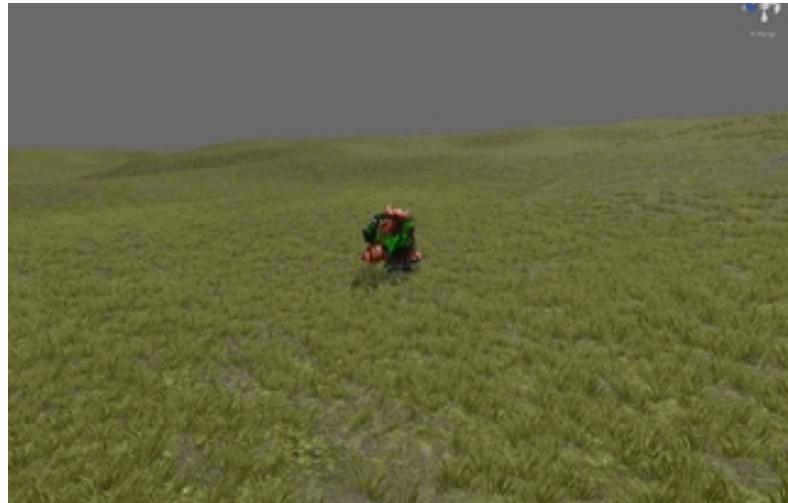
Controls whether a Companion AI will fight Offensively or Defensively.

Combat Style: Offensive

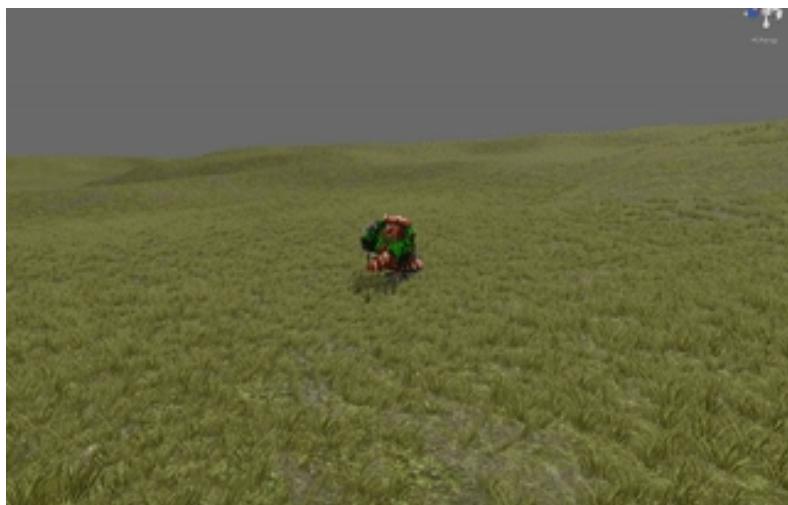
Offensive - Offensive Companion AI will attack any target that is within their trigger radius.

This will also open up an additional option called Destination Animation. This gives you control over which animation is used for moving to the AI's destination point. Walk setting will use the walk animation and walk speed you've set via the AI's Setting options and the Run setting will use the run animation and the run speed you've set via the AI's Setting options. AI will still react to targets according to their Behavior Type. Once they have successfully fled or killed their target, the AI will resume moving to their destination point.

Walk Destination Animation Setting



Run Destination Animation Setting



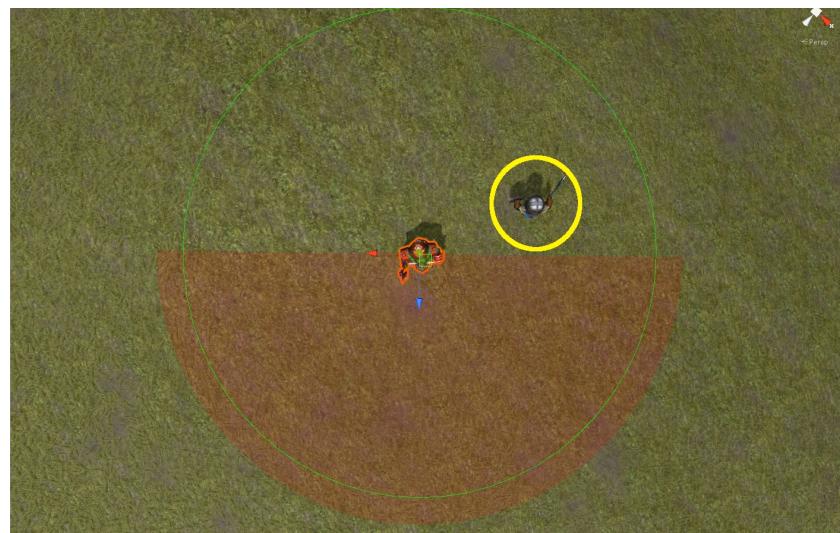
Setting Up an AI's Line of Sight

The Line of Sight feature makes it possible for AI to “see” allowing them to only target objects that they can detect with their Field of View. This works by casting a raycast according to an AI’s Field of View angle. Any object that’s greater than an AI’s Line of Sight will not be visible and anything that’s within it is visible to the AI.

Visible Target



Not Visible Target



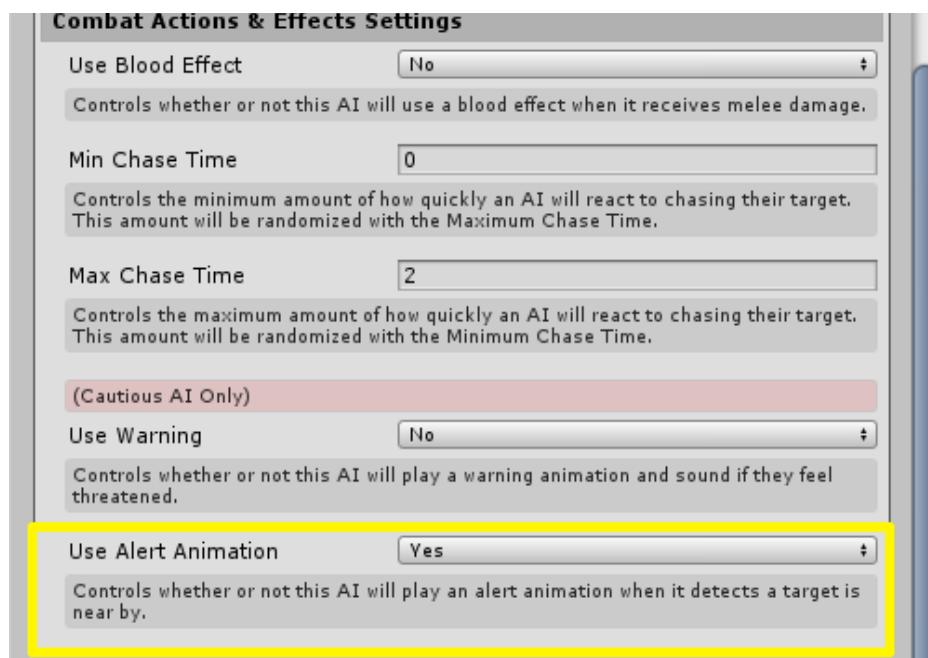
When an AI sees a target within their Field of View, and they have the appropriate tag, they will automatically set the target as their current target. They will react according to their Behavior Type.

Setting Up an AI's Alert Mode

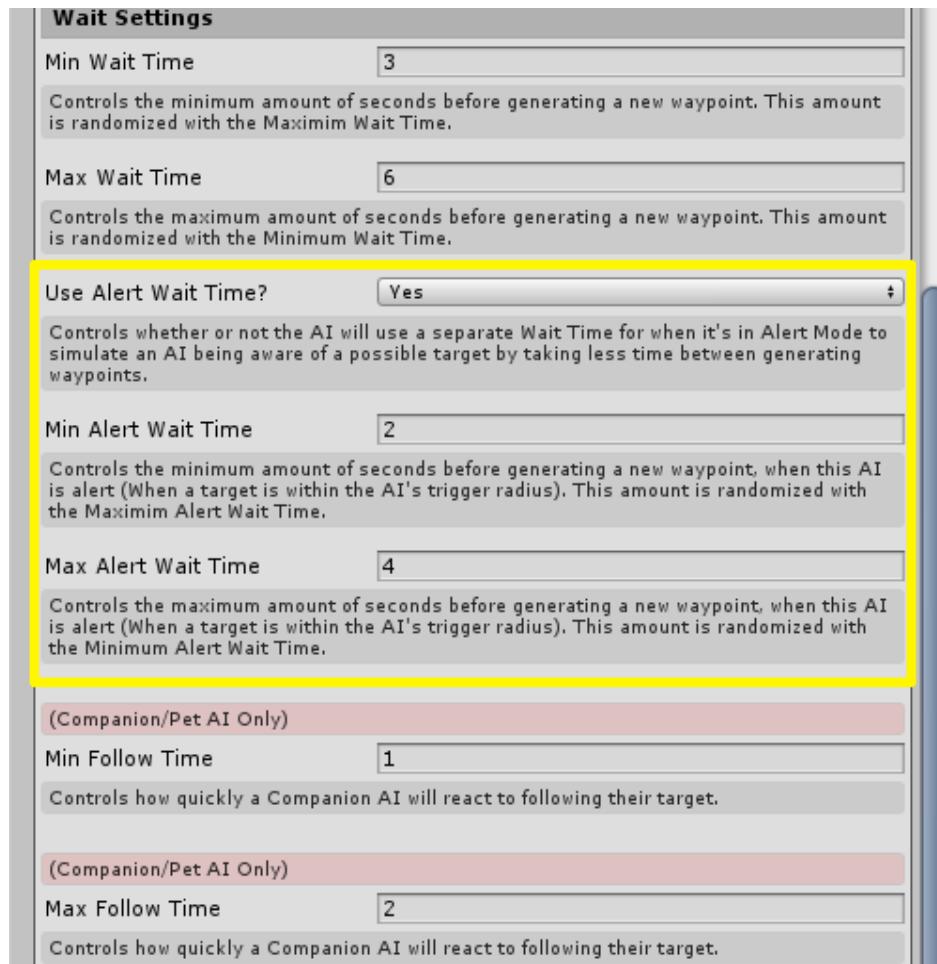
When a potential target is within an AI's Trigger Radius, they have the option to play an Alert Animation, even though they haven't seen the target yet. This is called Alert Mode and is meant to

simulate that an AI is aware/alert that there may be a target near by. While in Alert Mode, an AI also has the option to expand its Trigger Radius as well as its Wait Seconds. This is also meant to simulate that an AI is aware/alert and can make it harder for players/targets to go undetected. However, these features are completely optional.

Use Alert Animations can be found under AI' Settings tab under the Combat section. Setting it to true will give your AI the ability to use this feature. The Alert animation can be set within the Animations tab under the Idle section.

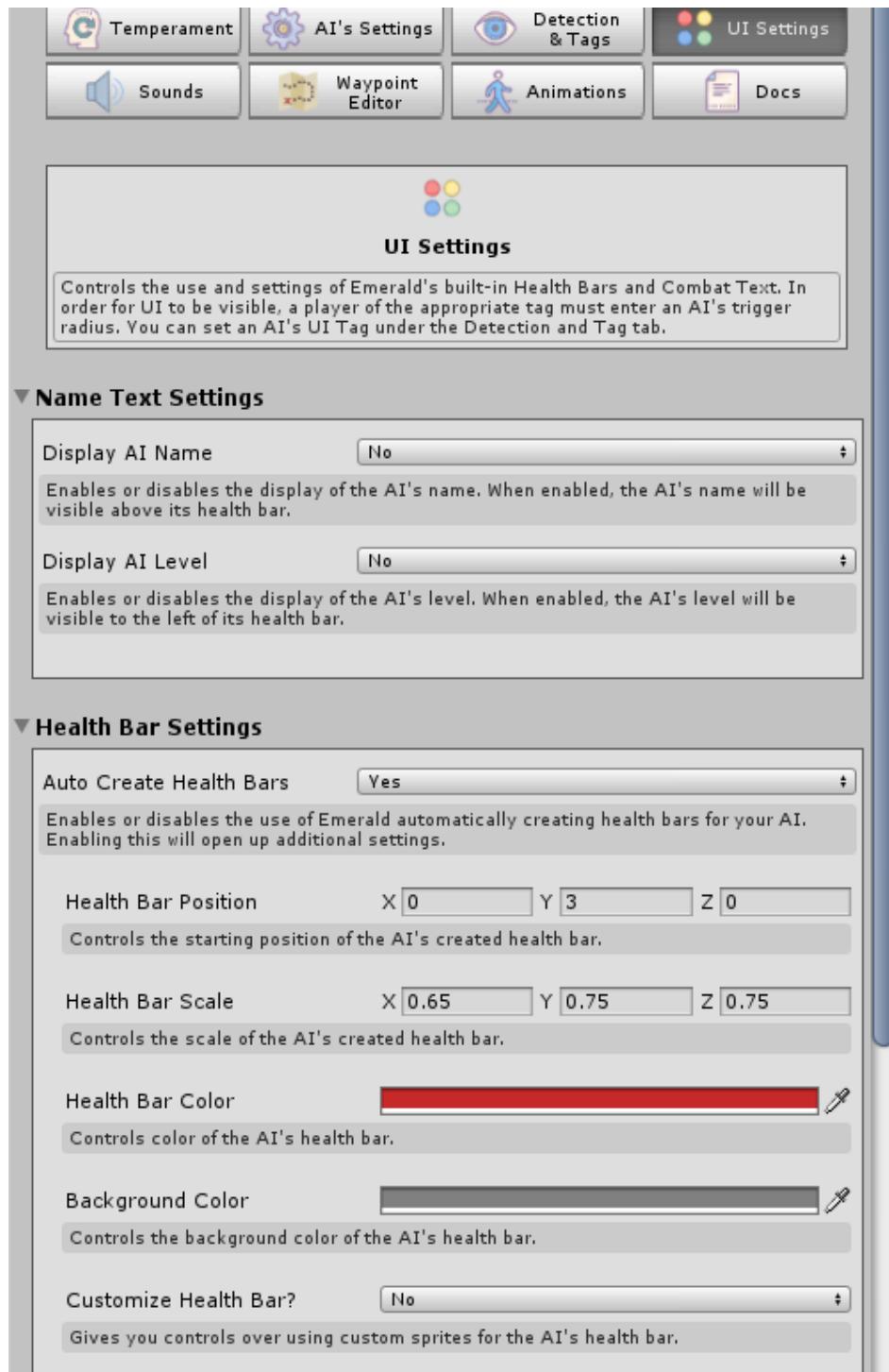


The Use Alert Wait Time can be found AI' Settings tab under the Movement section.



Setting Up an AI's UI

Emerald AI features a built-in customizable UI system capable of automatically creating combat text, health bars, and displaying an AI's name and level. To enable an AI's UI, go to the UI Settings tab. Here you will see various options to enable and customize certain features.



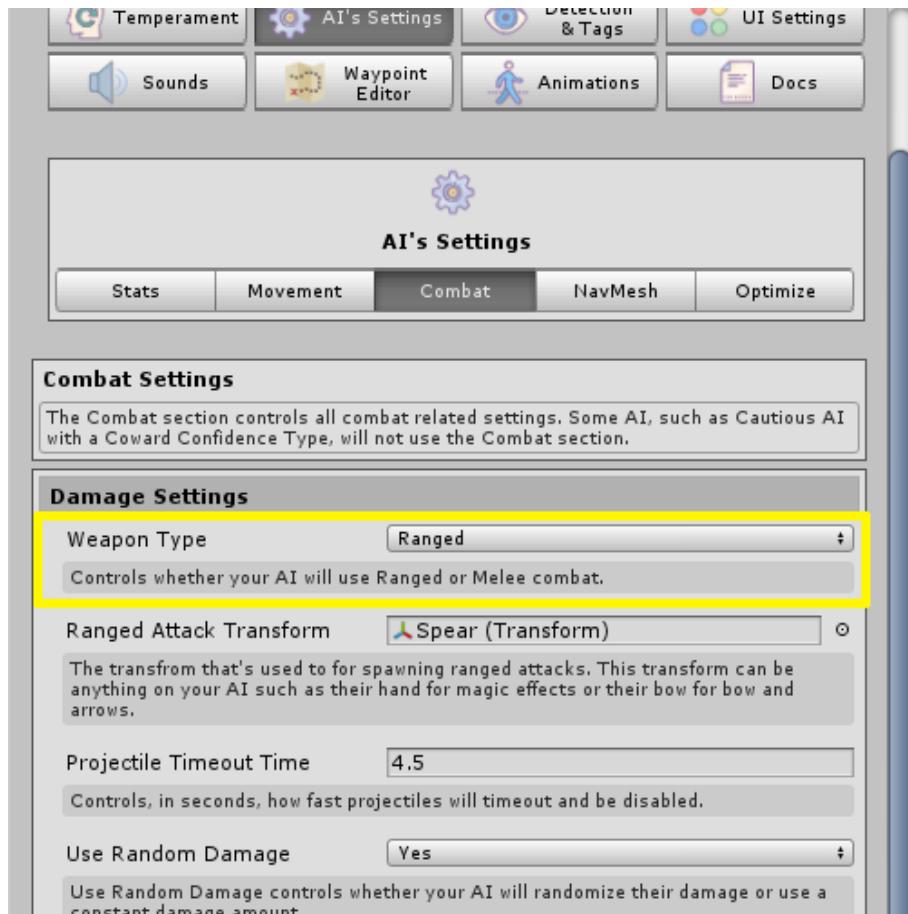
If you have the Health Bar and/or the Combat Text features enabled, and you have your AI selected, your model will have lines indicating the position and color of the UI. In this example, the red line is the health bar and the yellow line is the combat text. This makes it easier for users to place their UI on their models without having to guess an optimal position.



Setting Up Ranged Combat

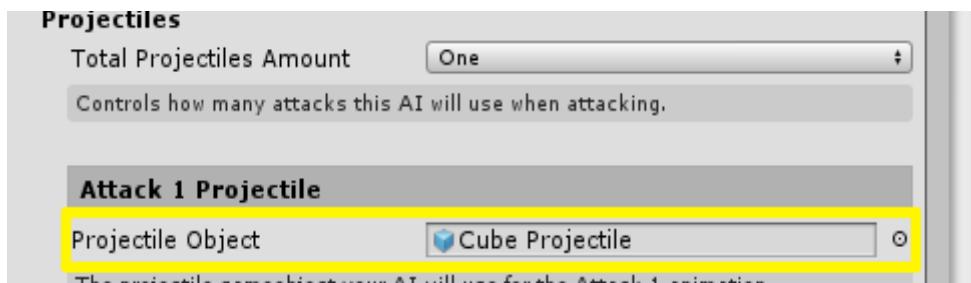
Emerald's ranged combat system allows any AI the ability to use ranged combat all without having to program anything. Everything is handled within the editor. All users have to do is customize the projectile and its settings. In this guide, we'll be using 1 projectile. To add more, simply repeat the steps for each additional projectile. Also, if you haven't already done so, ensure that you have properly setup your AI's tags or an AI will not properly be able to engage in combat.

Step 1) Go to the AI's Settings tab, go to the Combat section, and set the AI's Weapon Type to Ranged.

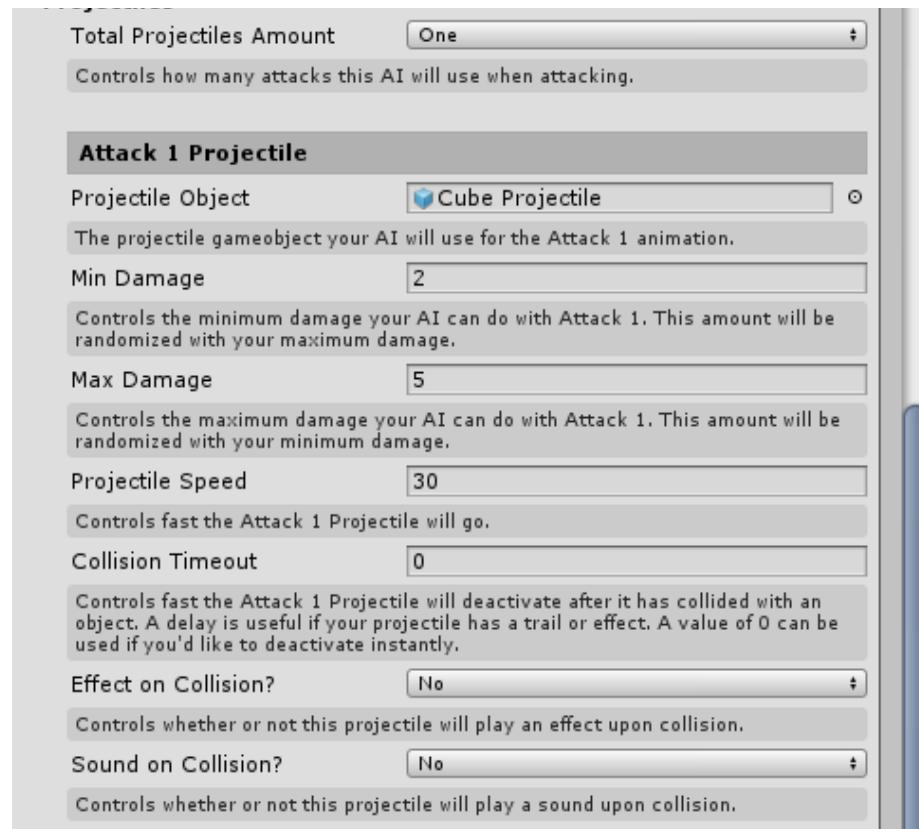


Step 2) Next, you will need to assign the AI's Ranged Attack Transform. This is the transform where projectiles will be created and given speed to move towards a target. This can be anything such as an AI's hand for magic effects or an AI's bow for arrows. (This can be seen in the above picture)

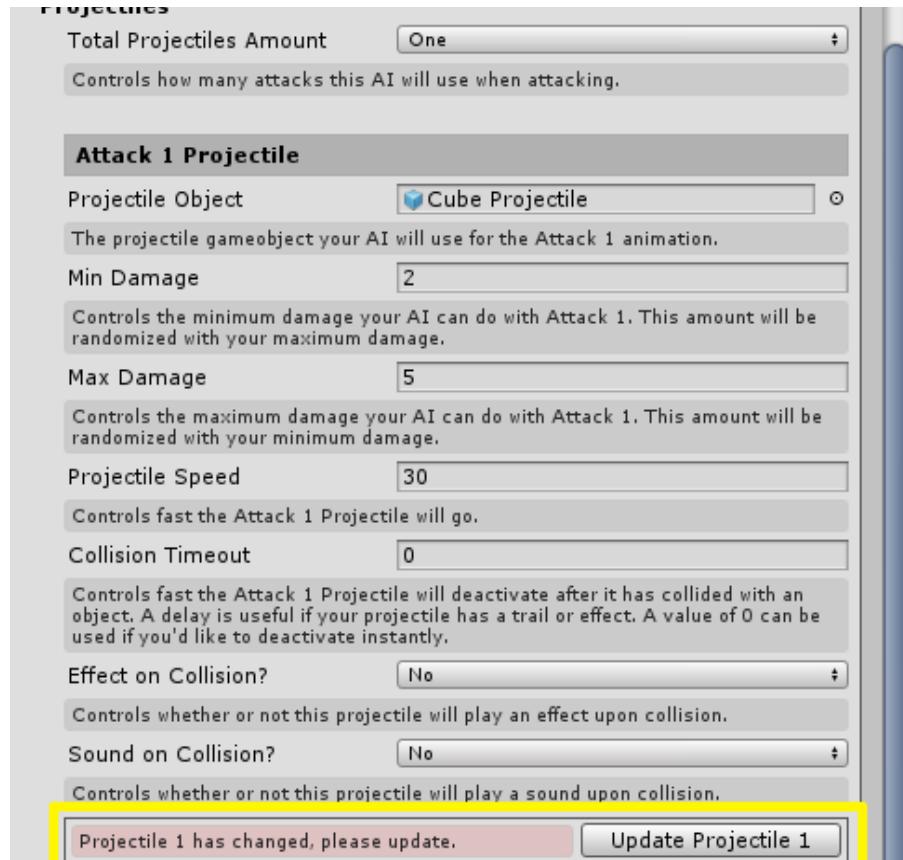
Step 3) Now, you will need to assign a projectile for your AI to shoot. This can be any object desired. However, it is important that there are no other colliders on the projectile object. Emerald will apply all needed scripts and components in a later step.



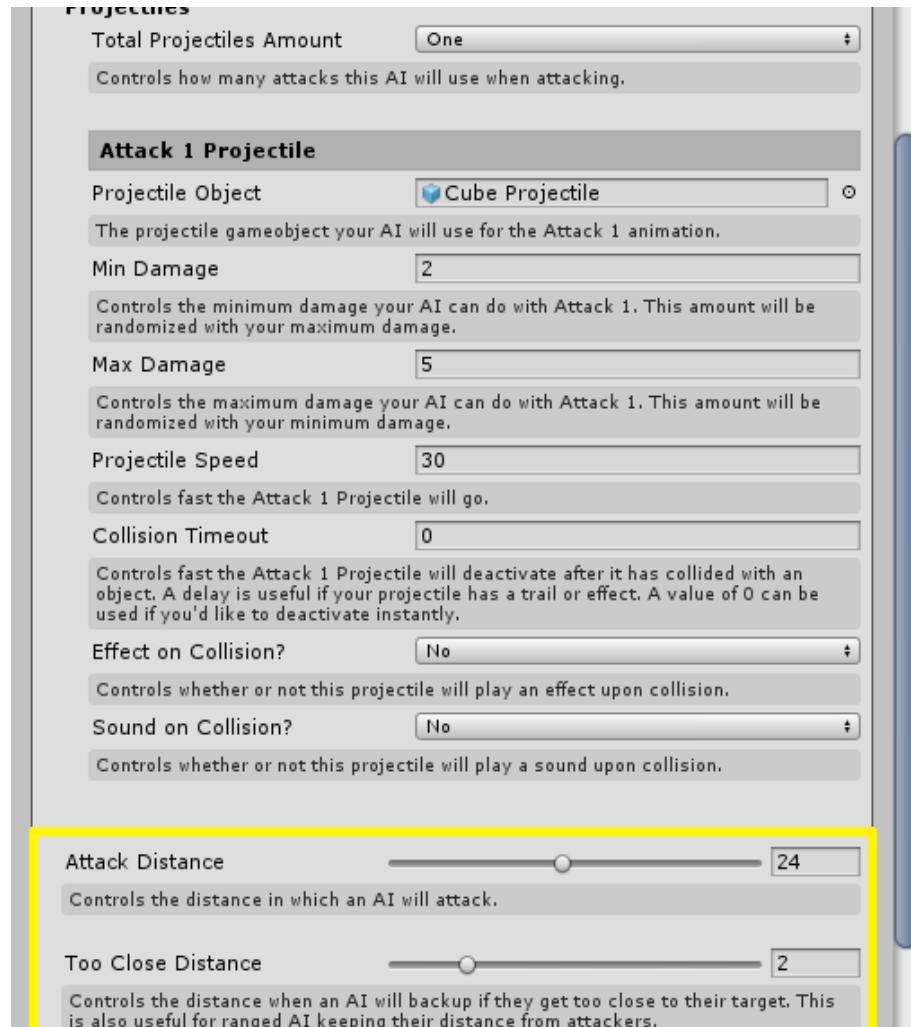
Step 4) With this step, you get to customize your projectile. Magic effects that explode on impact or swift firing arrows are all possible. Emerald's self-documented editor will explain each setting in-depth. You can ignore the “Update Projectile” message that pops up as you’re adjusting your settings as this will be covered in the next step.



Step 5) Once you have finished adjusting your projectile's settings, you can press “Update Projectile 1” button. This will apply and save the changes you have made to your projectile. When this happens for the first time, the projectile object will have a collider applied, its layer changed, a Rigidbody applied, and a EmeraldProjectile script applied to it. At any time, if you want to adjust your projectile’s settings, you will need to press this button so all the changes you’ve made are saved and updated to the EmeraldProjectile script. You will be prompted by the “Update Projectile” message when a projectile is in need of being updated.



Step 6) Lastly, you will need to adjust your AI's Attack Distance. This can be found under the projectiles settings. This setting controls the distance in which your AI can attack. The Too Close Distance will controls the distance to trigger an AI to backup if a target gets too close. It's worth noting that the Attack and Too Close distances are used by all of an AI's projectiles.



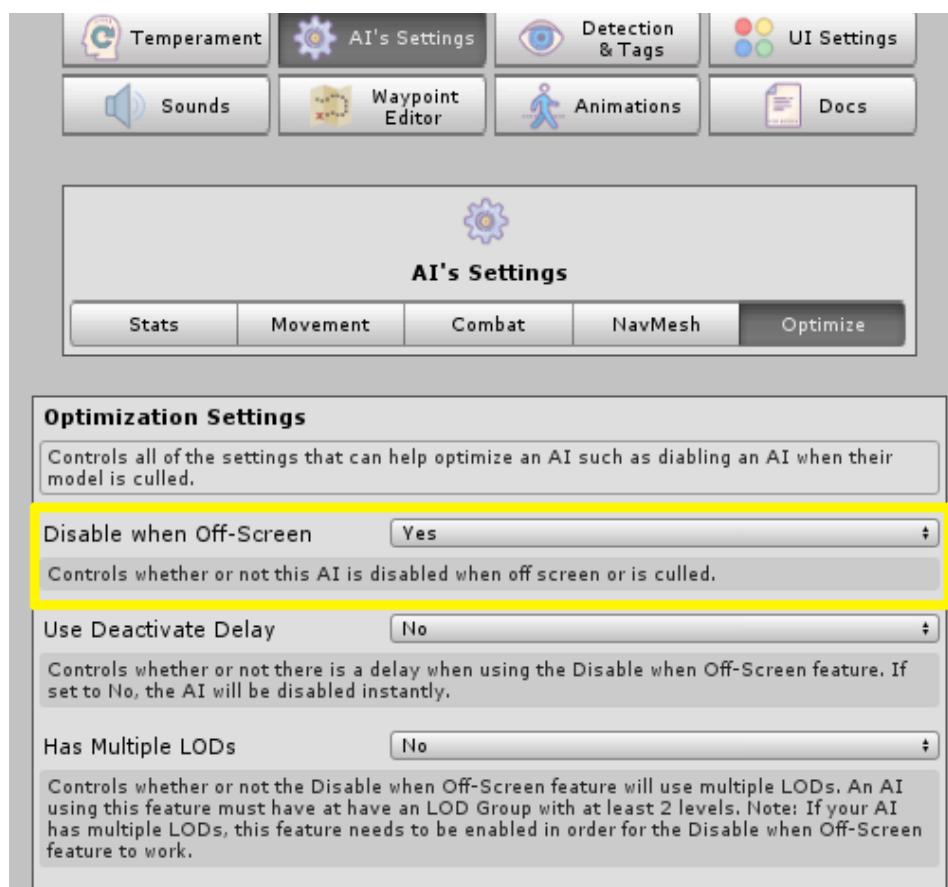
Step 7 (Optional) If you'd like to have more than 1 projectile, change the Total Projectiles Amount to the desired amount and repeat steps 1-5.

Continued Below

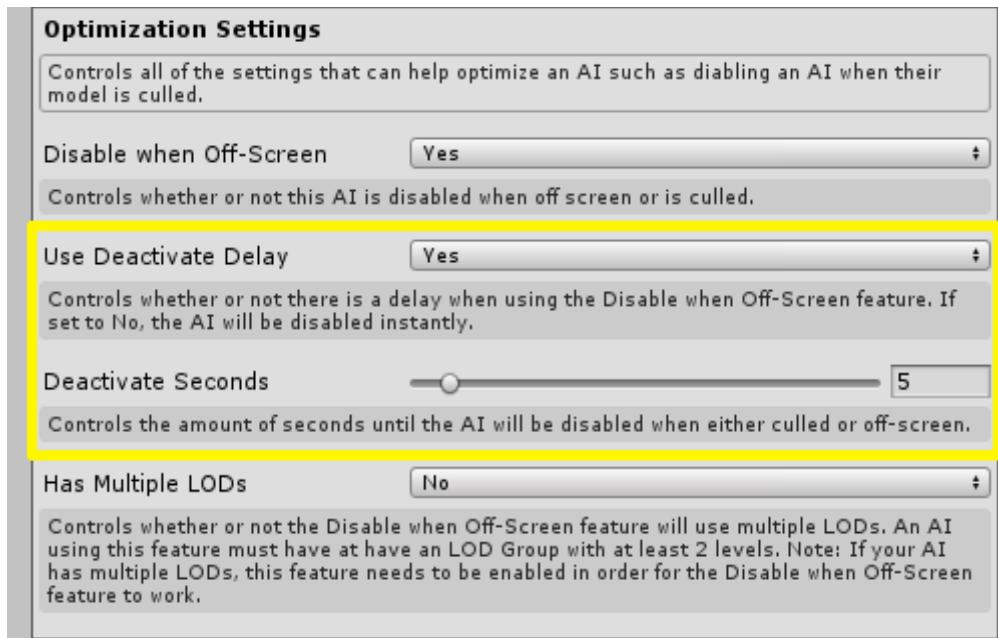
Setting Up an AI's Optimization Settings

Emerald has additional options to help further optimize your AI. This can help improve performance without having to sacrifice quality. This is achieved by disabling AI when they cannot be seen by the camera. It is recommended that this is handled during the initial setup process using the Emerald Setup Manager, but if this step was missed, it can be enabled with these steps.

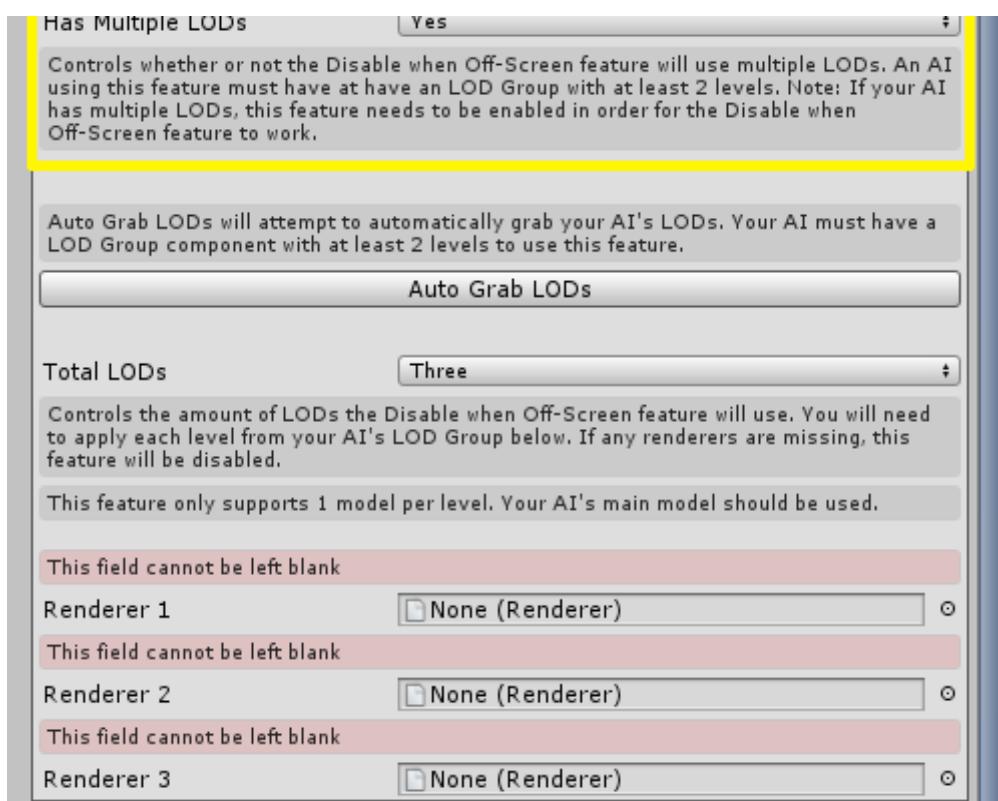
Step 1) To enable the optimization feature, go to the AI's Settings tab and then the Optimize section. You must have Disable when Off-Screen enabled for this feature to work.



Step 2) By default, an AI will be disabled instantly when is not visible by the camera. While this works for most game types, it might not for some. There is an additional option to have a delay before an AI is deactivated called Deactivate Delay. This lets you customize the seconds before an AI gets disabled after not being visible from the users set amount of seconds.



Step 3) If your AI's model has multiple LODs, you will need to set Has Multiple LODs to Yes. The optimization feature will not work properly if your AI has more than 1 LOD level. After you have done this, you will see additional options become visible. It is recommended that you use the Auto Grab LODs feature. This will automatically grab all of your AI's model's LODs and apply the needed settings. After you have pressed said button, there is nothing further you need to do. Emerald will have the optimize feature set for all of your AI's LODs.

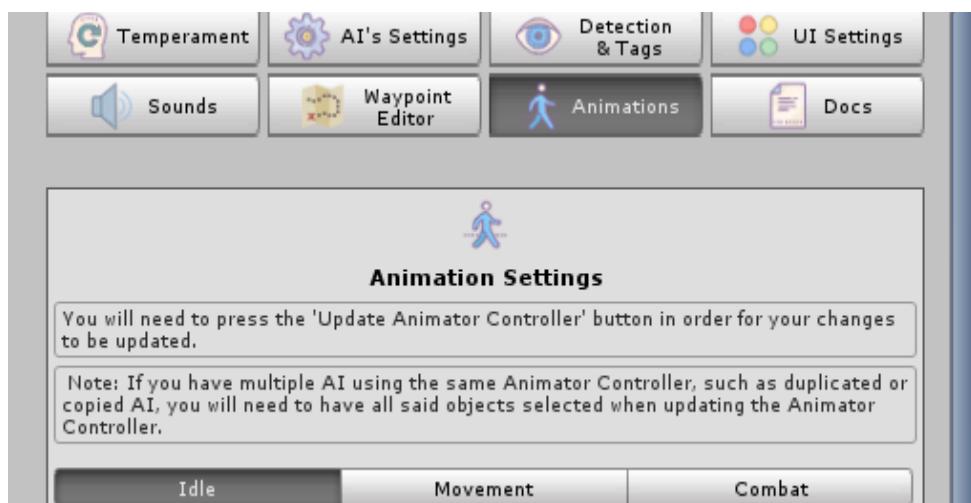


Setting Up an AI's Animations

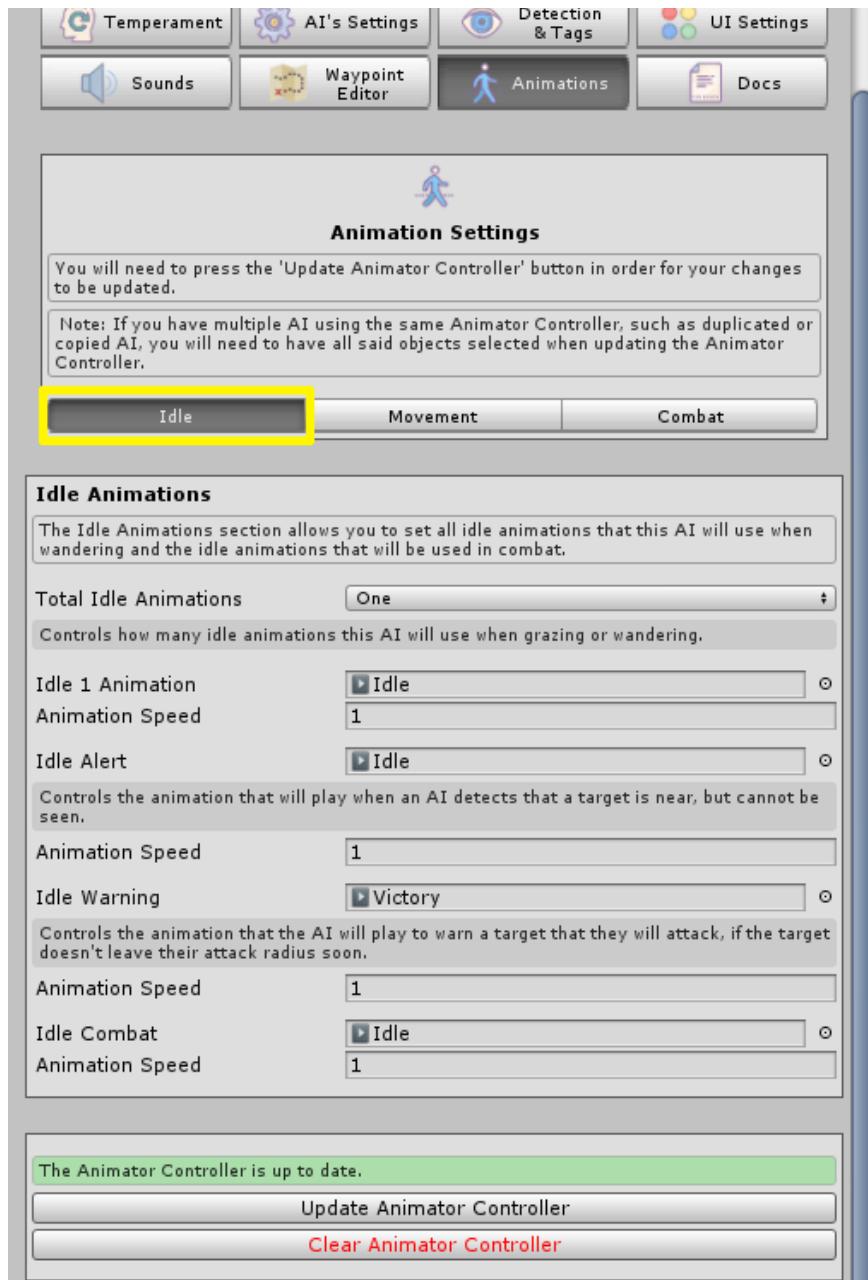
Emerald's Setup Manager makes setting up your AI's animations easy. However, there are additional settings to help get your AI's animations timed and looking good. If you haven't already done so, it is recommended that you use Emerald's [Setup Manager](#) for initially setting up your AI and its animations. If you have for some reason skipped setting up your animations through the Setup Manager, or you'd like to change/update your AI's animations, this guide will cover how to setup animations through the Emerald Editor.

Applying and Updating Animations Through the Emerald Editor

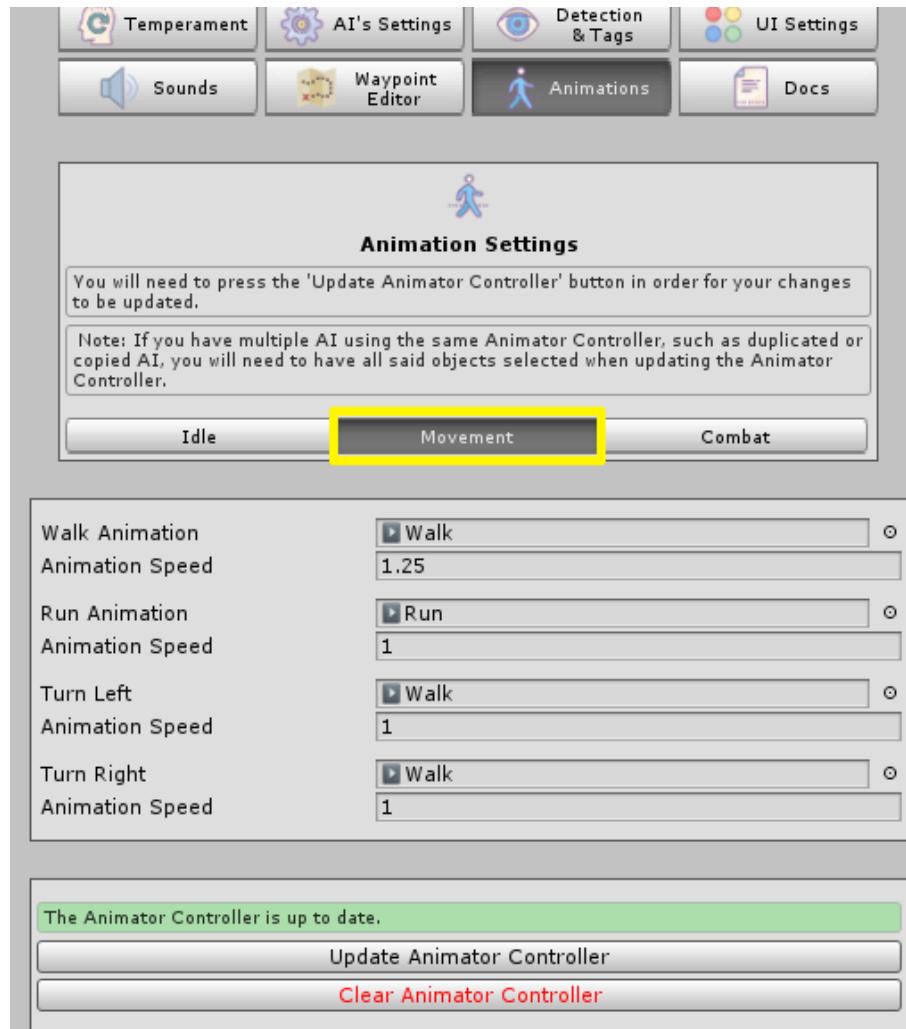
Step 1) Setting up and updating an AI's animations through the Emerald Editor is pretty straight forward. Upon selecting the Animations tab, you will see 3 sections; Idle, Movement, and Combat. Each category will consist of animations related to said categories.



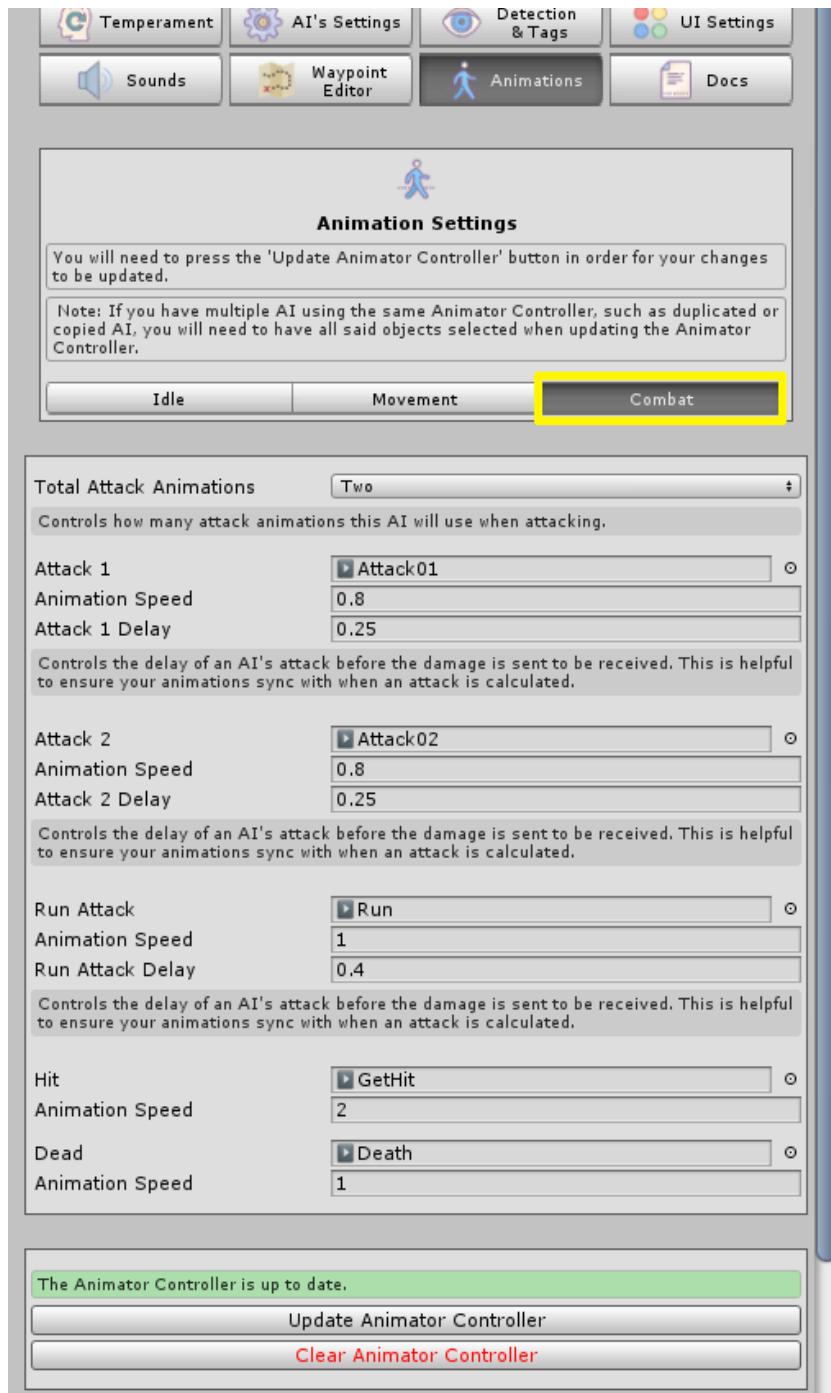
Step 2) To start off, select the Idle category. Here, you will find all of an AI's Idle animations such as its grazing, combat idle, alert idle, and idle warning animations. The Total Idle Animations controls how many idle animations your AI will use while wandering. These idle animations (idle 1-3) are played in order to ensure that none are played twice in a row. You will also see animation speed settings for each animation. These control how fast the animation will play.



Step 3) Next select the Movement category. This category covers all movement related animations such as walking, running, and turning. If for some reason your AI doesn't have a turn animation, and you would like your AI to play an animation while turning, an AI's walk animation will suffice. If you'd like to have no turing animations used, you can simply use an AI's idle animation. The animation speed settings here are important. In order to get believable walking and running movement, ensure that your movement animations aren't playing too slow or too fast. The key to believable footsteps is to match your AI's footsteps to your AI's speed.



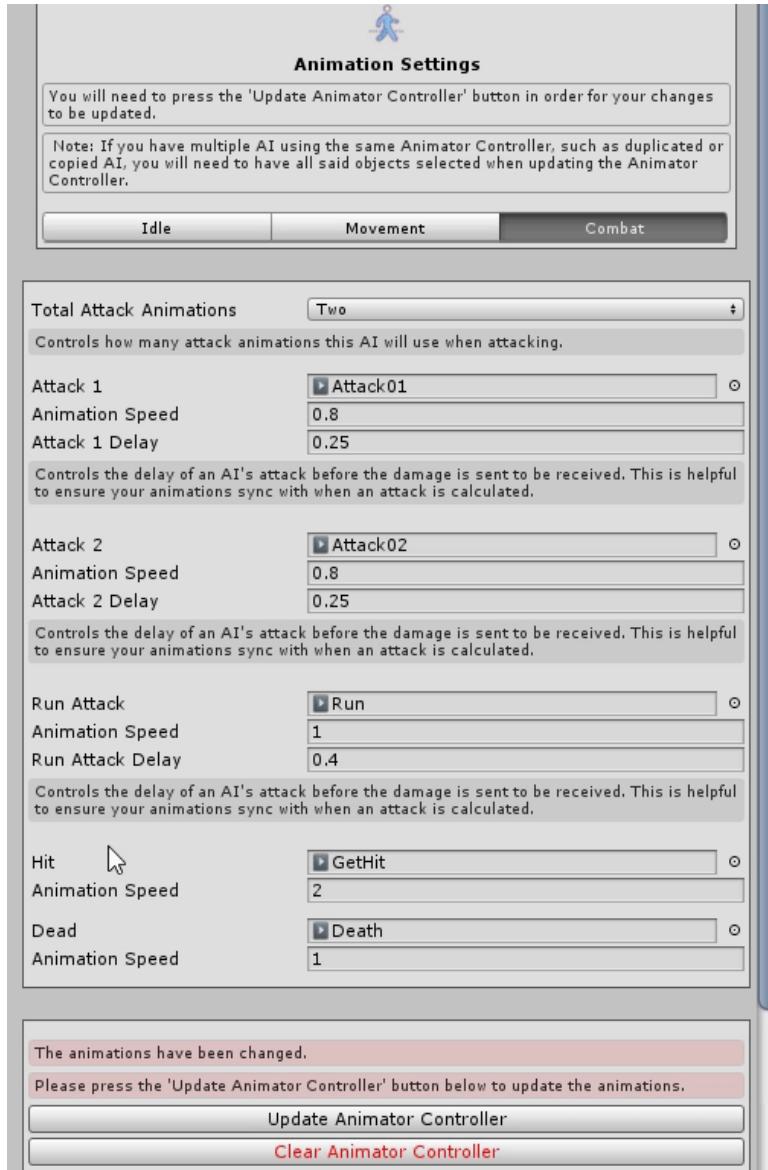
Step 4) Now, select the Combat category. This category covers all combat related animations such as its attack, getting hit, and death animations. This section has additional settings under each attack animation. These settings are called Attack Delays. These settings control the timing of each attack animation before they trigger a successful attack and apply damage to another AI or player.



Step 5) By now, you may have noticed the Animator Update buttons at the bottom of the Animations tab. This tab is important and is responsible for updating an AI's Animator Controller. When an animation has been altered, you will be prompted to update your AI's Animator Controller. You can wait until you have finished applying all needed animations before you do this. However, it is important to do this after you test the scene . Failing to do so will result in an AI's animations not being properly updated and applied to its Animator Controller. If for some reason you forget, Emerald will notify you with a Debug message in the Unity Console.

All you have to do is press the “Update Animator Controller” button and Emerald will update all the

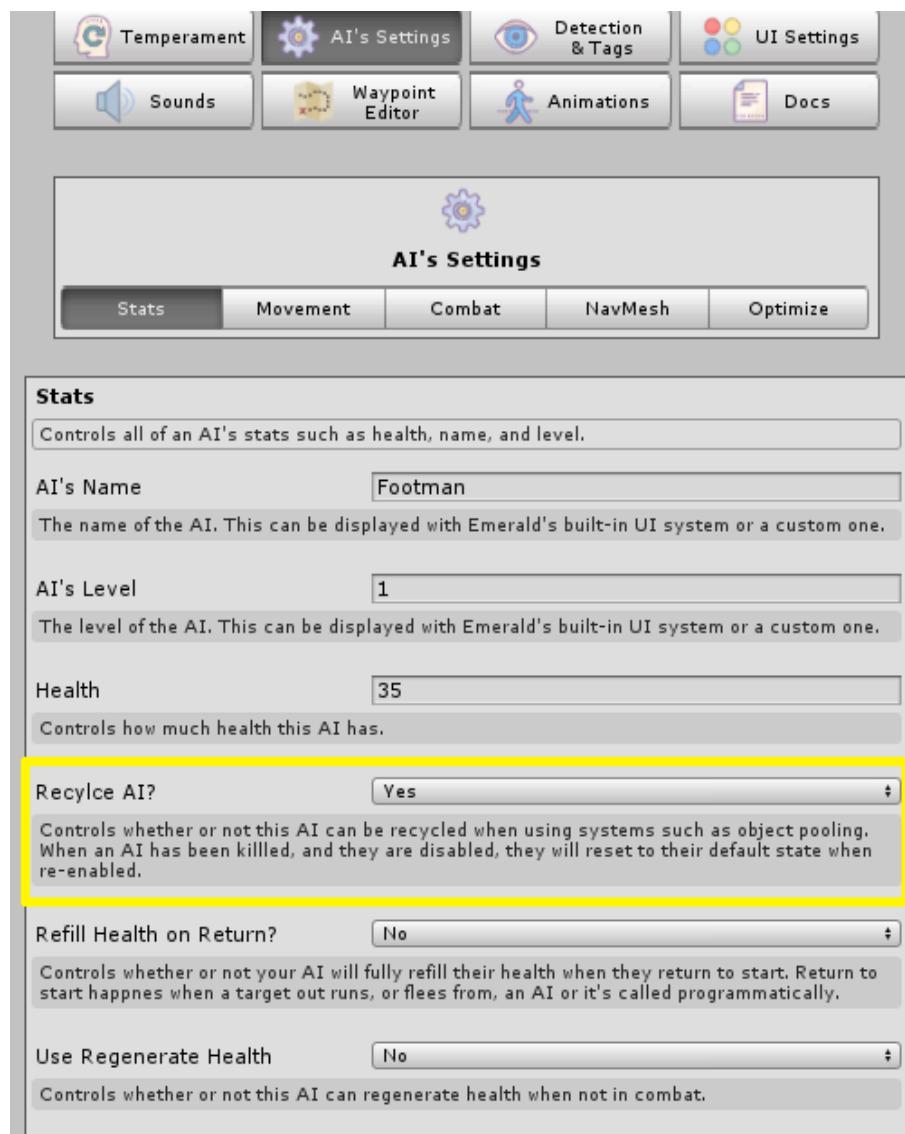
changes you've made. This button is not dependent on categories and updates all animations in all categories.



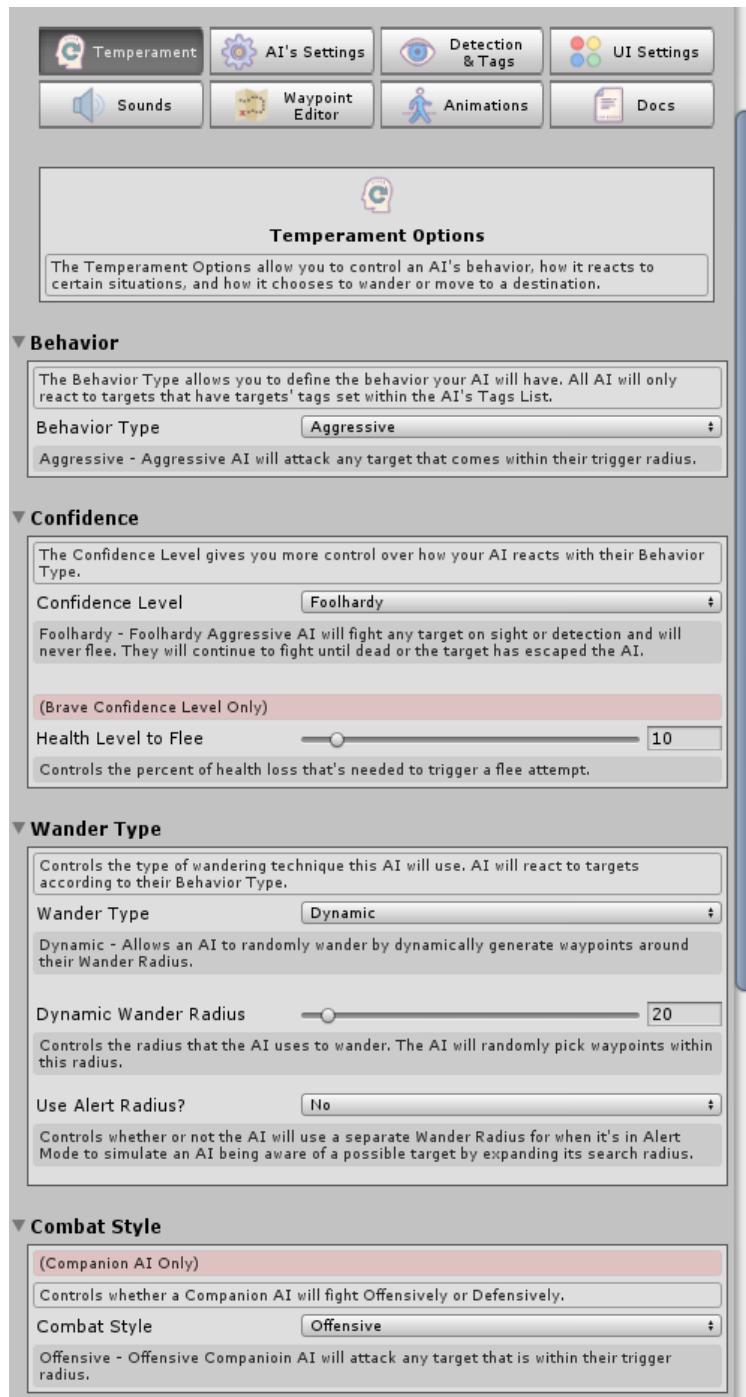
Setting Up an AI to be used with Object Pooling

If you'd like your AI to be usable with spawning systems that use object pooling, such as [Crux - Procedural AI Spawner](#), you will need to have Recycle AI set to Yes. This will allow AI that have

been killed to be reset back to their default settings before they are reused, given that they have been disabled after they have been killed.



Understanding an AI's Temperament



An AI's Temperament controls various settings of how an AI reacts to targets. Emerald makes things easy by having 5 preset behaviors that your AI can follow. They also have a Confidence level that gives you further control to make your AI's Behavior more unique and customizable. What an AI does with its Confidence Level varies based on its Behavior Type. Below, these settings will be further explained.

Behavior Types

Behavior Type	Description
Passive	Passive AI will not attack. They will simply wander around. If they are attacked, they will react according to their Confidence Level which is either to fight back or flee.
Cautious	Cautious AI will either flee or act territorial depending on their Confidence Level. Territorial AI will warn targets before attacking their target. An AI is set as territorial if their Confidence Level is set to Brave or higher.
Companion	Companion AI will follow around a target and help them fight. Companion AI will wander until their follow target is set. This is best done with a script and calling the public function SetTarget.
Aggressive	Aggressive AI will attack any target that comes within their trigger radius.
Pet	Pet AI will follow around a player target. They will not fight, engage in combat, or be targeted. They are simply for cosmetic purposes.

Confidence Level

What an AI does with its Confidence Level varies based on its Behavior Type. So, each Confidence Type has been categorized by Behavior Type. Note: The Pet and Companion Behavior Types do not use the Confidence Level setting.

Cautious AI

Confidence Level	Description
Coward	Coward Cautious AI will flee when they encounter a target.
Brave	Brave Cautious AI will become territorial when a target enters their trigger radius. If the target doesn't leave its radius before its territorial seconds have been reached, the AI will attack the target. They will attempt to flee once its health reaches the percentage you've set.
Foolhardy	Foolhardy Cautious AI will become territorial when a target enters their trigger radius. If the target doesn't leave its radius before its territorial seconds have been reached, the AI will attack the target. This AI will never flee and continue to fight until dead or the target has escaped the AI.

Passive AI

Confidence Level	Description
Coward	Coward Passive AI will wander according to their wander settings,

	but only flee when attacked.
Brave	Brave Passive AI will wander according to their wander settings, but only attack when attacked. They will attempt to flee once its health reaches the percentage you've set.
Foolhardy	Foolhardy Passive AI will wander according to their wander settings, but only attack when attacked. They will never flee and continue to fight until dead or the target has escaped the AI.

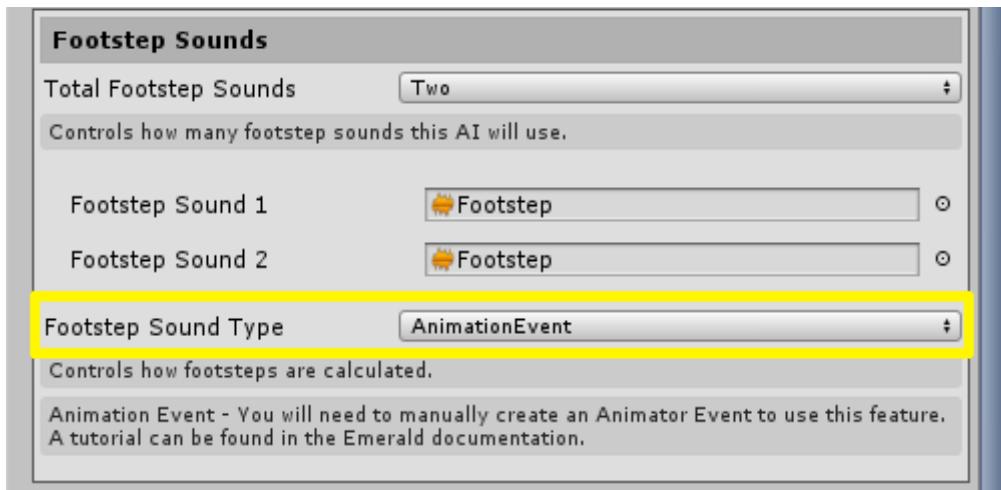
Aggressive AI

Confidence Level	Description
Coward	Aggressive AI cannot be set to Coward. AI with this setting will automatically be set to Brave on Start.
Brave	Brave Aggressive AI will fight any target on sight or detection, but attempt to flee once its health reaches the percentage you've set.
Foolhardy	Foolhardy Aggressive AI will fight any target on sight or detection and will never flee. They will continue to fight until dead or the target has escaped the AI.

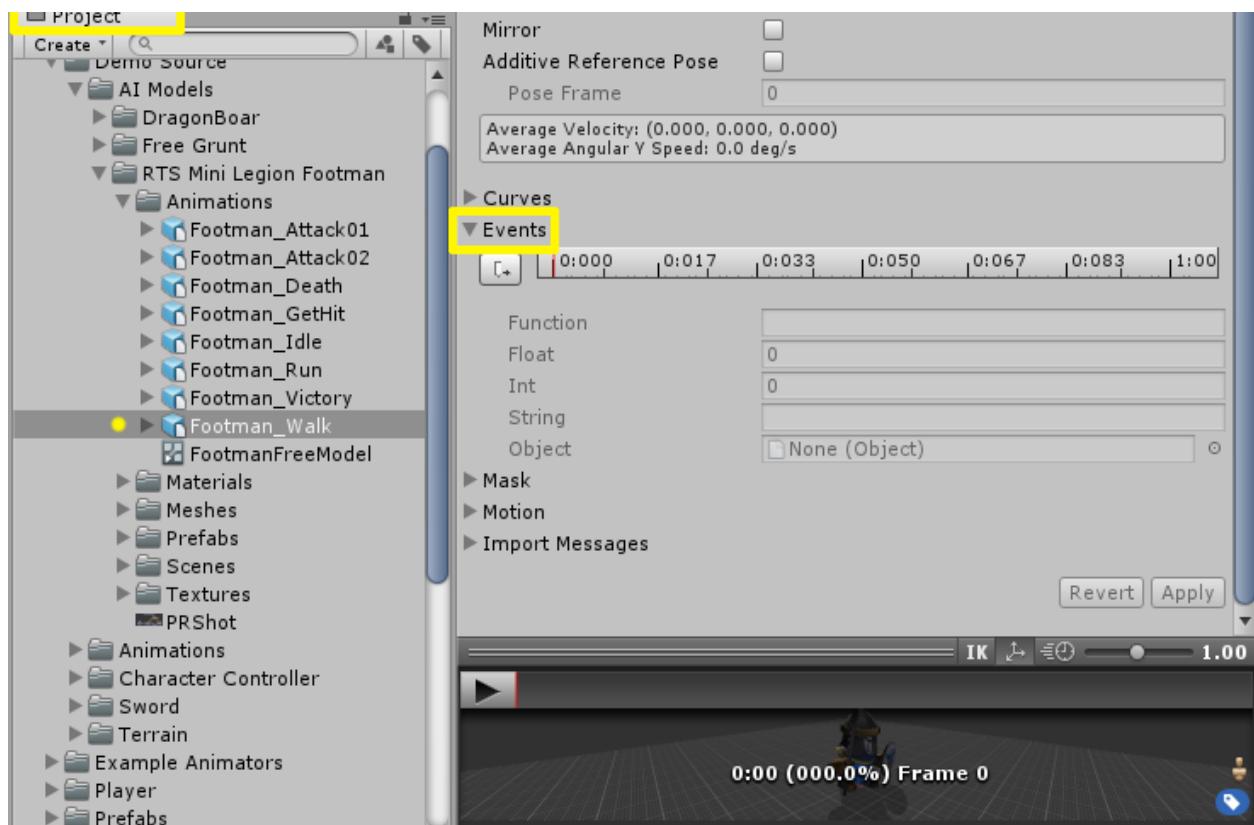
Adding Footstep Sounds Timed to Custom Animations

By default, Emerald's footsteps use the Seconds setting to time an AI's footsteps. While this is the easiest option, it does not allow footsteps to perfectly match an AI's walking and running animations. If you'd like your AI's footsteps to perfectly match your AI's animations, you can follow this guide to add Animation Events to your walk and run animations.

First you will need to go to the AI's Sounds tab and go to the Footstep Sounds section. Here, you will need to set your Total Footstep Sounds to at least one. Once you've done this, you will see the Footstep Sound Type option. Set the Footstep Sound Type to Animation Event, like shown below. Make sure that you have all needed sounds applied to the each sound slot.



Next, you will need to select your AI's walk animation. This can be done by going to the Project tab within Unity, selecting your AI's walk animation FBX object, and opening the Events foldout to expose the Animation Event options.

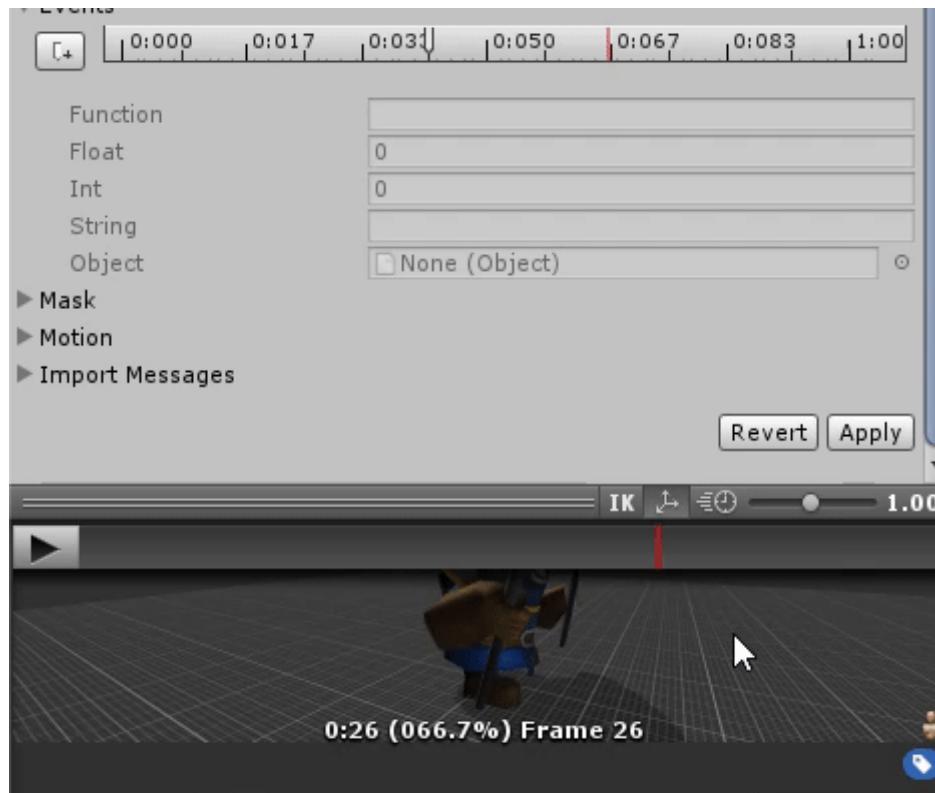


While still in Animation Event as explained above, you will need to create an Animation Event for your AI's walk animation. This can be done by finding the right frame that your AI's foot touches the ground and pressing the + button in the upper left corner. This may be done for both feet or just one. It really depends on your AI's animation. You can refer to the GIF below for a visual guide.

Before you press the Apply button, ensure that you have applied the **Emerald_AI** script to the

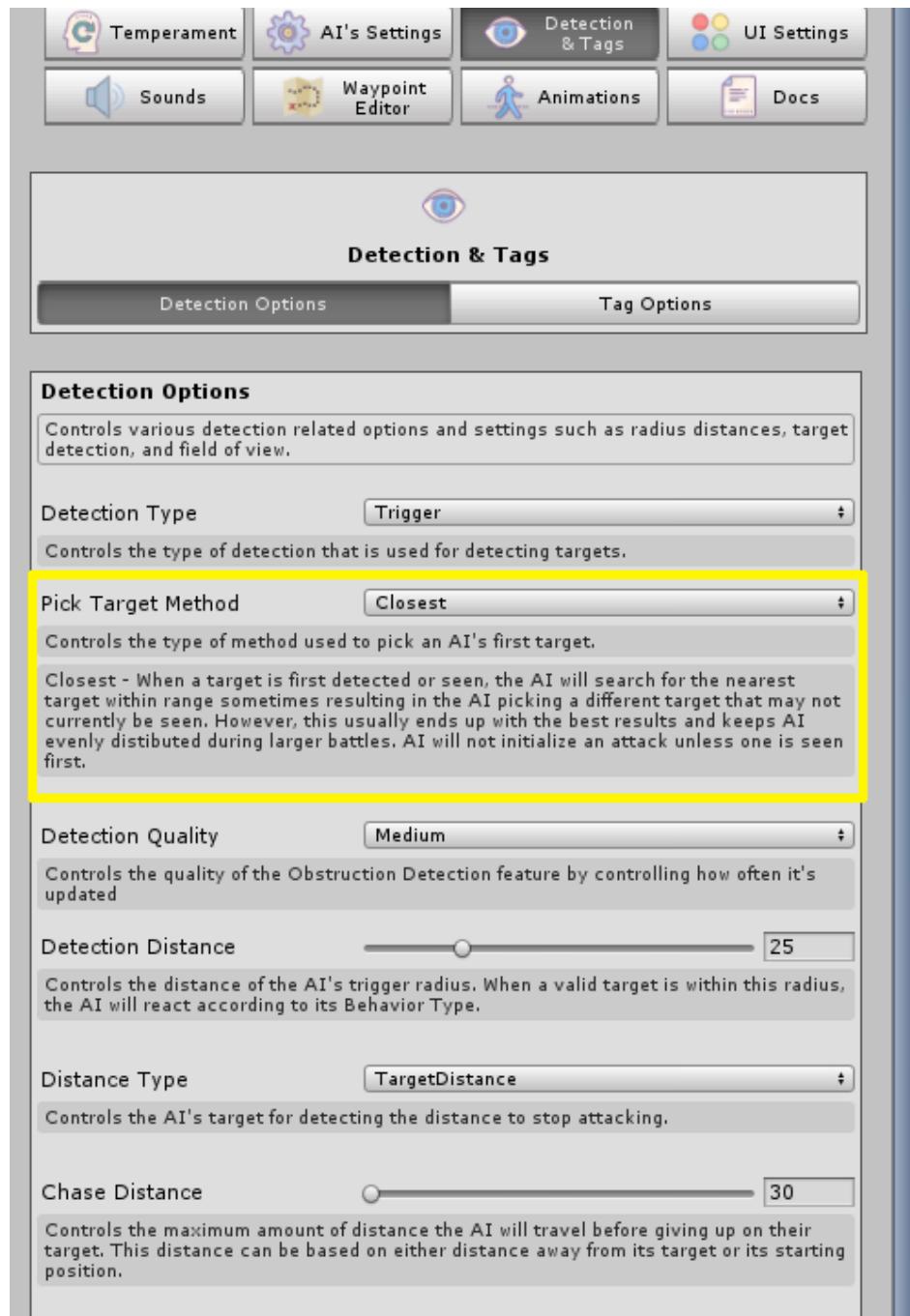
Object slot. This can be done by dragging the Emerald_AI script from the project tab onto the Object tab. Once you press Apply, you're all done. Your AI will play a footstep sound each time its foot hits the ground with its walk animation. What's perfect about this method is that it also consistently regardless of your AI's animation speed.

If you'd like to add footstep sounds to your AI's run animation, repeat the steps, but with the AI's run animation selected.



Understanding an AI's Target Picking Method

An AI's Target Picking Method is important to get the results you are looking for. Both options have their strengths, but it depends on the situation and playing style. The Target Picking Method can be found under the Detection Options and has two options, First Detected and Closest.



First Detected

The First Detected method can be useful for accurate small scale battles where you want your AI to visually see or detect their targets and allow your player to sneak up on an AI. This is more applicable to the Line of Sight Detection Type as this feature actually requires an AI to see their target before it can start to attack. The drawback to this method is that, when in large scale battles, it can make multiple AI all mark the same target resulting in clusters or groups that are too close.

Closest

Closest is useful for nearly all situations and is important for large scale battles. This method still requires an AI to visually see or detect a target with their Detection Radius, but it will search for targets within the area and assign the closest one, even though that may mean that the target may not be fully in sight. This feature also keeps AI evenly distributed and prevents AI from all picking

the same target resulting in unappealing battle formations that are much too close.

Comparisons of the First Detected and Closest Target Picking Method in a larger battle situation.

First Detected



Closest

