A.I. Behavior - Made Easy

Overview / Purpose Statement

A.I Behaviors is designed to accomplish the task of creating simple and complex behaviors for games. Structured around a straight forward and easy to understand set of options, the Inspector driven system provides the user with control and flexibility.

Components

Agent FSM
FSM Animation States
Character Animator
Agent FSM Animation State

Agent FSM *(script)*

General

Player Tags

Add	– Add tags for the Agent to use in searching
Initial State	– Sets the starting state of the Agent

Agent Properties

Raycast Layers	– Layermask that checks for what the Agent is looking at *(Default Player is unchecked)*
Use Sight Falloff	– Toggle option for setting a fall off for the line of sight
Sight Distance	– The distance the Agent is able to 'see'/ check for objects *(If Sight Falloff is checked)*
Sight FOV	– Set the Angle for Field Of View
Eye Position	– The height / location used to check / 'see'
Health	– An Example Script which sets up a basic health for the Agent

Animation Component Callback

Component	– The script/component to which the following method will be called when an animation is told to play
Method	– The method which is called when an animation is told to play. This method should include the code that actually plays the animation.

Triggers

Line of Sight – *Check with the Agent Properties to determine if a target can be seen*

Change to State	– If true, set the new state of the Agent to the selection

No Player in Sight – *Check if no target can be seen*

Change to State	– If true, set the new state of the Agent to the selection

Within Distance – *Check if a player tag is inside the selected distance*

Within Distance	– Set the distance the Agent uses for checking
Change to State	– If true, set the new state of the Agent to the selection

Beyond Distance – *Check if a player tag is outside the selected distance*

Beyond Distance	– Set the distance the Agent uses for checking
Change to State	– If true, set the new state of the Agent to the selection

Timer – *Set an amount of time to count down from before the trigger is activates*

Timer Duration	– Set the amount of time
Plus or Minus Duration	– Set a range above or below
Change to State	– If true, set the new state of the Agent to the selection

Low Health – *Check the minimum health of the player*

Health Amount	– Set the health low trigger amount
Change to State	– If true, set the new state of the Agent to the selection

High Health – *Check the maximum health of the player*

Health Amount	– Set the health high trigger amount
Change to State	– If true, set the new state of the Agent to the selection

In Player View – *Check if the Agent is within the View of the Player*

Change to State	– If true, set the new state of the Agent to the selection

Animations

Add	– Add Animations for the Agent to use for current State

(Additional animations can be used and are randomly played during the state)

Cooldown Properties

Cooldown Duration	– Set how long the Agent must wait before changing to the next State
Has Cooldown Limit	– Set true if there is an imposed limit
Cooldown Limit	– Set number of times allowed to recover
Limit exceeded State	– Select the state the Agent will change to when the limit is reached

Audio Properties

Audio Clip	– Load audio specific files for each state of the Agent
Loop Audio	– Set true if the audio file should repeat
Volume	– Adjust the audio volume for the clip to play

States

Idle – *The Agent at rest waiting for trigger events*

Patrol – *The Agent navigates based on waypoint locations*

Movement Options

Patrol Speed	– Set the speed the unit moves while patrolling
Rotation Speed	– Set how quickly the turns to face the next waypoint
Patrol Mode	– Define the animation mode for the patrol
Continue Previous Patrol	– Set how the Agent continues when at the end of a Patrol Group
Lock To Path	– Keep the Agent on the Path
Use Node Action	– Allow a Node Script to activate

Patrol Points	
Patrol Point Group	– Load a Collection of transforms for the patrol path
Distance Threshold	– Set range to patrol point before searching for next

Follow – *The Agent trails the position of a chosen target*	
Standard Follow Properties	
Follow Speed	– Set the Agent speed for trailing the target
Chase Properties	
Chase Speed	– Increase to a chase speed when in range
Start Chase Distance	– Set range to target activate
Break Chase Distance	– Set range to target to stop chase
Chase Distance	– Set how far the chase can last
Move Direction	
Horizontal Move	– Allow movement along the Horizontal
Vertical Move	– Allow movement along the Vertical
Other Properties	
Stopped When Target Reached	– The Agent will stop following when it reaches the target
Passing Distance	– Set the amount of distance the Agent is allowed to go pass the target

Flee – *The Agent navigates away from a selected target*	
Flee Speed	– The speed at which the Agent will move away from the target
Start Flee Distance	– Set the range at which the Agent begins this action
Flee To Target	– Load a target object for the Agent to move to
Flee Mode	– Set type to Nearest Tagged Object / Fixed Target / Direction
Use nearest Object with Tag	*– Set true for near tag use*
Flee To Target	*– Assign object for the Agent to move toward*
Flee Direction	*– Set a Vector Direction for the Agent to move toward*
Other Properties	
Flee Health Amount	– The Agent will flee if this amount of health is reached

Seek – *The Agent looks for the selected set of Items*	
Seek Item with Tag	– Set the tag identifier the Agent looks for

Seek Ammo – *The Agent looks for the selected set of Items*	
Seek Item with Tag	– Set the tag identifier the Agent looks for *(ammunition based)*

Melee Attack – *The Agent combats an object at a close range*	
Attack Distance	
Min Distance	– Minimum range the Agent must be from the target before attacking
Max Distance	– Maximum range the Agent must be from the target before attacking
Attack Values	
Attack Speed	– Set how quickly the attack occurs
Attack Damage	– Set the value of damage to inflict on the target
Reloading Values	
Reload Time	– Set the amount of time it takes to reload the Agents melee attack
Has Reload Limit	– Check if the Agent has a reload limit value
Reload Limit	– Assign a number limit for reloads
Animation Settings	

Reload Animation	– Assign an animation for reloading
Attack Animation	– Assign an animation for attacking
Attack Point	– The point in the animation when the AI will call the following attack method
Attack Method	
Attack Method	– The method that's called on the GameObject when the attack point is reached for the attack animation
If none – Example Box	– Set the method for the attack

Ranged Attack – *The Agent combats an object at a far range*

Attack Distance	
Min Distance	– Minimum range the Agent must be from the target before attacking
Max Distance	– Maximum range the Agent must be from the target before attacking
Attack Values	
Attack Speed	– Set how quickly the attack occurs
Attack Damage	– Set the value of damage to inflict on the target
Reloading Values	
Reload Time	– Set the amount of time it takes to reload the Agents melee attack
Has Reload Limit	– Check if the Agent has a reload limit value
Reload Limit	– Assign a number limit for reloads
Animation Settings	
Reload Animation	– Assign an animation for reloading
Attack Animation	– Assign an animation for attacking
Attack Point	– The point in the animation when the AI will call the following attack method
Attack Method	
Attack Method	– The method that's called on the GameObject when the attack point is reached for the attack animation
If none – Example Box	– Set the method for the attack

Defend – *The Agent Protects itself*

Defensive Armor Bonus	– Set the number of bonus points the defense has
Lock Position	– Keep the Agent locked to its current position
Lock Rotation	– Keep the Agent locked to its current rotation

Got Hit – *The Agent is being assaulted*

Change – The Agent GameObject may be substituted for a different GameObject

Change Into	– Assign the GameObject that will replace the current Mesh

Heal – *The Agent may restore its properties*

Seek Item with Tag	– Set the tag identifier the Agent looks for *(health based)*

Help – *The Agent responds to the needs of other Agents nearby*

Within Help Point Distance	– Set the range of influence for helping the Agent
Help Move Speed	– Set the Speed of the Agent while moving in a Help state
State when Help Point Reached	– Assign a new state to the Agent when the destination is reached

Get Help – *The Agent sends messages to other Agents requesting assistance*

Get Help from Tagged Objects	– Set tag to the Agents that will help
Help Radius	– Set the range of influence for helping the Agent

Dead – *The Agent is now at the end of its game life*

FSM Animation States *(script)*

General

Player Tags

Is 3D	– If the animation is using the built in Animation system check this
Add	– Add an animation File to the list
U	– Move Item Up in the list
D	– Move Item Down in the list
Remove	– Delete the Item from the list

Animation Properties

Name	– Set the name of the animation
Speed	– Set the speed of the animation
Animation Wrap Mode	– assign a animation playback mode *(Default / Clamp / Loop / Ping Pong / Clamp Forever)*

FSM Basics

An AI FSM

It is a collection of states and transitions that outline a path of actions that may occur.

State

A state is a position in time. For example, when you are at the bus stop, you are currently in a waiting state. The 'Flow' looks like this – *Enter Action / Execute Action / Exit Action*

Event

An event is something that happens in time. For example, the bus has arrived.

Action

A task performed given a certain event that occurred. For example, you enter the bus.

Transition

A link between 2 states which may be unidirectional or bidirectional.

Classes

There are two major classes.

State

The state class is responsible for setting transitions, specifying actions, and pointing to the right function that represents the action.

FSM

The FSM class is a collection of states. Its sole purpose is to function as a controller between states and execute the necessary state actions based on the events passed to it.