

# Unreal FPS

Unreal FPS Manual  
Last Update 07.06.2018

Publisher: Infinite Dawn  
Author: Tamerlan Favilevich  
Copyright © Tamerlan Favilevich 2017 - 2018 All rights reserved.

# Contents

<b>1. Introduction</b>	3
<b>2. Getting Start</b>	4
<b>3. Player</b>	5
3.1 Create Player from the Tab	5
3.2 Create Player from scratch	5
3.3 Controller	6
3.4 Inventory	8
3.5 Player Health	9
3.5.1 Health	9
3.5.2 KillCam	9
3.5.3 OnDeadEvent	9
3.6 Input Manager	10
3.6.1 Creating the input configurations	10
3.6.2 Save Play Mode Tweaks	10
3.7 Spawn Manager	11
<b>4. Weapon</b>	12
4.1 Create Weapon	12
4.2 Create FPWeapon from the scratch	13
4.3 Weapon Identifier	14
4.4 Weapon Animation Handler	14
4.5 Weapon Attack Handler	15
4.5.1 Shoot Parameter	15
4.5.1 Spread System	15
4.6 Weapon Reload Handler	16
4.8 Ray Bullet	16
4.8.1 Create Ray Bullet	16
4.8.2 Setting-up Ray Bullet	17
4.9 Physics Bullet	18
4.9.1 Create Physics Bullet from the tab	18
4.9.2 Create Physics Bullet from the scratch	18
4.9.3 Physics Bullet	19
4.7 Pickable Weapon	20
4.7.2 Create Pickable Weapon	20
<b>5. AI</b>	21
5.1 Create AI for the Tab	21

5.2 Create AI from scratch	21
5.3 AI Handler	22
5.4 AIHealth	23
5.5 AIFieldOfView	23
<b>6. Behaviour Manager</b>	<b>24</b>
6.1 Create Class	25
6.2 Create Behaviour Asset	26
<b>7. Plugins</b>	<b>28</b>
7.1 Install Plugins	28
7.2 Add Plugins	29
7.3 Plugin folder	30
<b>8. Preferences</b>	<b>31</b>

# 1. Introduction

Unreal FPS is a framework for the Unity engine, designed to create powerful games in the genre of shooter, action, adventure, adventure and more. With Unreal FPS, you can create completely different and unique games. This framework greatly simplifies the development of the game, you can use it both to write a completely new game, and to improve the already developed or developed game.

Quality. Unreal FPS is constantly expanding, regular updates that improve performance, and new features are constantly added.

Support. We are always glad to our clients and we maintain constant contact with them, answer all questions and even help in the development of their projects, help write scripts, etc., our support is one of the best in the Asset Store.

Updates. We release only free updates, we are adherents of the principle "Once you pay and use". Absolutely all major updates and additions to the functionality will be free.

## 2. Getting Start

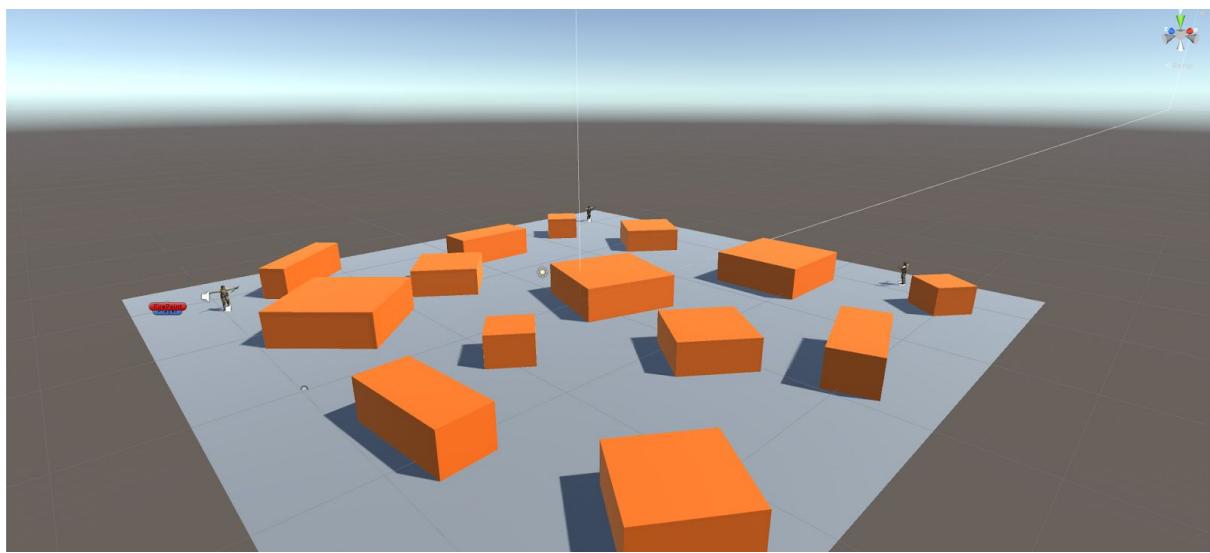
If you want to immediately try the basic functions, open the demo scene that is in this folder:

Unreal FPS → Scene → Demo Level → DemoLevel.scene



You can also try artificial intelligence, open one of the demo scenes that are in this folder

Unreal FPS → Scene → Demo Level AI

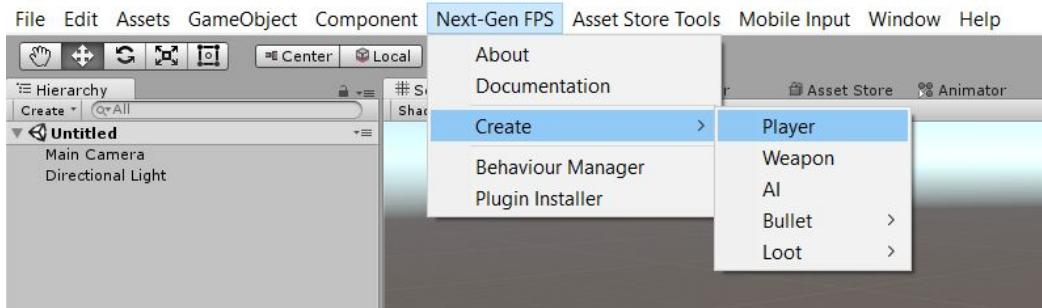


# 3. Player

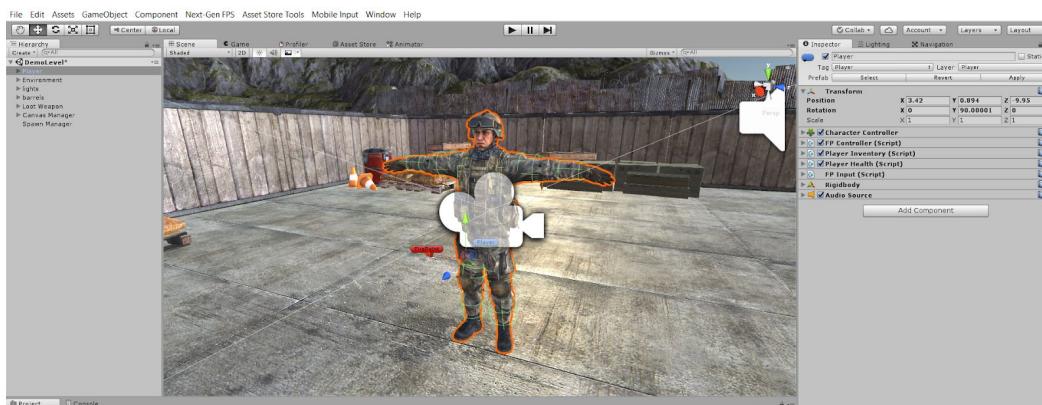
## 3.1 Create Player from the Tab

For the create player go to the tab:

Unreal FPS → Create → Player



## 3.2 Create Player from scratch

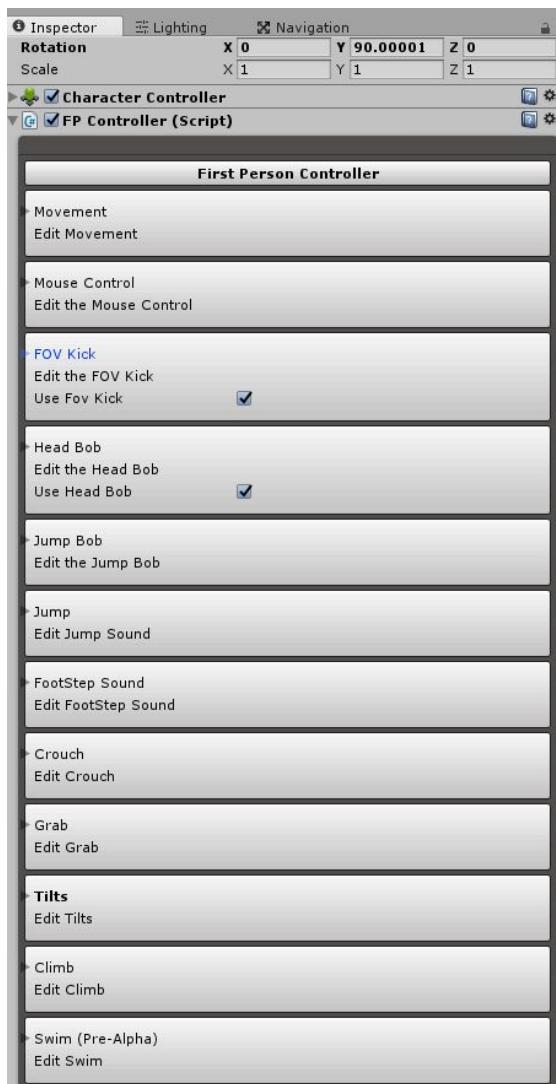


1. Create empty GameObject and name it Player
2. Create a plane or any kind of floor with collision and place the player GameObject above it.
3. For player set Tag - "Player" and Layer - "Player"
4. Drag the FPCController script onto Player GameObject and set-up FPCController
5. Drag the PlayerHealth script onto Player GameObject and set-up PlayerHealth
6. Drag the PlayerInventory script onto Player GameObject and set-up PlayerInventory
7. Add CharacterController Component onto GameObject Player
8. Add Rigidbody Component onto GameObject Player and set false for property useGravity
9. Add AudioSource Component onto GameObject Player
10. Create Camera name it "FP Camera" and set it child for Player

11. For FP Camera set Tag - "FP Camera"
12. In FPCamerain CullingMask disable layers: Weapon, Player, Remote Body
13. Create one more Camera, name it WeaponLayer and set it child for WeaponCamera
14. For Weapon Camera set Tag - "Weapon Layer"
15. In WeaponLayer, in Culling Mask enable only one layer: Weapon

### 3.3 Controller

In the Unreal FPS a powerful controller system, which is easy to configure due to the unique advanced editor



Unreal FPS have documentation inside editor

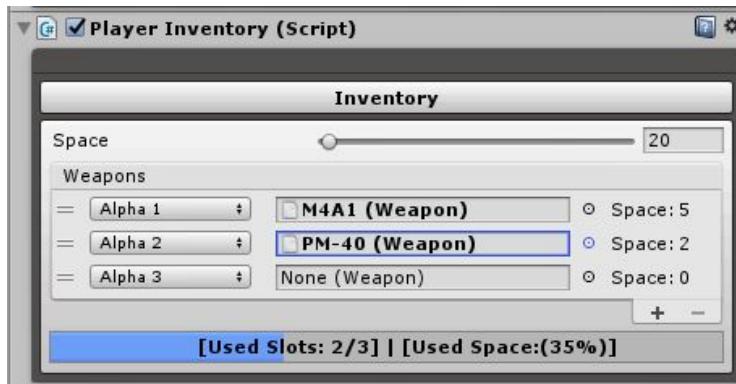
For get information about variables, hover your mouse over the variable about which you want to know the detailed information.



## 3.4 Inventory

Unreal FPS has an inventory system in which you can store various items not only weapons.

Also it has a unique editor supporting drag and drop function.



The inventory space is the maximum space that a player can carry.

You can also add an unlimited number of slots, this is how many items you can carry simultaneously.

The item will be added to the slot only if the maximum space of the inventory is not exceeded when the item is added.

Example:

Inventory Space = 20

Slot count = 3

At the moment in the slot added 2 Item with space 5,  $2 = 7$

Available inventory space = 13

You can pick up an item the space of which less or equals 13.

Each slot has a KeyCode parameter by which it can be called.

Before starting the game, you create the required number of slots and set Space, also you can pre-insert in the slot the item, that you want the player to have at the start. You can also change Inventory Space and Slot Count for this read the API.

## 3.5 Player Health

Base class for manage Health of the Player



### 3.5.1 Health

Health - current health count;

Start Health - health count that will be specified at the start

Max Health - The maximum health count that can get the player.

### 3.5.2 KillCam

Camera - Camera that be used for show player

Model - Player Body

Look At - The point at which the camera will focus

Radius - The minimum radius of collision detection

### 3.5.3 OnDeadEvent

This is events that will be called when player dies

## 3.6 Input Manager

Unreal FPS use very powerful Input Manager created by [daemon3000](#).

### 3.6.1 Creating the input configurations

- Go to the Luminosity > Input Manager menu and select Create Input Manager.
- In the inspector press the Input Editor button. The first time you open the *Input Editor* you will be prompted to overwrite your project's input settings. You can also do it from the File menu of the *Input Editor* at a later time.
- Use the options available in the Edit menu of the *Input Editor* to set up your input configurations.

It is recommended that you have only one *Input Manager* in your game. Add it in the first scene and enable Don't Destroy On Load in the inspector. You should also enable Ignore Timescale if you pause the game by setting the timescale to zero.

### 3.6.2 Save Play Mode Tweaks

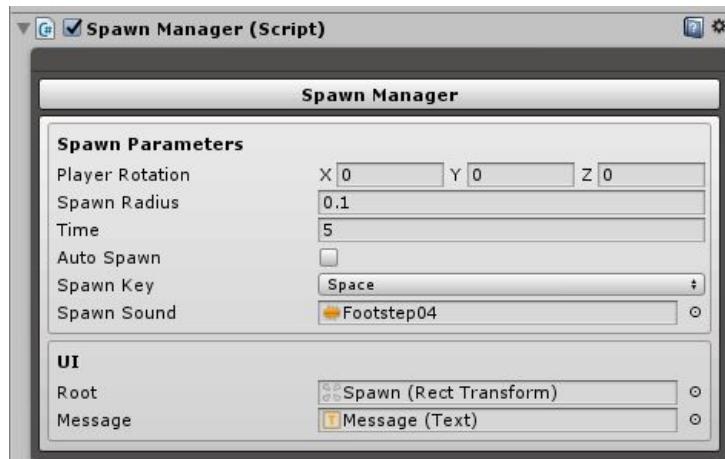
To save tweaks made during play-mode press the *Create Snapshot* button located in the Inspector.

After you exit play-mode press the *Restore Snapshot* button to restore the changes.

For get more information go to link: [Documentation](#)

## 3.7 Spawn Manager

Unreal FPS has a built-in ready-to-use Player spawn system



Spawn manager automatically finds the player on the scene (if the player is not found, manager auto creates it), if the player dies respawns him.

Player Rotation - Player Rotation then spawn

Spawn Radius - Spawn zona

Time - Time before spawn

Auto Spawn - If True player Player automatically spawn after the time runs out, if False will wait for pressing the button (Spawn Key)

Spawn Key - Key for spawn Player

Spawn Sound - Sound played when player spawned

Root - Root object of spawn message

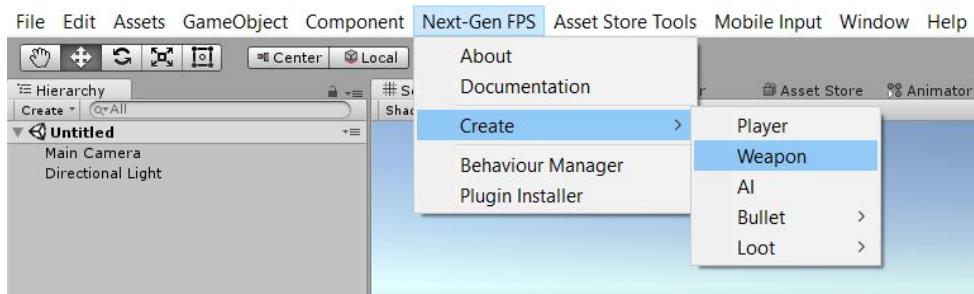
Message - Spawn message

# 4. Weapon

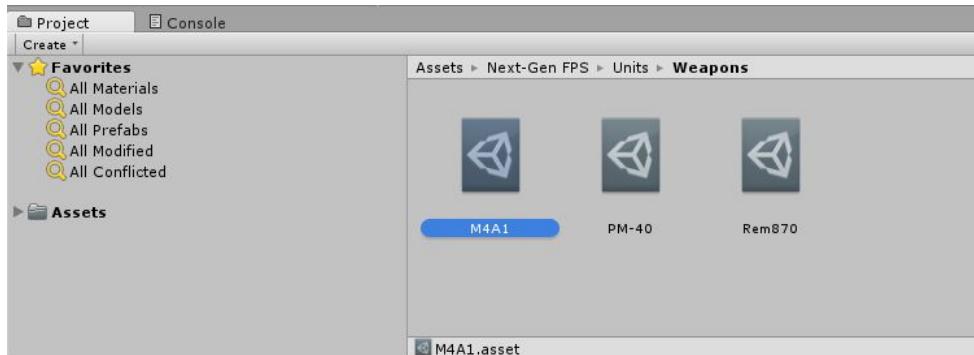
## 4.1 Create Weapon

Each weapon has an identifier, for create it go to the tab:

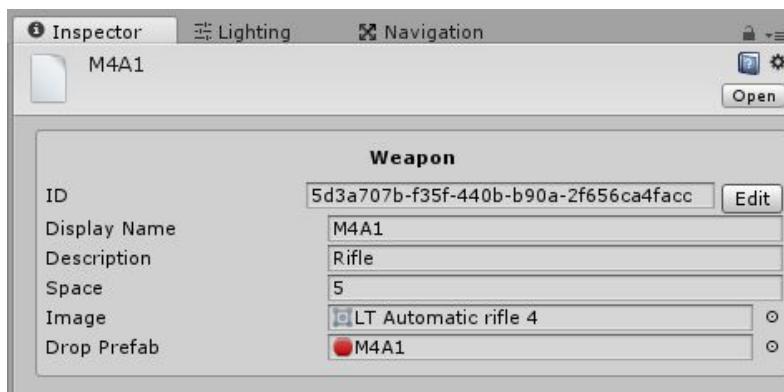
Unreal FPS → Create → Weapon



After this, a file must be created



Click on it to configure it



ID - Automatically generated Weapon ID, you can edit it by clicking on the Edit

Display Name - Weapon name displayed on HUD

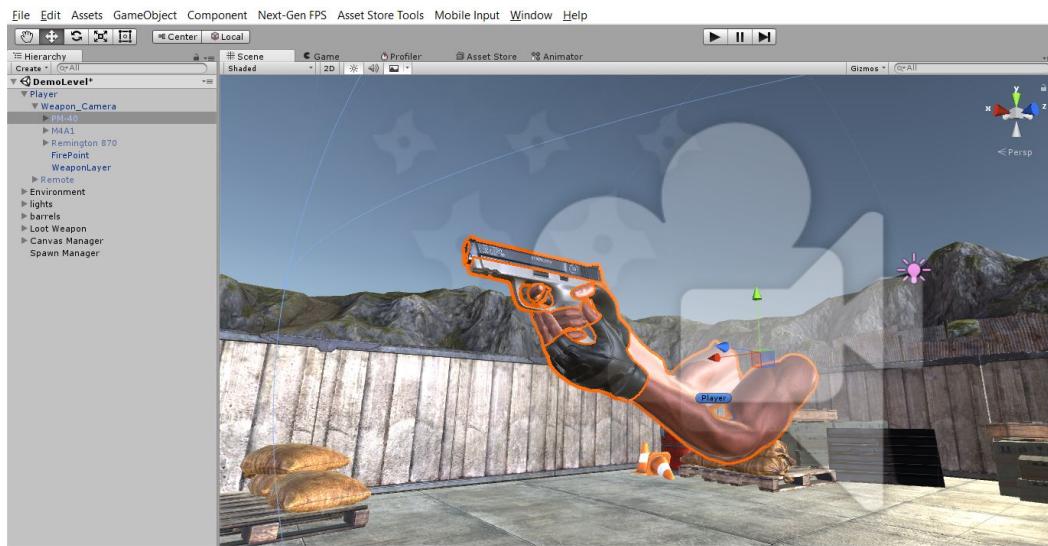
Description - Weapon description

Space - What is the weight of the weapon will take up in the inventory

Image - Weapon image displayed on HUD

Drop Prefab - Weapon prefab that will appear when Player drop this weapon

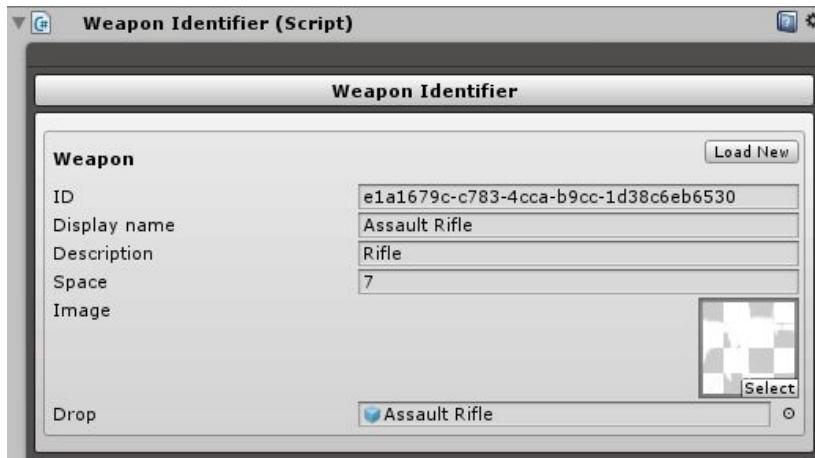
## 4.2 Create FPWeapon from the scratch



1. Create Weapon Identifier
2. Create empty GameObject and name it for example Weapon
3. Set the Weapon GameObject child for WeaponCamera
4. Add in Weapon weapon model
5. For Weapon set Tag - “Weapon”, set Layer - “Weapon” and say “Yes, change children”
6. Add Animator Component onto GameObject Weapon
7. Drag the Weapon Behaviour script onto Weapon GameObject and set-up Weapon Behaviour
8. Drag the Weapon Animation Handler script onto Weapon GameObject and set-up Weapon Animation Handler
9. Drag the Weapon Attack Handler script onto Weapon GameObject and set-up Weapon Attack Handler
10. Drag the Weapon Reload Handler script onto Weapon GameObject and set-up Weapon Reload Handler
11. Add Audio Source Component onto GameObject Player

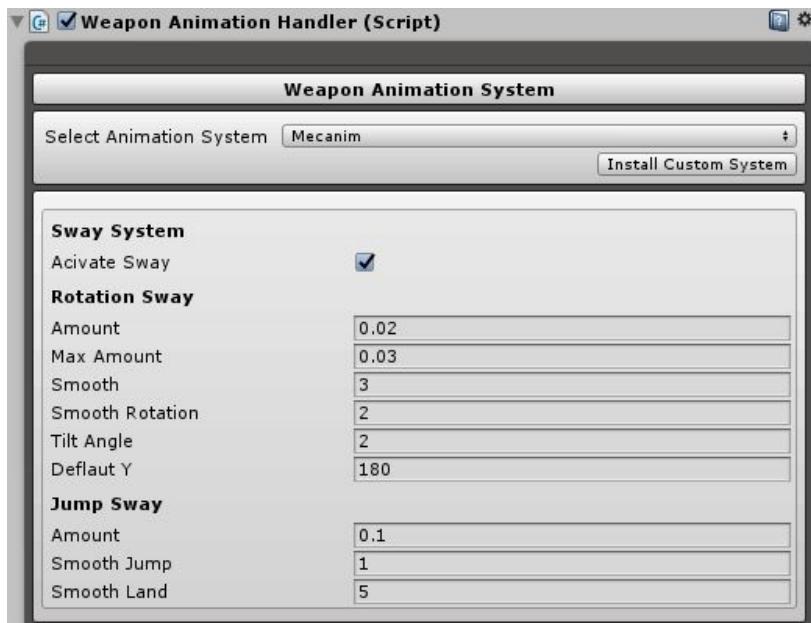
## 4.3 Weapon Identifier

Weapon Identifier The basic script that identifies the weapon.



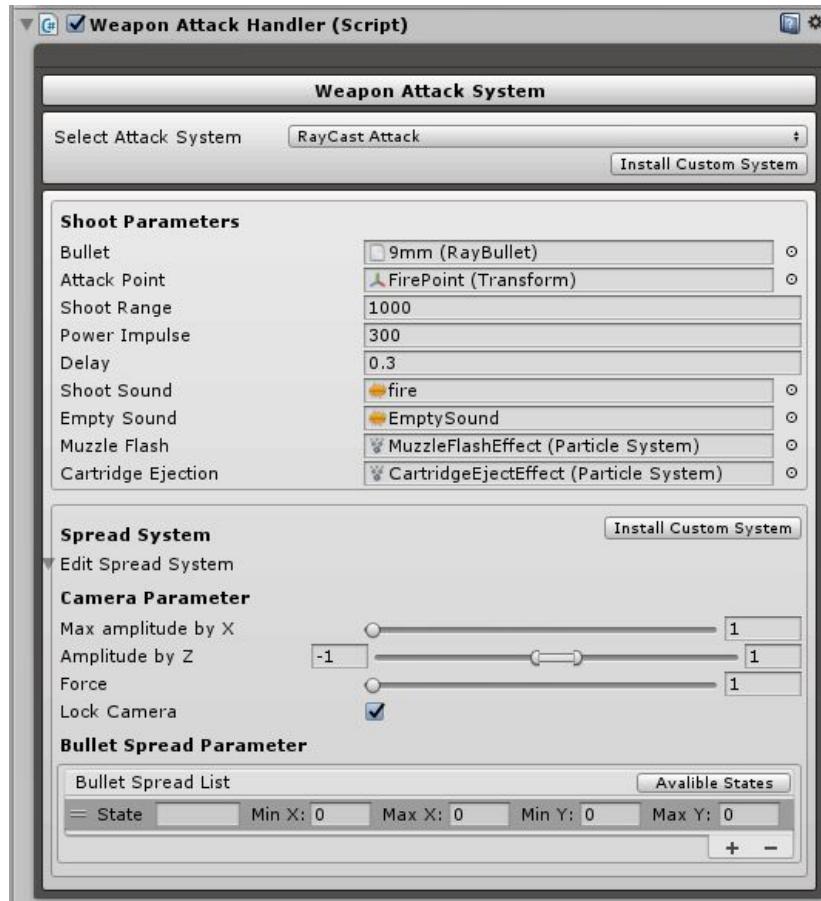
Drag&Drop in Weapon Identifier the Weapon  
you can change the Weapon by clicking on the button Load New

## 4.4 Weapon Animation Handler



Activate Sway - enable/disable sway system

## 4.5 Weapon Attack Handler



### 4.5.1 Shoot Parameter

Bullet - Weapon bullet([#4.8](#))

Attack Point - Shoot/Attack position

Shoot Range - Max shoot/attack distance

Power Impulse - Max impulse of the flight of the bullet/impact

Delay - Shoot/Attack speed

Shoot Sound:

Shoot Sound:

Muzzle Flash:

Cartridge Ejection:

### 4.5.1 Spread System

Max amplitude by X - The recoil force of the weapon on the X-axis

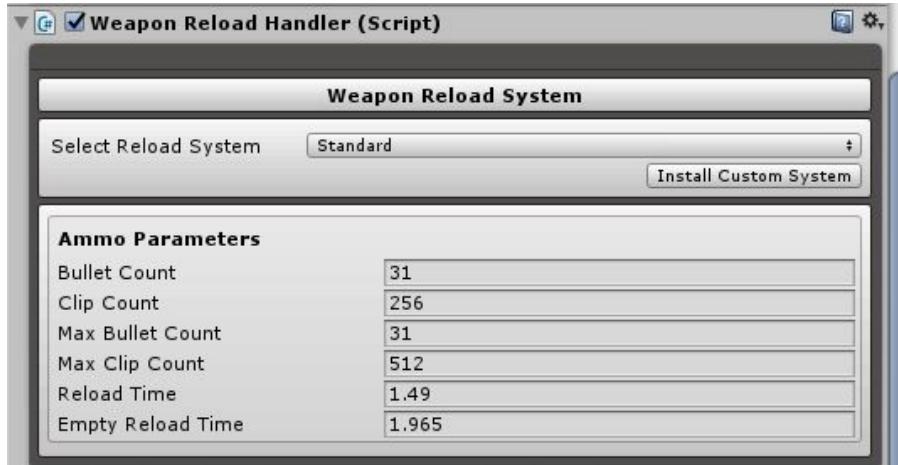
Amplitude by Z - The recoil force of the weapon on the Z-axis

Force - The recoil force of the weapon

Lock Camera:

## 4.6 Weapon Reload Handler

Weapon Reload Handler - the basic component responsible for the ammo in the weapon.



Bullet Count - Bullet count in the weapon at the moment

Clip Count - Clip count in the weapon at the moment

Max Bullet Count - Max Bullet count in the weapon

Max Clip Count - Max Clip count in the weapon

Reload Time - Reload time when clip is not empty

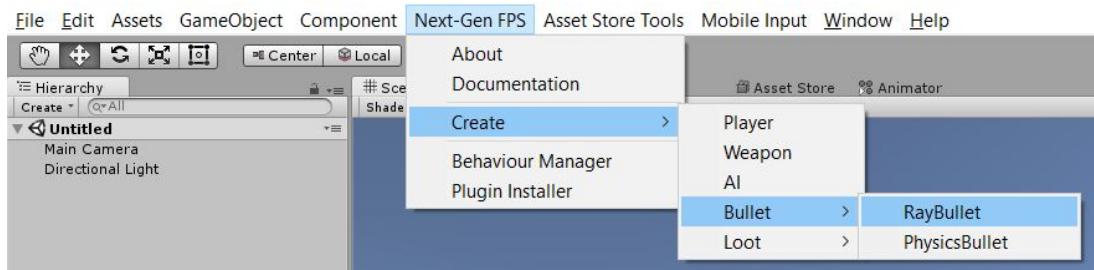
Empty Reload Time - Reload time when clip is empty

## 4.8 Ray Bullet

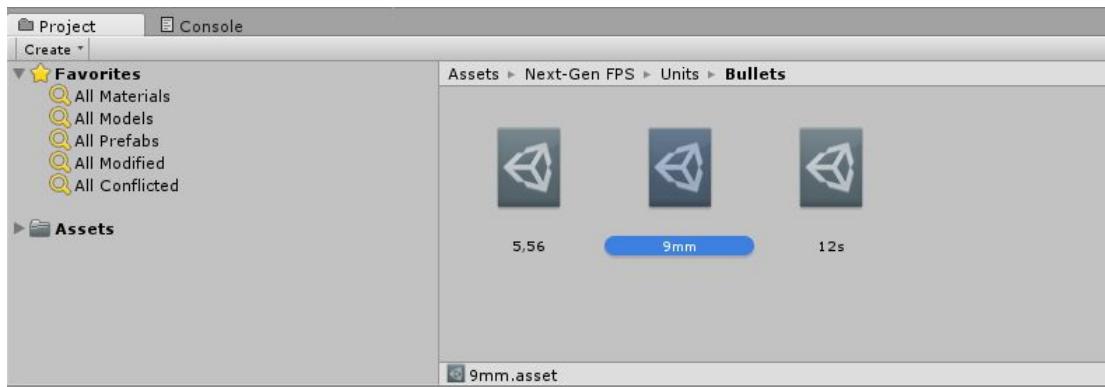
### 4.8.1 Create Ray Bullet

For the create ray bullet go to the tab:

Unreal FPS → Create → Ray Bullet

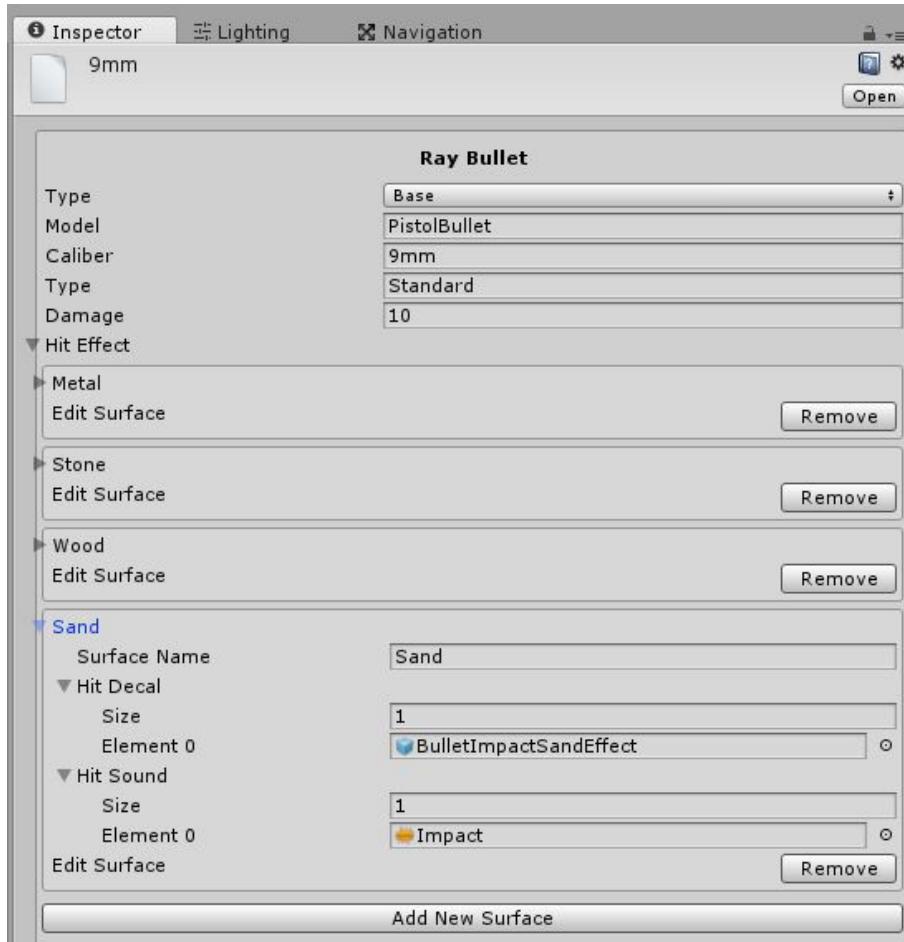


After this, a file must be created



#### 4.8.2 Setting-up Ray Bullet

Click on it to configure it



Type - Base/Shotgun

Hit Effect - Hit effect when bullet hit on collider, you can use different hit effect on different surface

For create new surface click "Add new surface", for remove specific surface click on "Remove"

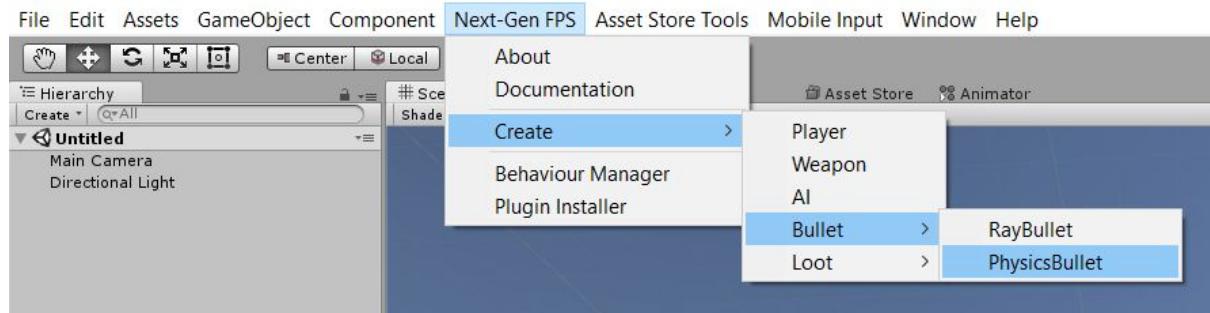
Surface Name must coincide with the physical material

## 4.9 Physics Bullet

### 4.9.1 Create Physics Bullet from the tab

For the create physics bullet go to the tab:

Unreal FPS → Create → Physics Bullet



### 4.9.2 Create Physics Bullet from the scratch

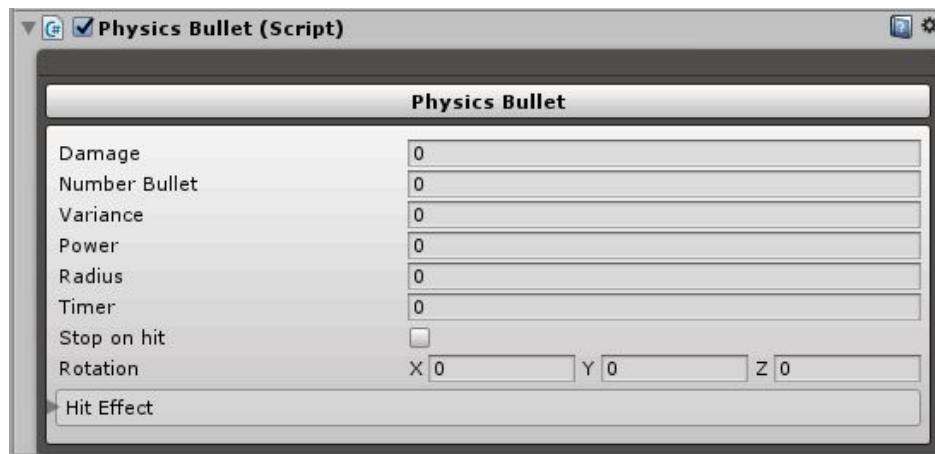


Drag on the scene bullet model and name it for example Bullet

Add onto Bullet Collider

Drag Physics Bullet script onto Bullet GameObject

### 4.9.3 Physics Bullet



Number bullet and Variance used only for shotgun

Power - The force of the bullet

Radius - radius of injury

Timer - time before the bullet is removed from the scene after the shot

Stop on hit - bullet will stopped when hit on collider

Rotation - Bullet rotation when shoot

Hit Effect - Hit effect when bullet hit on collider, you can use different hit effect on different surface

## 4.7 Pickable Weapon

### 4.7.2 Create Pickable Weapon



1. Drag&Drop weapon model on the scene
2. Set tag - Weapon
3. Add Weapon Identifier Component
4. Set-up Weapon Identifier ([#4.3](#))
5. Add Box Collider Component

Optional:

1. Add Marking Component
2. Set-up Marking Component as Grab
3. Add Rigidbody

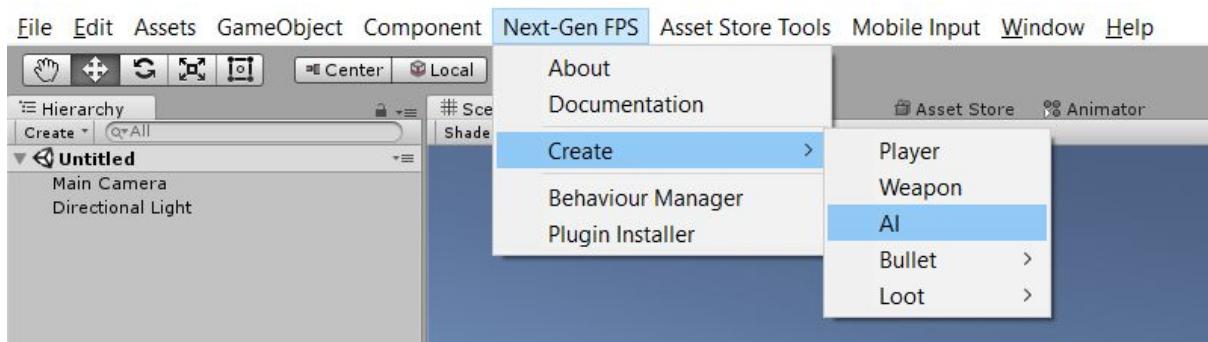
# 5. AI

## 5.1 Create AI for the Tab

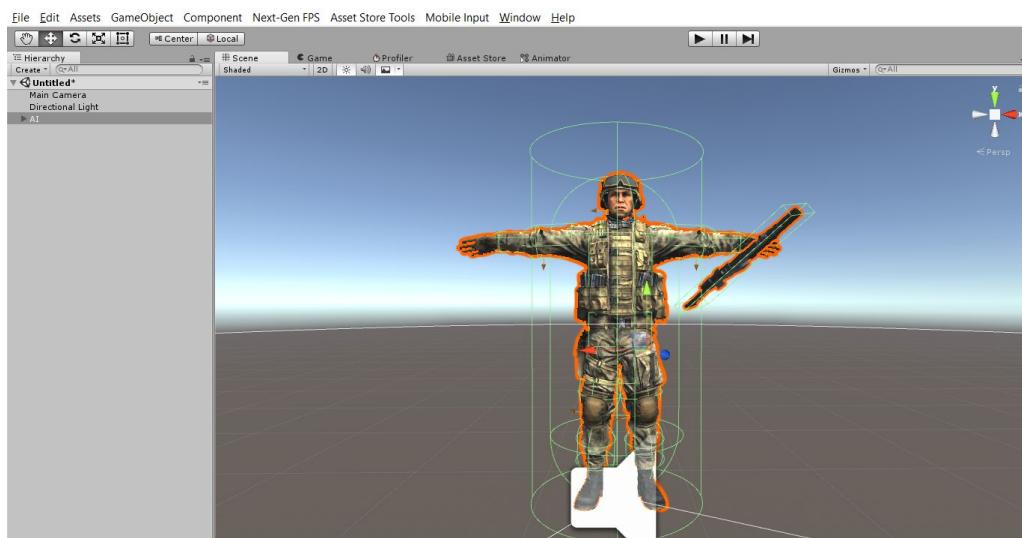
For the create AI go to the tab:

Unreal FPS → Create → AI

After AI Instantiate on the scene



## 5.2 Create AI from scratch



1. Create empty GameObject and name it for example AI
2. Add in AI, AI model
3. For AI set Layer - “AI” and say “Yes, change children”
4. You can specify any tag you want
5. Add Animator Component onto GameObject AI
6. Drag the AIHandler script onto AI GameObject and set-up AIHandler
7. Drag the AIHealth Handler script onto AI GameObject and set-up AIHealth

8. Drag the AIFieldOfView Handler script onto AI GameObject and set-up AIFieldOfView
9. Add NavMeshAgent Component onto GameObject AI
10. Add Capsule Collider Component onto GameObject AI
11. Add Audio Source Component onto GameObject AI

## 5.3 AI Handler

AI Handler - the base component responsible for conducting AI.



Select an AI system from the available systems

## 5.4 AIHealth



Health - Health point at the moment

Max Health - Max health point

## 5.5 AIFieldOfView

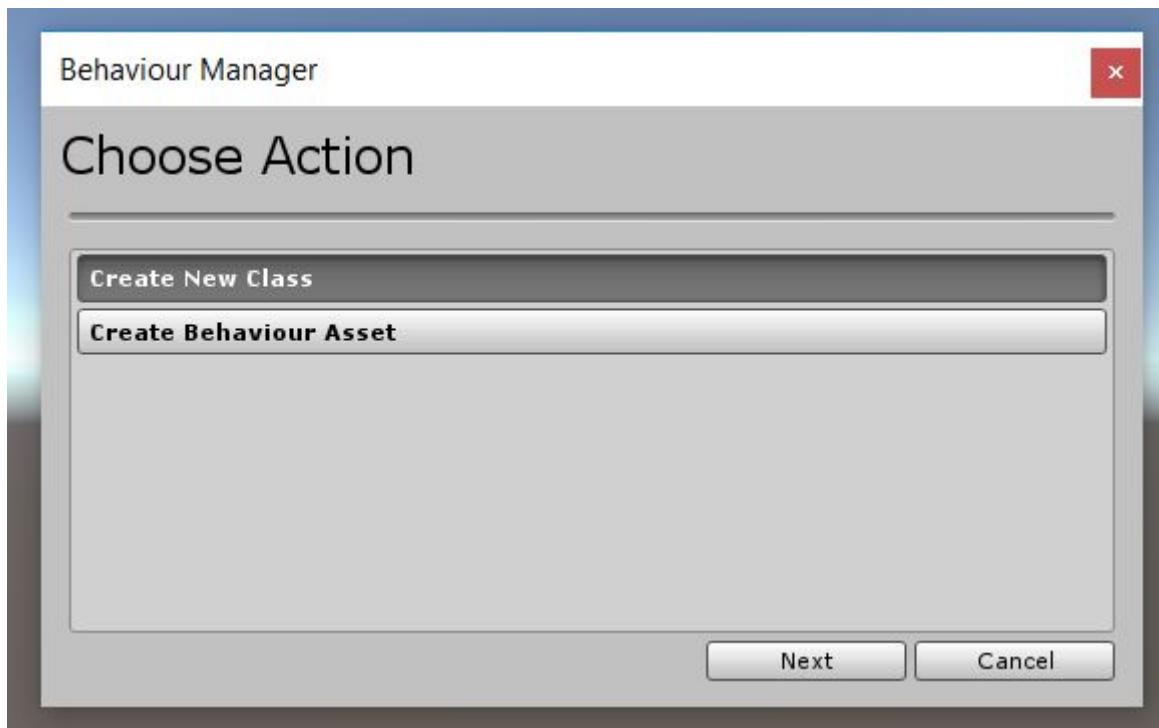


You can select more than one Target Masks, in Target mask you point out on current enemies of this AI

In Obstacle Mask you point out on objects through which the AI can't see

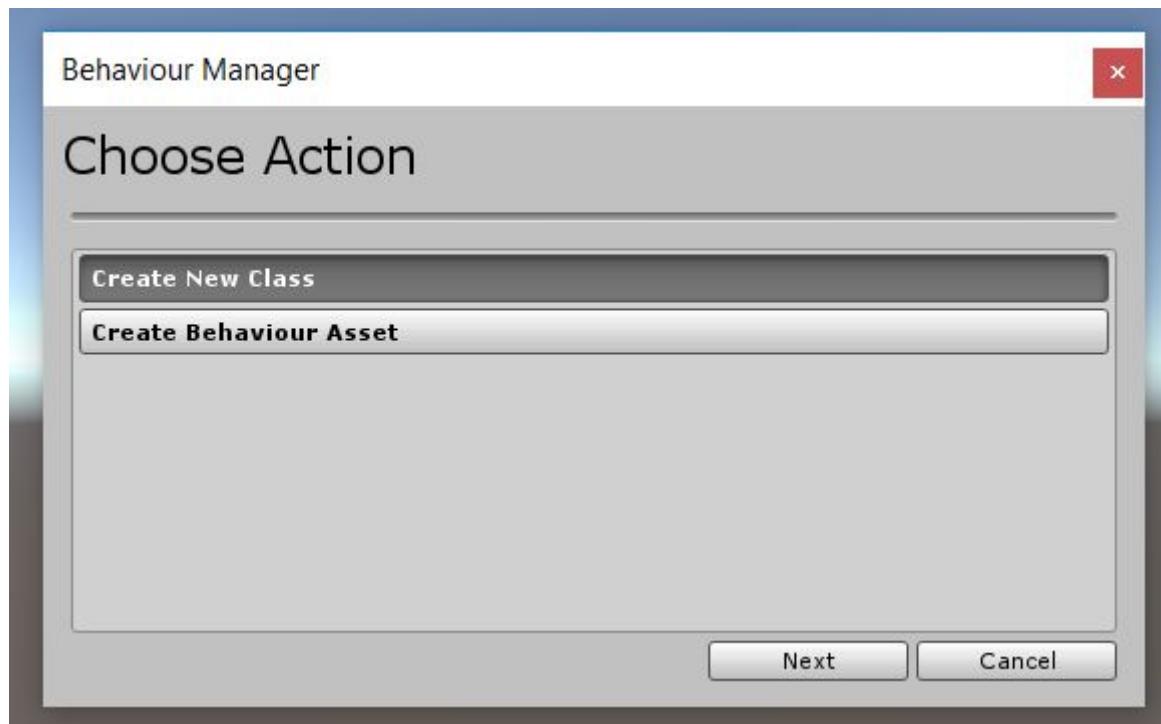
# 6. Behaviour Manager

Behaviour Manager is a powerful manager for create classes and Behaviour assets for Unreal FPS

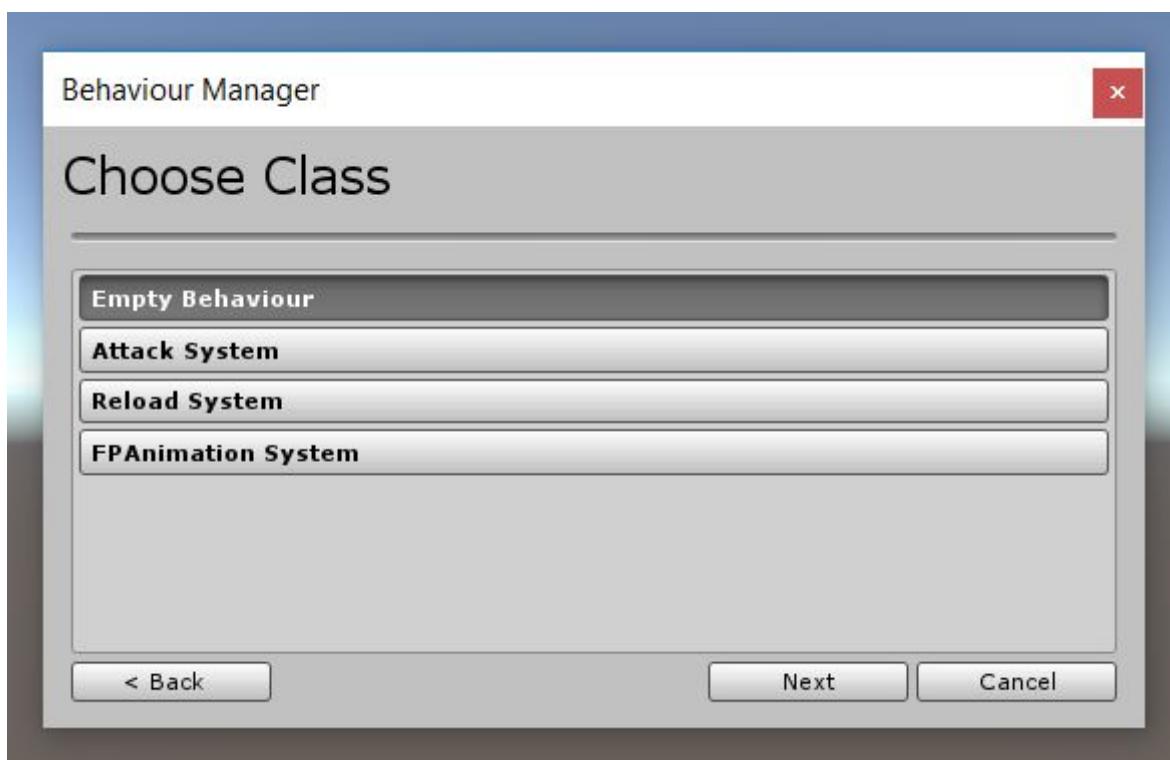


## 6.1 Create Class

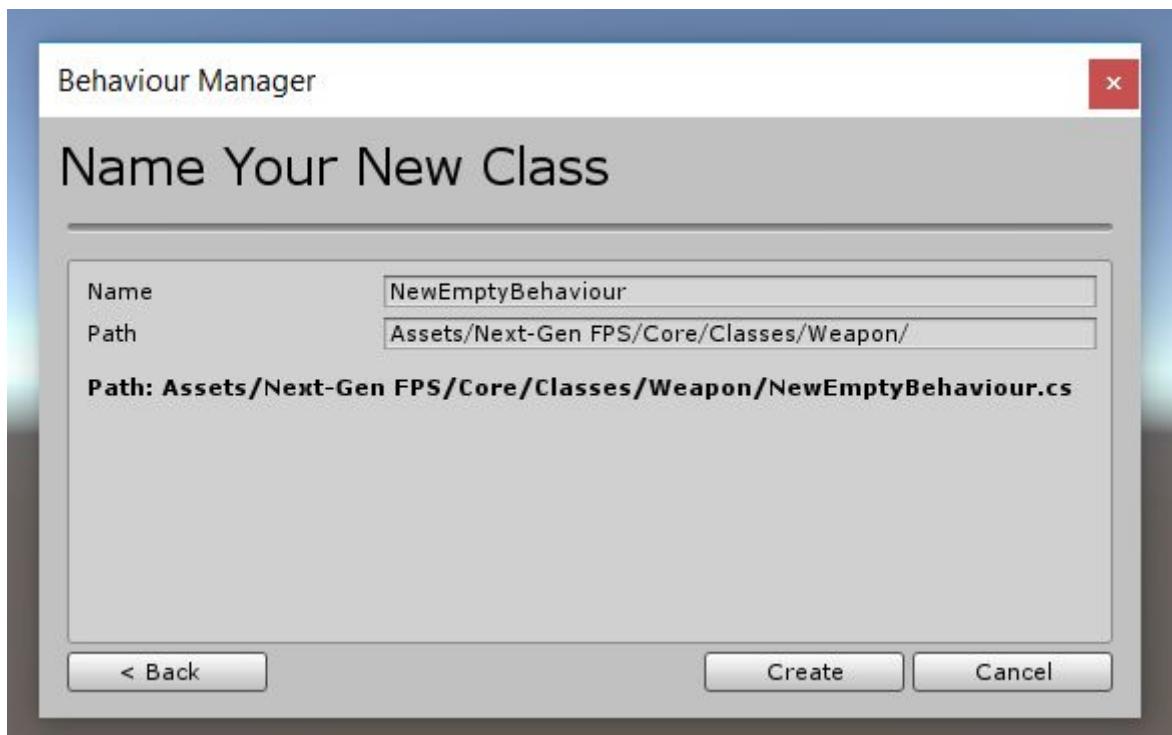
For create custom system select “Create New Class”



Choose the class

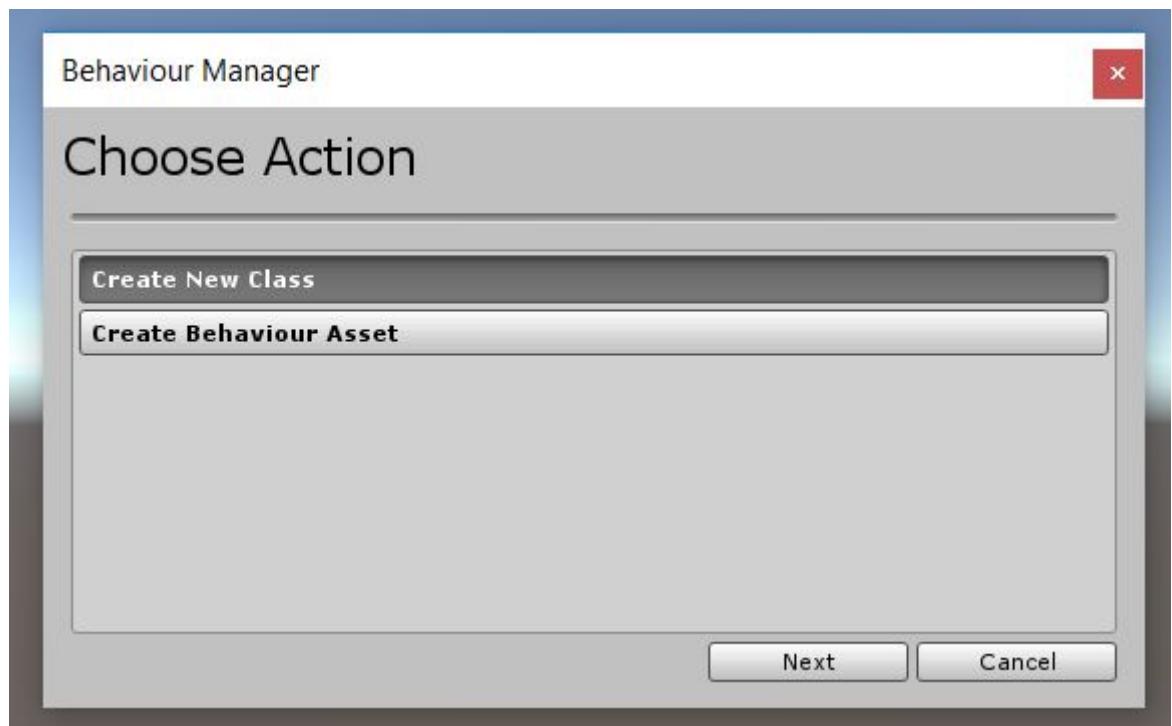


Set name and path for the new class



## 6.2 Create Behaviour Asset

For create custom system select “Create Behaviour Asset”



Choose Behaviour for create



You can set-up new Behaviour immediately in Behaviour Manager or later from the Inspector

Set name and path for the new Behaviour Asset



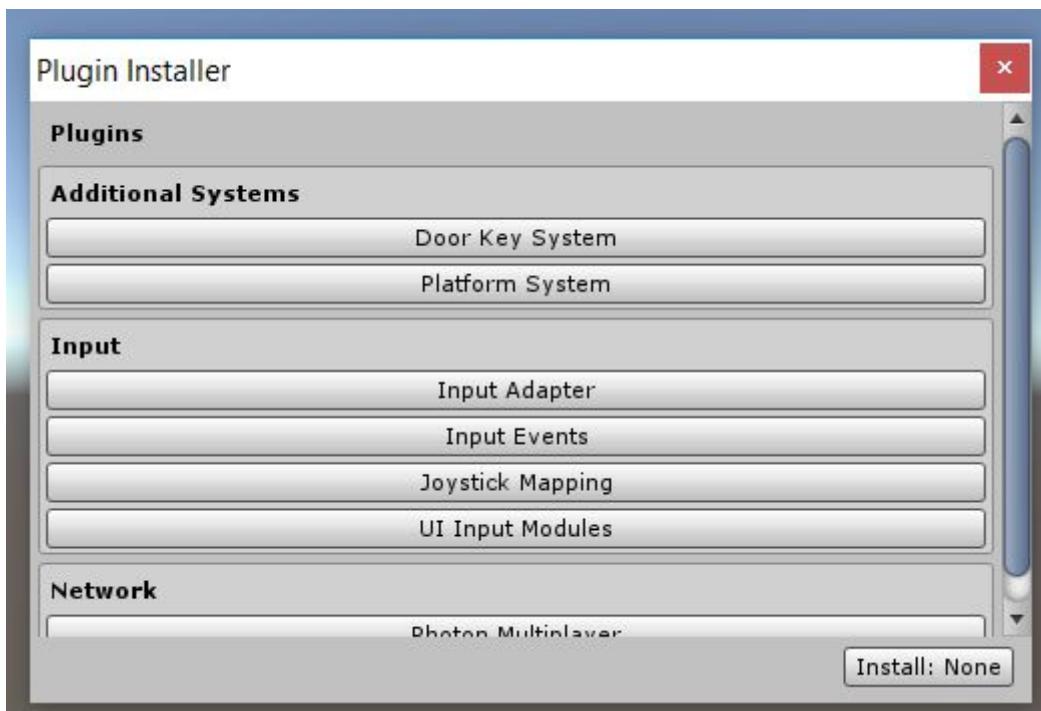
# 7. Plugins

## 7.1 Install Plugins

For install plugins go to the tab:

Unreal FPS → Plugin Installer

Select plugin and press install button



## 7.2 Add Plugins

All plugins are located in the folder: Unreal FPS/Resources/Plugins



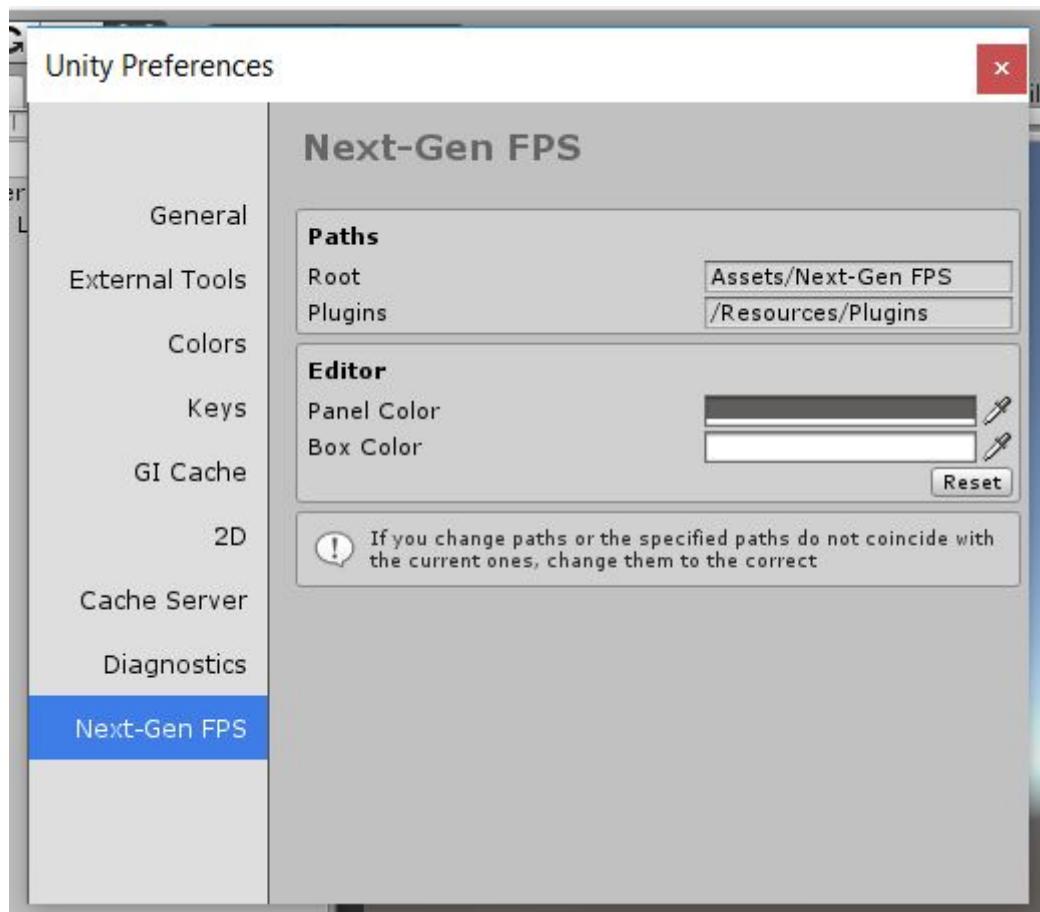
All the folders that are in the "Plugins" folder is a groups. For create new group create new folder.

For the add new plugin put you plugin in the other folder (Group). All plugins must have an extension (.unitypackage)

## 7.3 Plugin folder

For the change plugin folder path go to the tab:  
Edit → Preferences → Unreal FPS

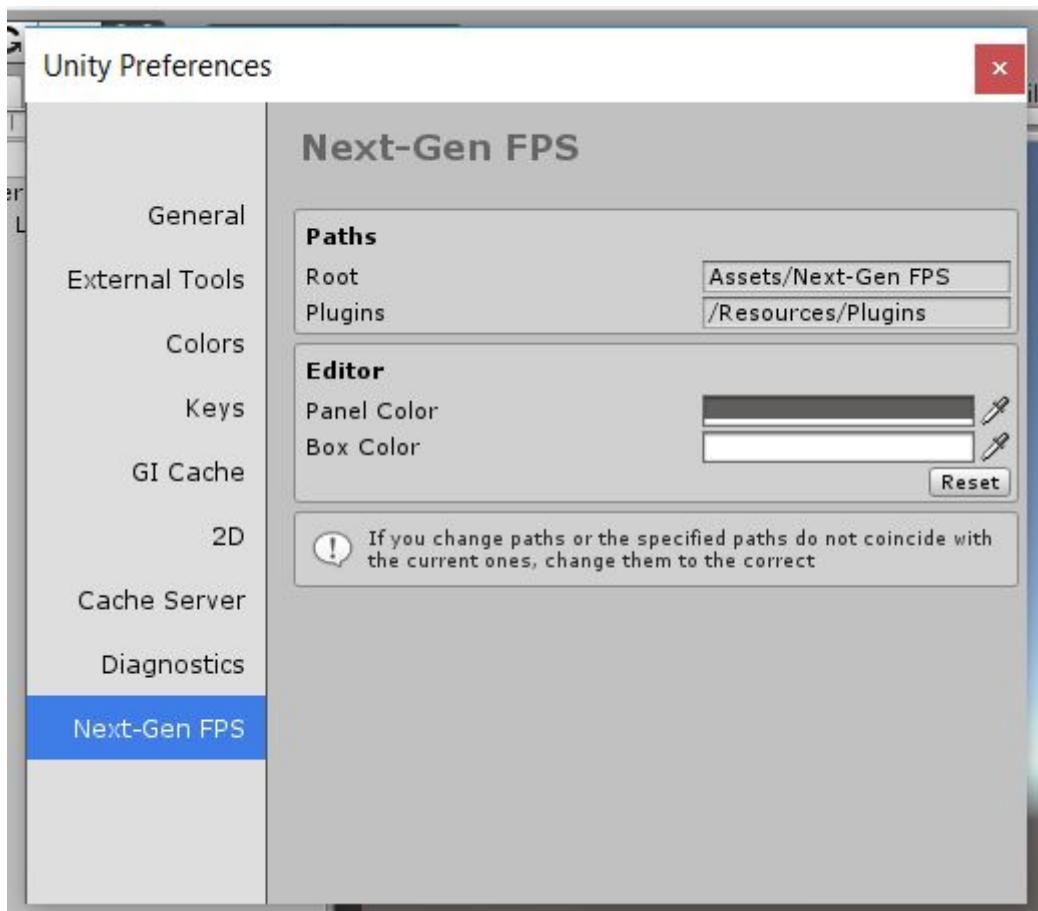
And set new path in the Plugins field



# 8. Preferences

In order to configure the editor Unreal FPS more finely go to the tab:

Edit → Preferences → Unreal FPS



## Paths

Root - Unreal FPS root folder path

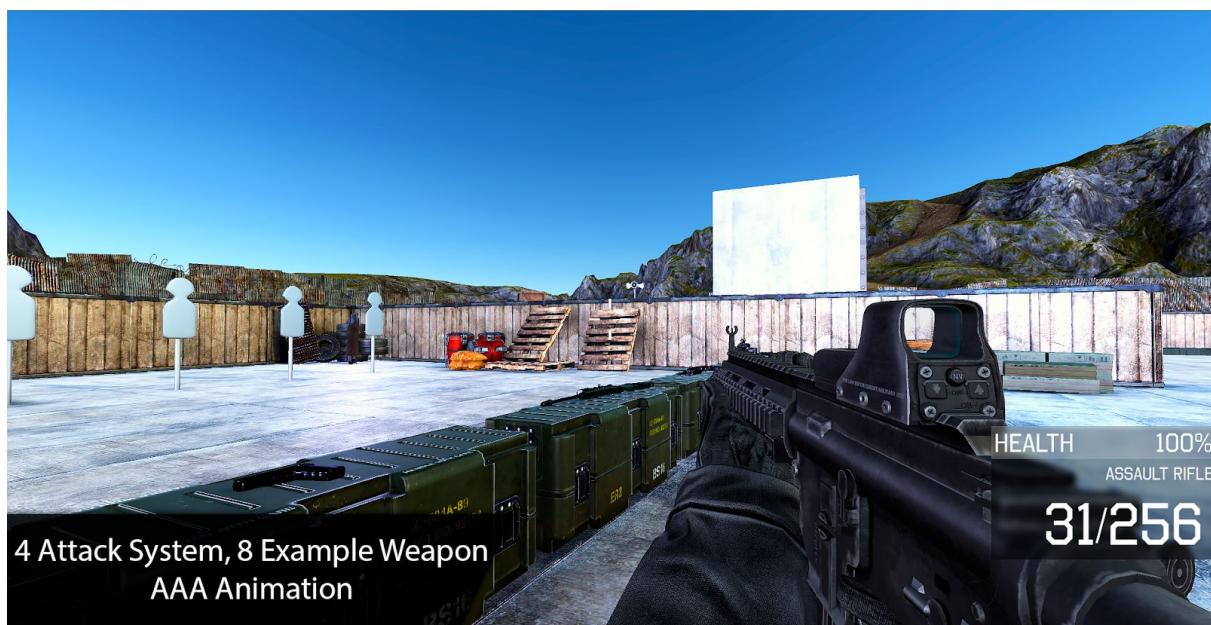
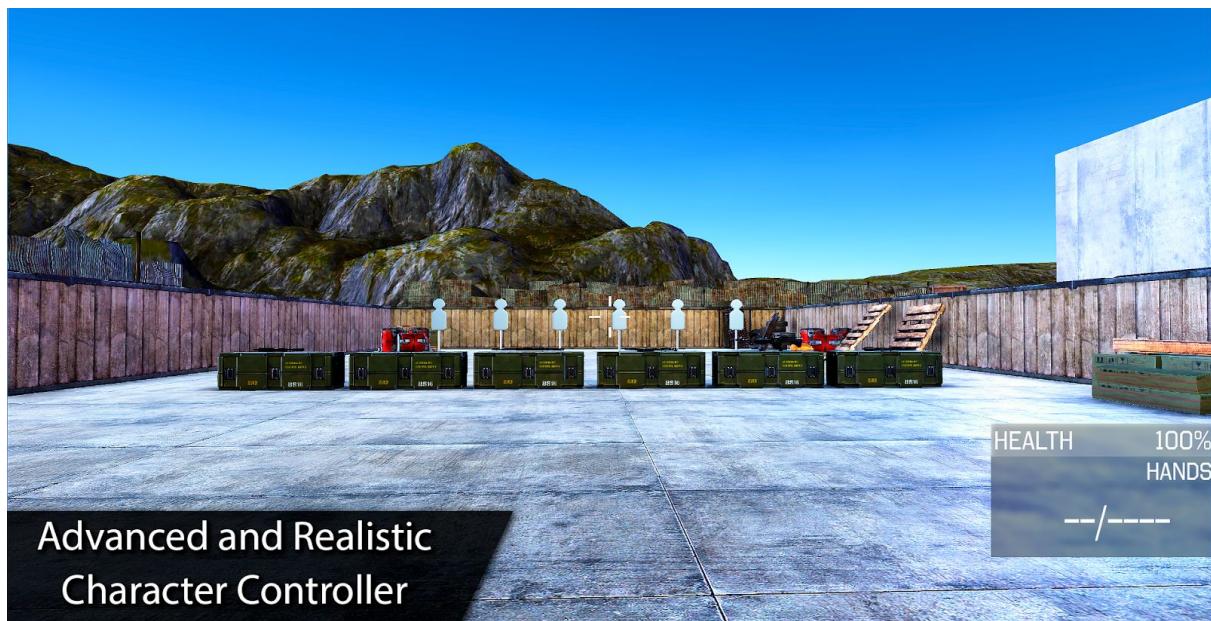
Plugins - Plugins folder path

## Editor

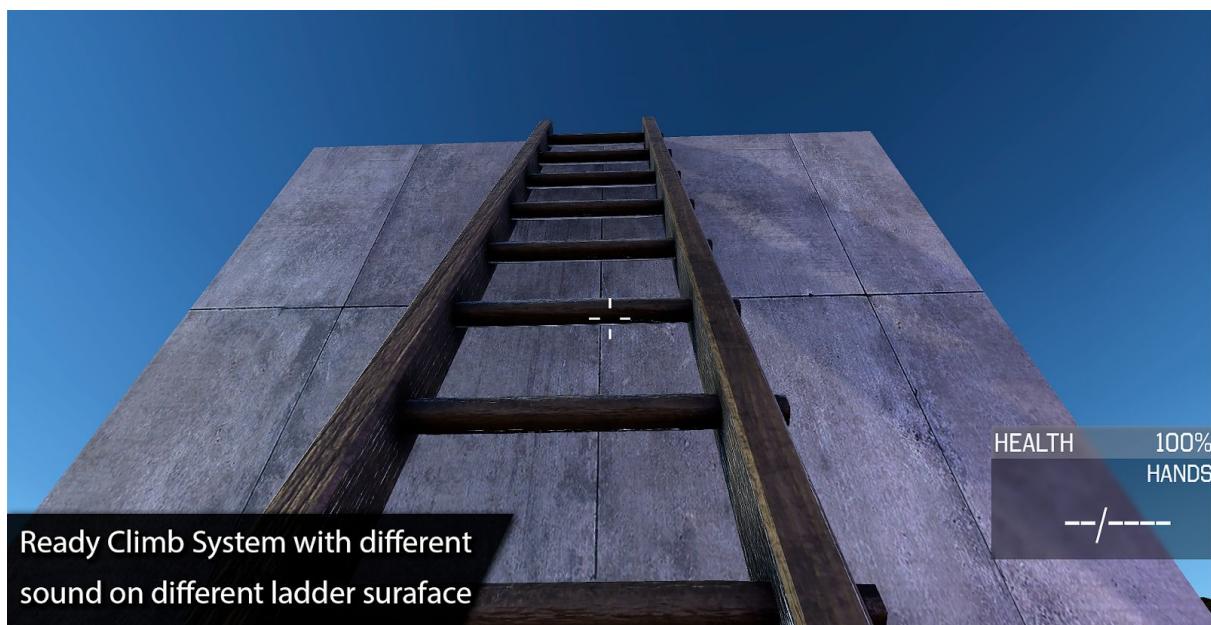
Panel Color - Background panel color of the Unreal FPS component in the Inspector GUI

Box Color - Box panels color of the Unreal FPS component in the Inspector GUI

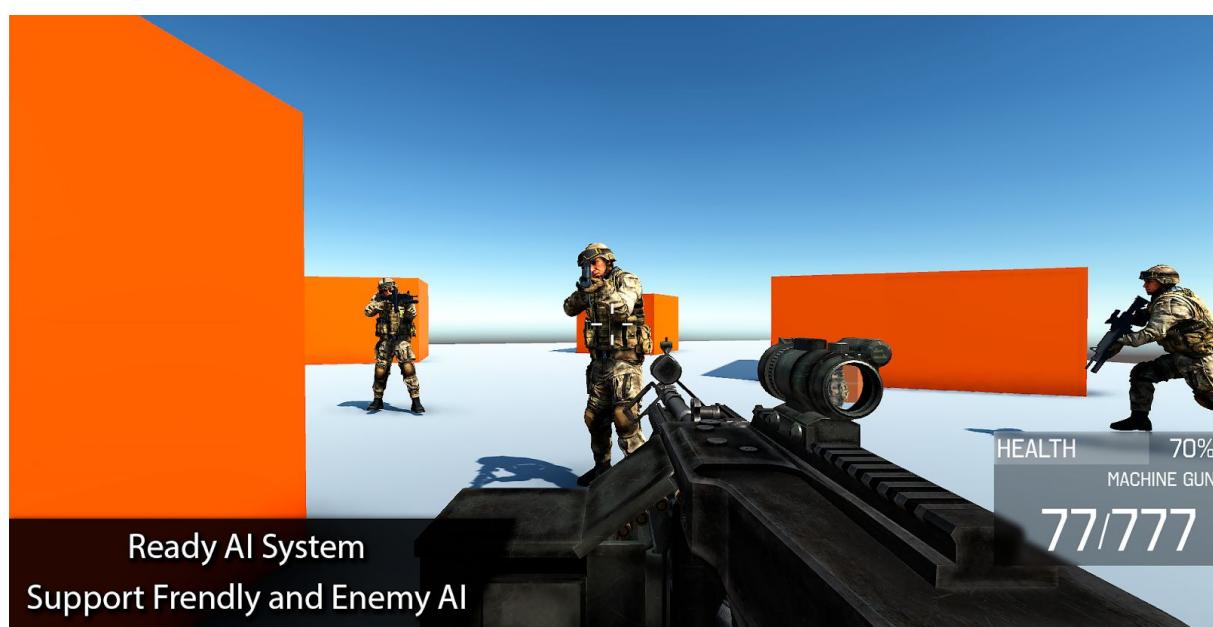
Reset - Reset colors on default value



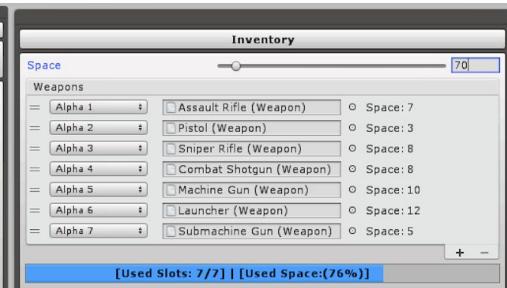
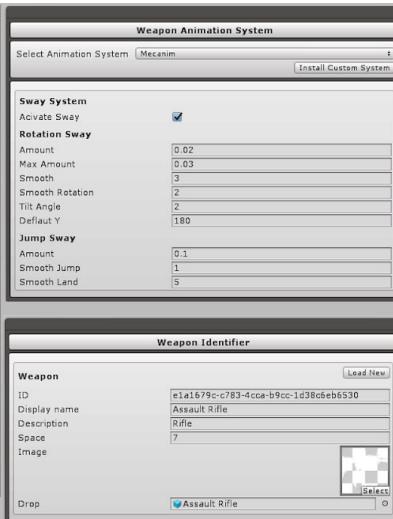
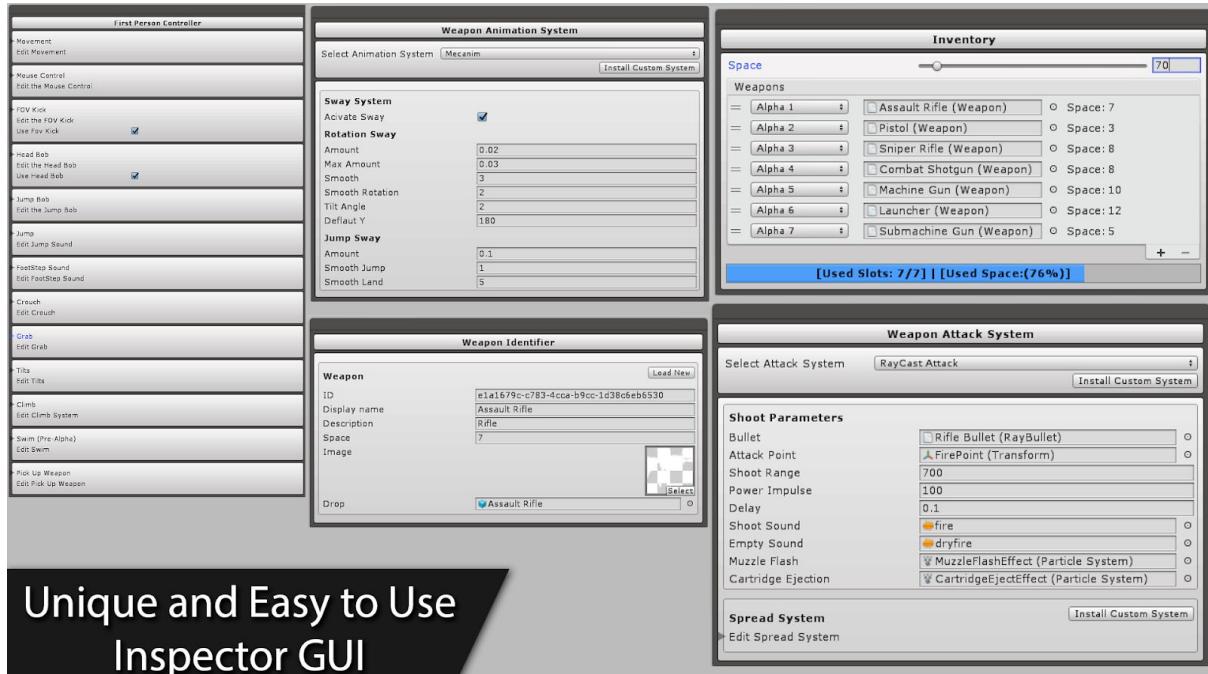












## Unique and Easy to Use

## Inspector GUI

