

# Weapon Drop Script 1.0

**Game Engine: Unity**

Kevin Iglesias - [www.keviniglesias.com](http://www.keviniglesias.com)

Contact Support:  
[support@keviniglesias.com](mailto:support@keviniglesias.com)

[Online Documentation](#)

## **FOLDER CONTENTS**

### **Animations.**

A total of 4 example animations compatible with Mecanim (Death, Giving up, Hands up loop and Aiming with rifle loop).

### **Prefabs.**

Example of uses. These assets are prepared with the scripts already configured for taking as reference for your game characters.

### **Scripts.**

Inside this folder you will see 2 scripts. One is the core of WeaponDropScript and is needed to be added as a component to your game character. The other is optional and is a State Machine Behaviour for the Animator Controller. Further details and configuration are explained below.

### **Source Models**

The FBX models and the textures used by the example character and the weapons.

### **Demo scene**

A demo scene containing all the prefabs.

### **Documentation**

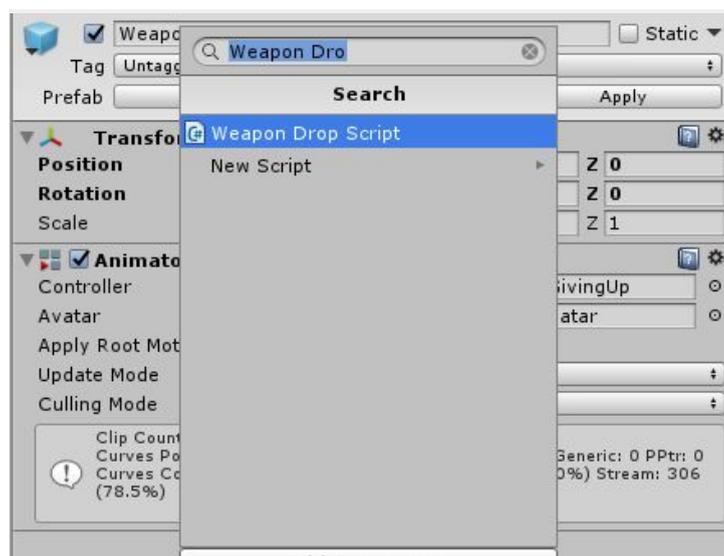
This documentation in PDF file format.

## HOW TO USE

First of all, you need to add the main script to your game character: the WeaponDropScript (MonoBehaviour component).

### WeaponDropScript (MonoBehaviour)

The WeaponDropScript must be added as a component to your game character where the Animator component is.



Adding the WeaponDropScript from the Inspector.



Adding and configuring one weapon entry.

Click on **[+] Add New Weapon Drop** for adding a new weapon entry. You will need one entry for each weapon you want to be dropped.

**[+] Remove Weapon Drop** will remove the selected weapon entry. You can use **Drop weapon** button for debugging and trying the best configuration that fits your drop.

## Drop fields

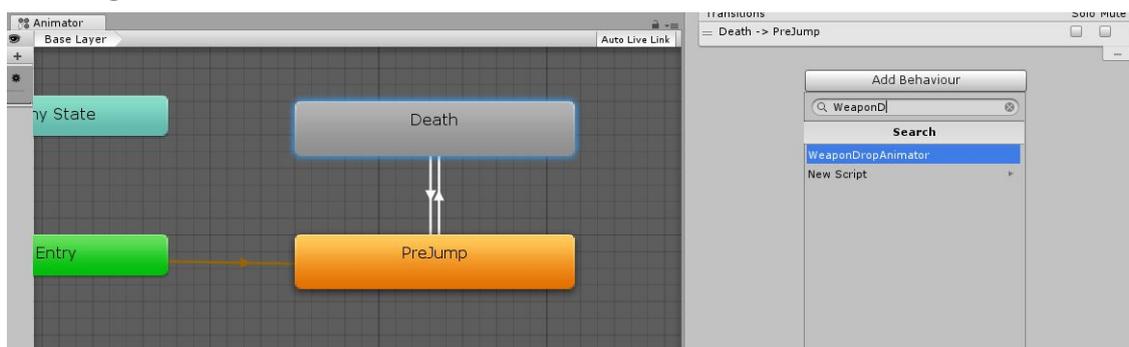
- **Weapon.** This field is for the weapon `GameObject` that will be dropped. Drag here your weapon.
- **Initial Jump.** This will make a jump as soon as the weapon is dropped. It can give a more cartoony effect in some cases or give a parabola effect.
- **Use Rotation.** This will make the weapon or item rotate when falling. Value can be negative to make it rotate to the other way.

To actually drop the weapon without using the **Drop weapon** button you can call '**Drop(int weaponId)**' from other script, use Animation events or use the State Machine Behaviour included in the package.

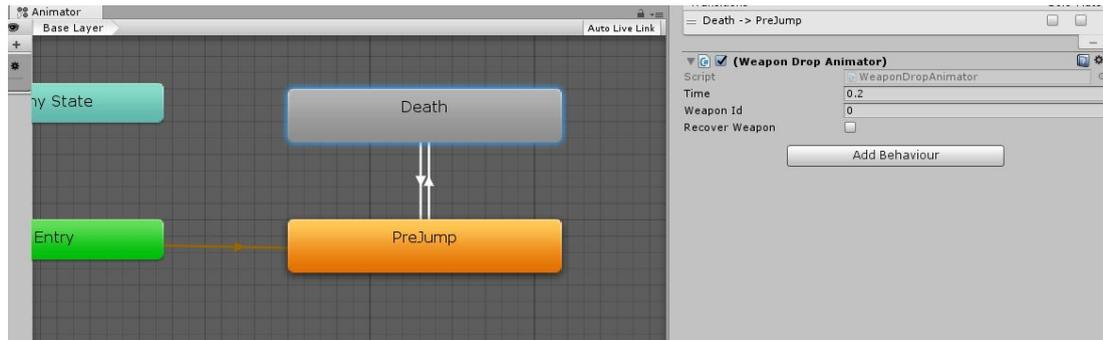
## WeaponDropAnimator (State Machine Behaviour)

If you opt for using the State Machine Behaviour you will need to add it to the state in the Animator where the animation that drops the weapon is placed.

See images below:



**Adding the WeaponDropAnimator to 'Death' state.**



Configuring values: Time, Weapon ID and Recover Weapon.

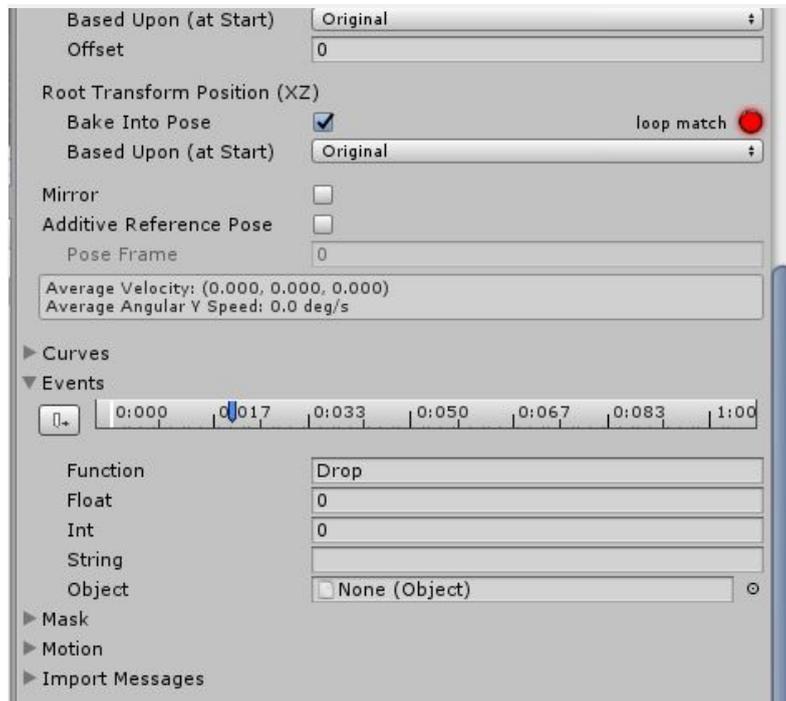
## Weapon Drop Animator fields

- **Time.** This is the point of the animation that will the drop or recover occur. This is percentage (%) value, 0.5 means in the middle of the animation. Adjust this value if the character drops the weapon too early or too late.
- **Weapon Id.** The entry of the main script component attached to the character. This will dictate which weapon will be dropped. Use 0 if you only have one weapon to drop.
- **Recover.** Checking this field will make the script to call '**Recover(int weaponId)**' instead of '**Drop(int weaponId)**'. That will make the weapon return to its original position.

## Alternative way of dropping weapons: Animations Events

If you don't want to use your own script or adding State Machine Behaviours, you can use Animation Events for triggering drops. However, it is recommended to trigger drop using State Machine Behaviour or using a custom script since Animation Events can be skipped if they are out of the transition between 2 animations.

For Animation Events: Find your animation file and go to the animation Import Settings using the Inspector, select your animation and add an event in the Events section. See image below:



Place the event in the point of the animation that you want the drop to occur. Then write either 'Drop' or 'Recover' in the **Function** field depending on if you want to drop or recover an already dropped weapon.

Make sure the **Int** field value is the same as the entry of the desired weapon to be dropped or recovered based on the WeaponDropScript component of your character. Leave at 0 if your character only have one weapon to drop.

### **Custom script calls**

If you want to trigger your weapon drops with your own script there are 2 methods that you can call:

```
public void Drop(int weaponId);  
public void Recover(int weaponId);
```

Where 'weaponId' is the number of the entry of the weapon you want to drop in the WeaponDropScript attached to your game character.

## **CONTACT SUPPORT**

For support, questions or suggestions regarding this product send me an email to:

[support@keviniglesias.com](mailto:support@keviniglesias.com)

Kevin Iglesias - [www.keviniglesias.com](http://www.keviniglesias.com)

- [More Unity Assets](#)
- [Facebook](#)
- [Twitter](#)
- [YouTube Channel](#)