

3d Menu System (a Unity3d Asset)

by Justin Garza

Introduction

Hello Thanks for purchasing this product.

If you need to contact me for any reason you can find my contact information [here](#).

Description

This product allows users to create a basic menu system without the need for Unity Pro, or expensive plugins like NGUI.

Features

Add effects to display when an option is in focus

Add easily assign action to options

Easily add Meshes, and Text to each option

Automatically save chosen options

Table of Contents

Types of Menus	Page2
How to Set Up a GridMenu	Page3
How to Set Up a RotationalMenu	Page6
FocusEffects	Page8
OptionAction	Page10

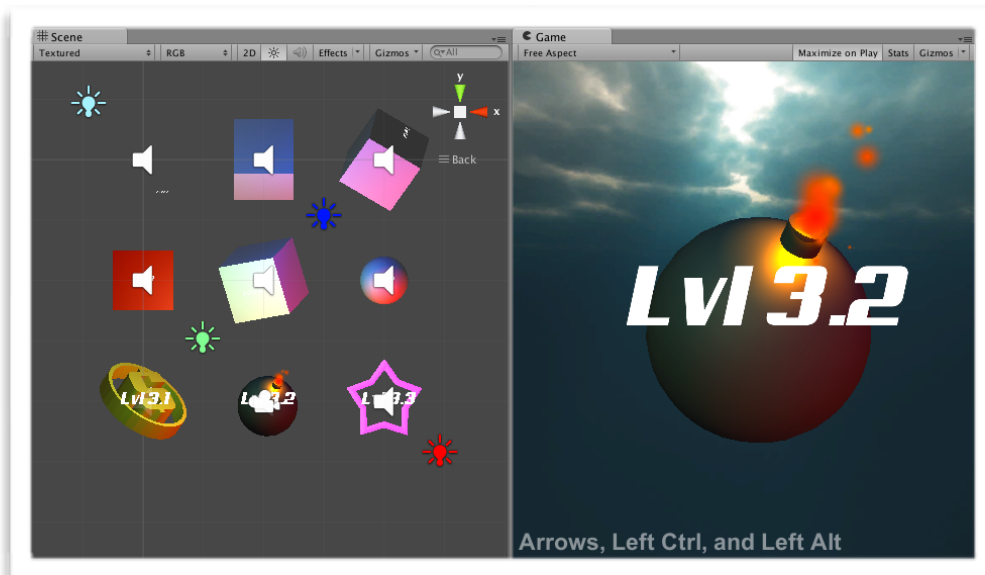
Types of Menu

In this asset there are two types of Menu Systems.

Grid Menu

In this Menu the options are set up in a grid.

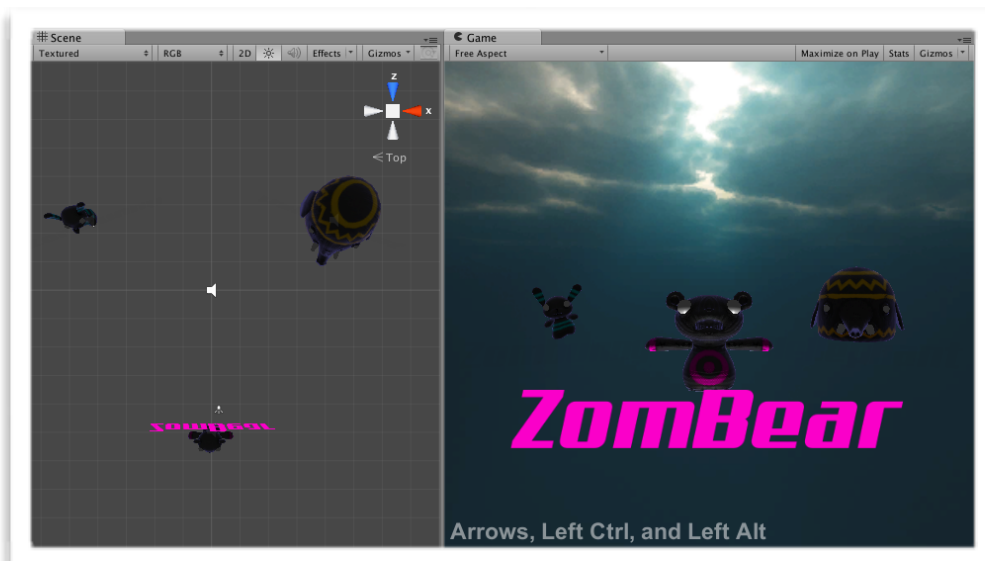
An example of this is the Title Menu, or the Level Picker.



Rotational Menu

In this Menu Options rotate around a center.

An example of this would be the CharacterPickerExample.



How to set up a Grid Menu

You'll need:

1. Main Camera with the ControlCameraScript.cs attached to it.
2. GenericGridOption prefab (in the prefabs folder).

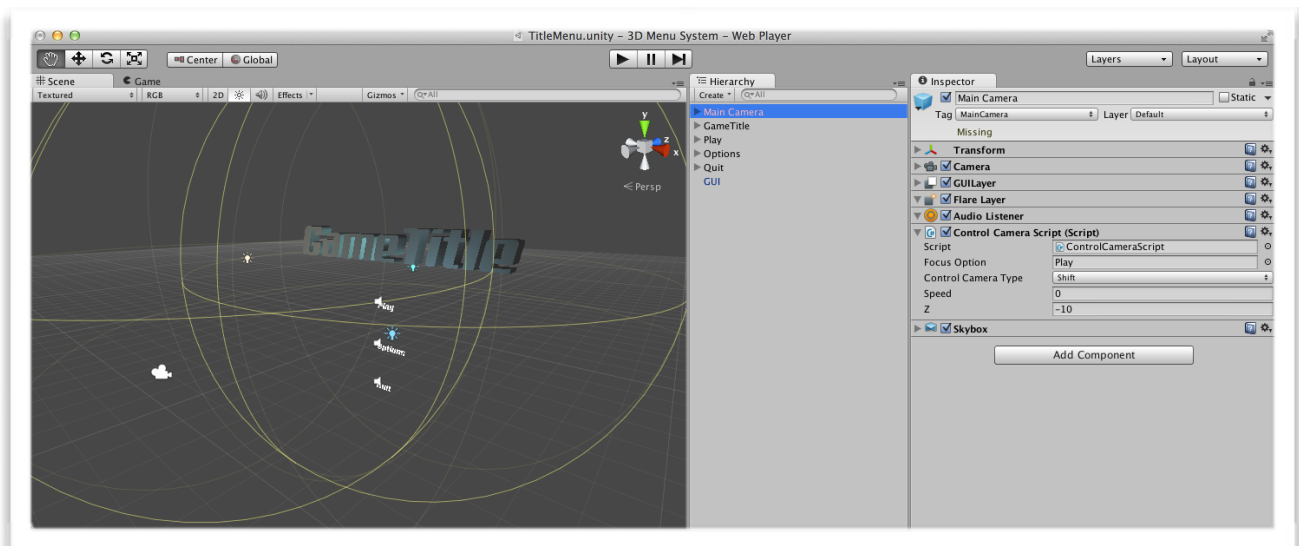
Steps

1. Attach the ControlCameraScript.cs to your Main Camera.
2. Drag the GenericGridOption.prefab into your scene.
 1. Change the Name of the GameObject, and Text in the TextMesh.
3. Add children to the Option to customize it (Optional).
4. Add FocusEffects, and OptionActions
 1. Assign appropriate values as you go.
5. Make copies of your Option until you have all the ones you want.
6. Move them on the X and/or Y Axis
7. Adjust their FocusEffect, and OptionActions as you see fit.
8. Assign your default option to the Focus Option in the ControlCameraScript.cs on your Camera

Note:

Disable or remove colliders on all the children of the Options.

At the end it should look something like this...



a little about the ControlCameraScript.cs

Focus Option:

The Option that the Camera is Focused on.

Control Camera Type:

Pick weather you want the Camera to Shift to a Option, or Rotate to it.

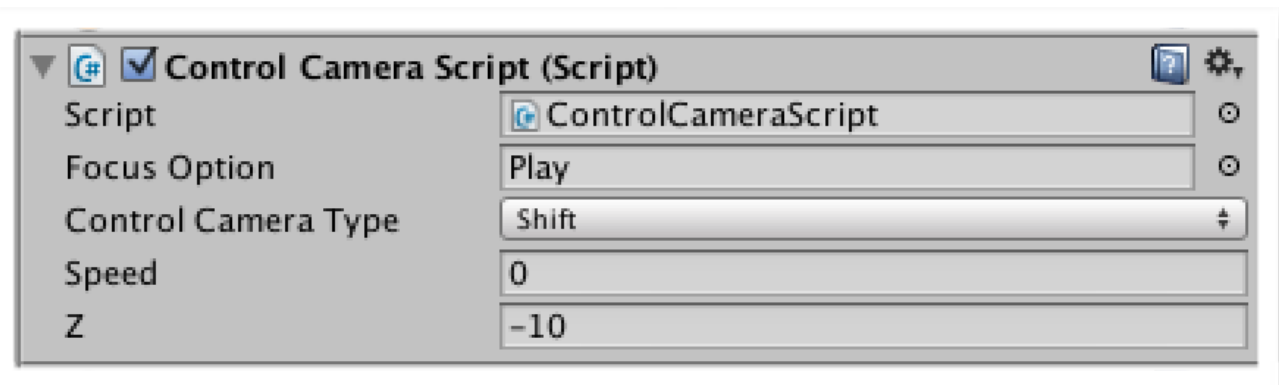
Speed:

The Speed that the camera moves

If you don't want the camera to move set it to 0.

Z:

the Distance the camera should be from the Z axis.



a little about the Grid_OptionScript.cs

Focus:

Weather or not this item is in Focus

Up:

The Option above this one

Down:

The Option below this one

Left:

The Option to the left this one

Right:

The Option to the right this one

AutoDetect:

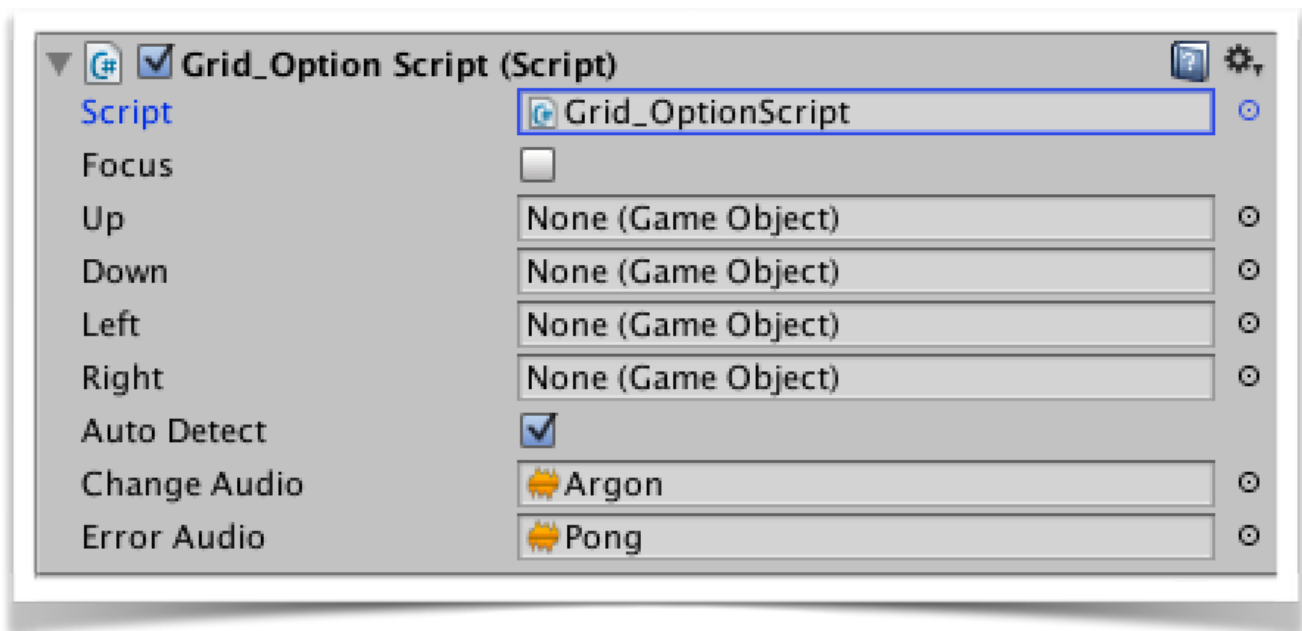
Weather or not the script should automatically assign the Up,Down,Left,Right values

Change Audio:

the AudioClip that will play when the focus changes to another Option.

Error Audio:

the AudioClip that will play when the focus can't change.



How to set up a Rotational Menu

You'll need:

1. RotationalMenu prefab (in the prefabs folder)
2. SetFocus prefab (in the prefabs folder)

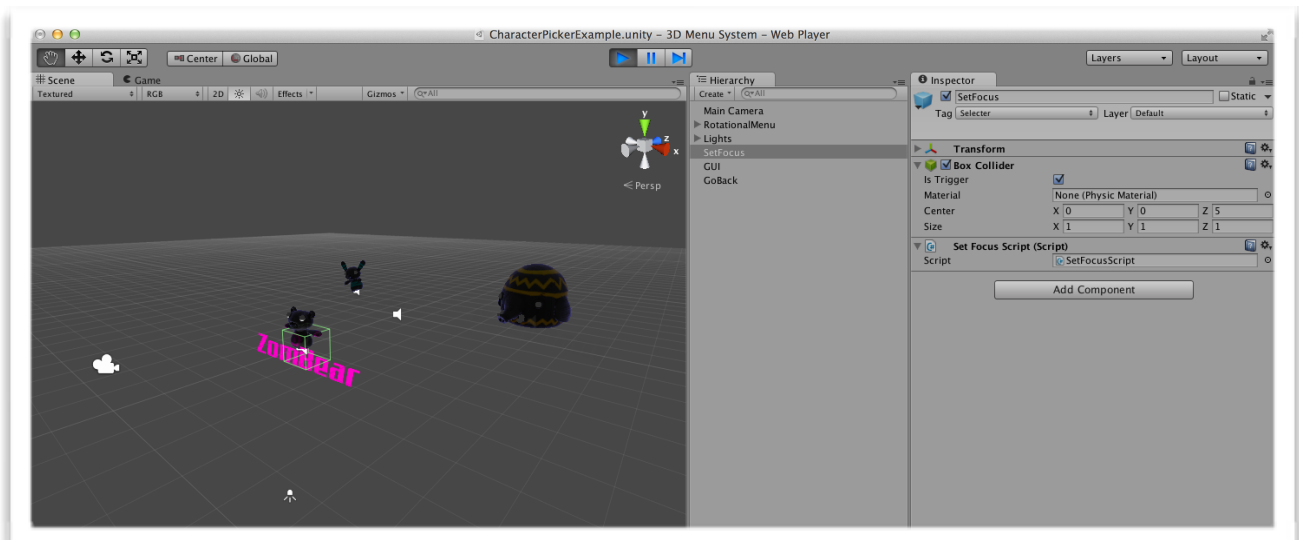
Steps

1. Drag the RotationalMenu.Prefab into your scene.
2. Adjust the settings until you get it in the position you want it.
 1. play the scene a few times to see it "live"
 2. make copies of the Option1 GameObject to see how it looks with more options.
 3. Change the Name of the GameObject, and Text in the TextMesh.
3. Add children to the Options to customize it (Optional).
4. Add FocusEffects, and OptionActions
 1. Assign appropriate values as you go.
5. Make copies of your Option until you have all the ones you want.
 1. Make sure all the options are children of the RotationalMenu GameObject.
6. Adjust their FocusEffect, and OptionActions as you see fit.
7. Drag and drop the SetFocus.Prefab into your scene.
 1. Move this GameObject to the location where the focused option should be.

Note:

Disable or remove colliders on all the children of the Options.

At the end it should look something like this...



a little about the RotationalMenuScript.cs

Speed:

The Speed at which objects move to the correct location

Radius:

The Circle size, from the center to a side.

Continuously Refresh:

Whether or not the Menu should continuously update each location.

This should be set to true ...unless you want to save some battery power.

Rotational Index:

determines what position to rotate the RotationalMenu to

Axis:

The Input Axis you want to use to rotate the Menu.

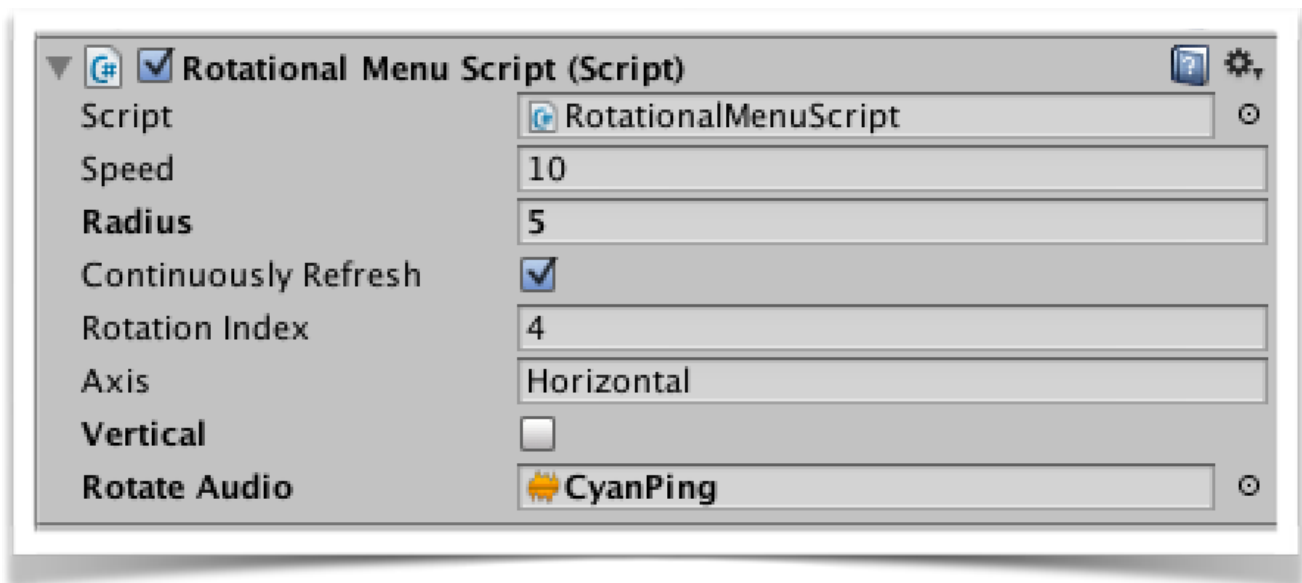
(Edit -> Project Settings -> Input)

Vertical:

Whether the Rotational Menu should be Vertical or Horizontal.

Rotate Audio;

AudioClip that will play when rotated.



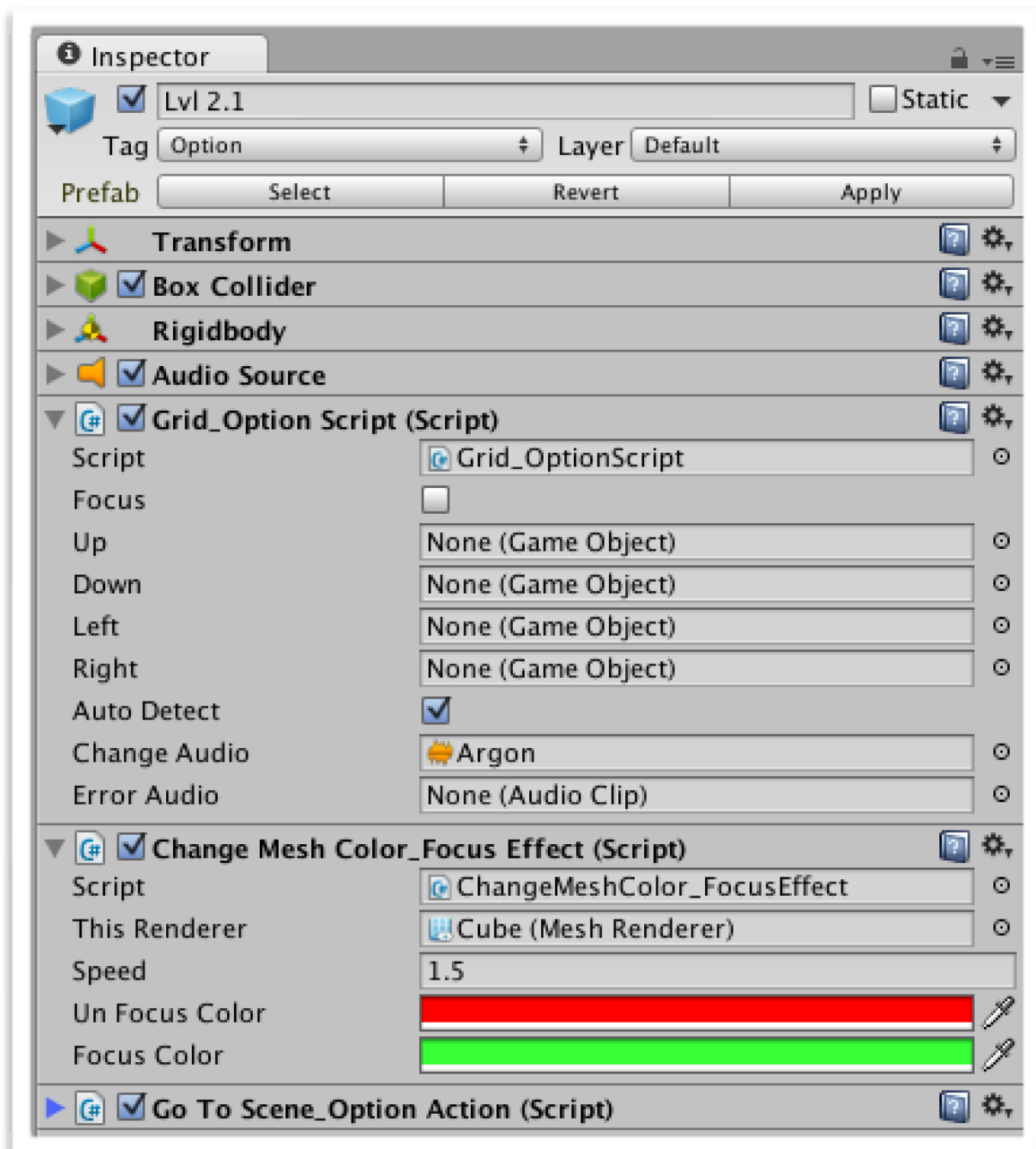
FocusEffect Scripts

These scripts can change a component of a GameObject based on whether or an Option is in Focus.

An example of this is in the LevelPicker.

In the screenshot below you can see the ChangeMeshColor_FocusEffect.cs.

This will change the color from Red to Green with a speed of 1.5 when the option is in focus.



Here is a List of All the FocusEffect Scripts

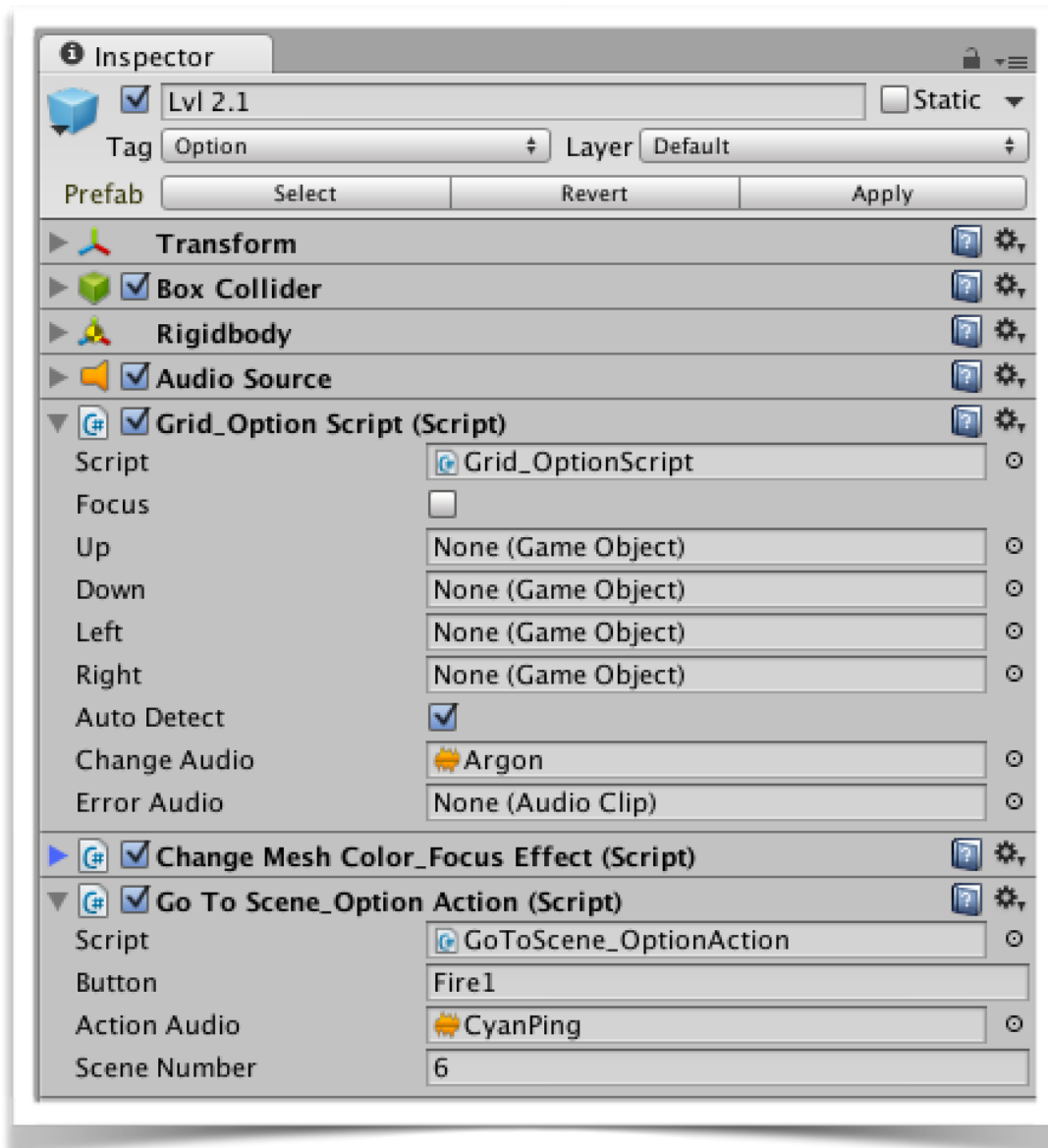
ActivateObject_FocusEffect.cs
ChangeLightIntensity_FocusEffect.cs
ChangeMeshColor_FocusEffect.cs
ChangeParticleSystemEmit_FocusEffect.cs
ChangeRotation_FocusEffect.cs
ChangeSize_FocusEffect.cs
ChangeTextColor_FocusEffect.cs
PulseMeshColor_FocusEffect.cs
PulseRotation_FocusEffect.cs
PulseSize_FocusEffect.cs
PulseTextColor_FocusEffect.cs

OptionAction Scripts

These scripts can change a change the action that option will execute when a button is pressed.

An example would be on in the LevelPicker.

In the screenshot below you can see the GoToScene_OptionAction.cs. This so when the Fire1 Button is pressed Unity load Scene Number 6.



Here is a List of All the OptionAction Scripts

ChangeValueBool_OptionAction.cs
ChangeValueInt_OptionAction.cs
ChangeValueStrings_OptionAction.cs
GoToScene_OptionAction.cs
GoToScene_WDelay_OptionAction.cs
Quit_OptionAction.cs