

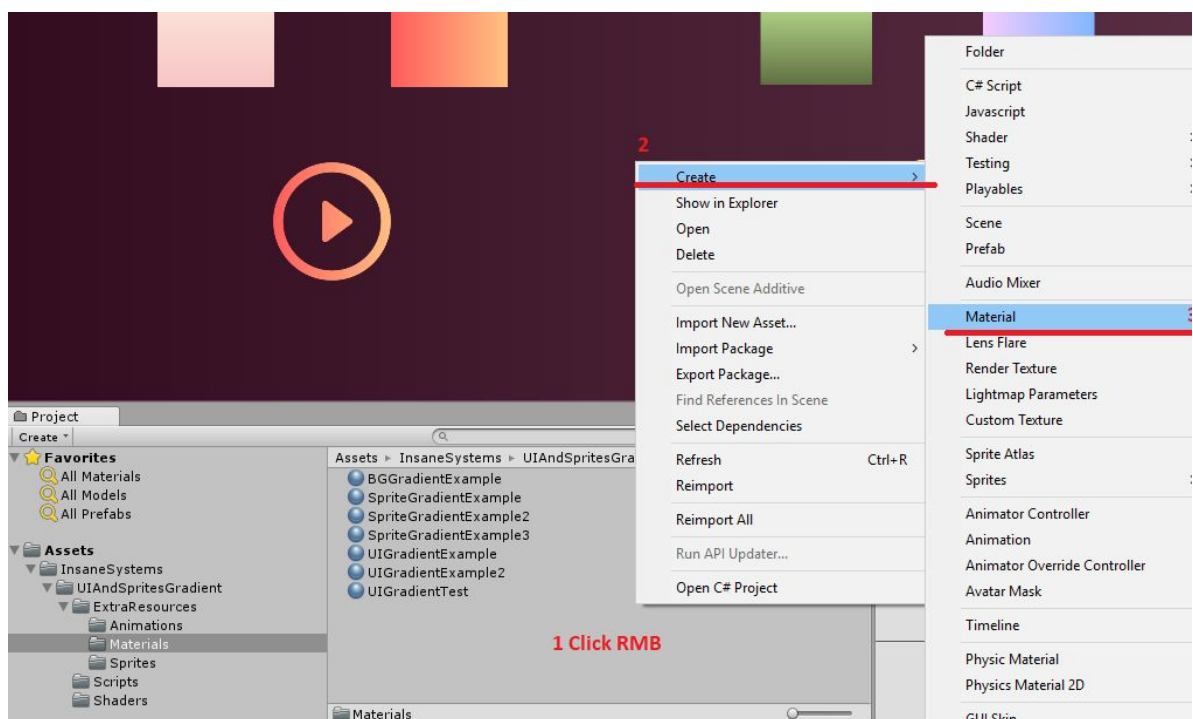
Insane Systems UI And Sprites Gradients Guide

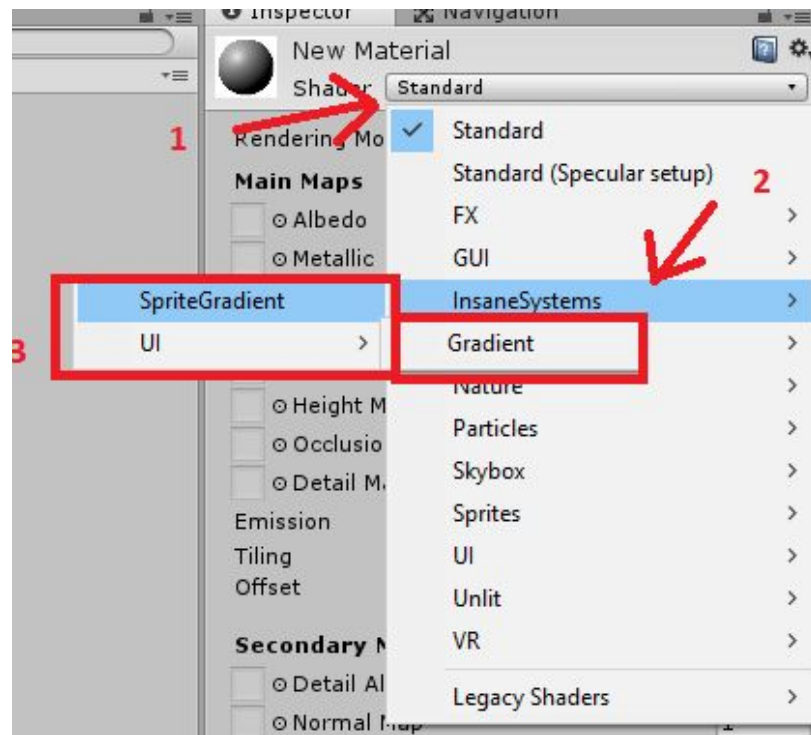
Content

1. Quick Start
2. Detail Guide
 - a. Gradients fade
 - b. Blend script and code usage
 - c. Animation
3. Updates
4. Contacts

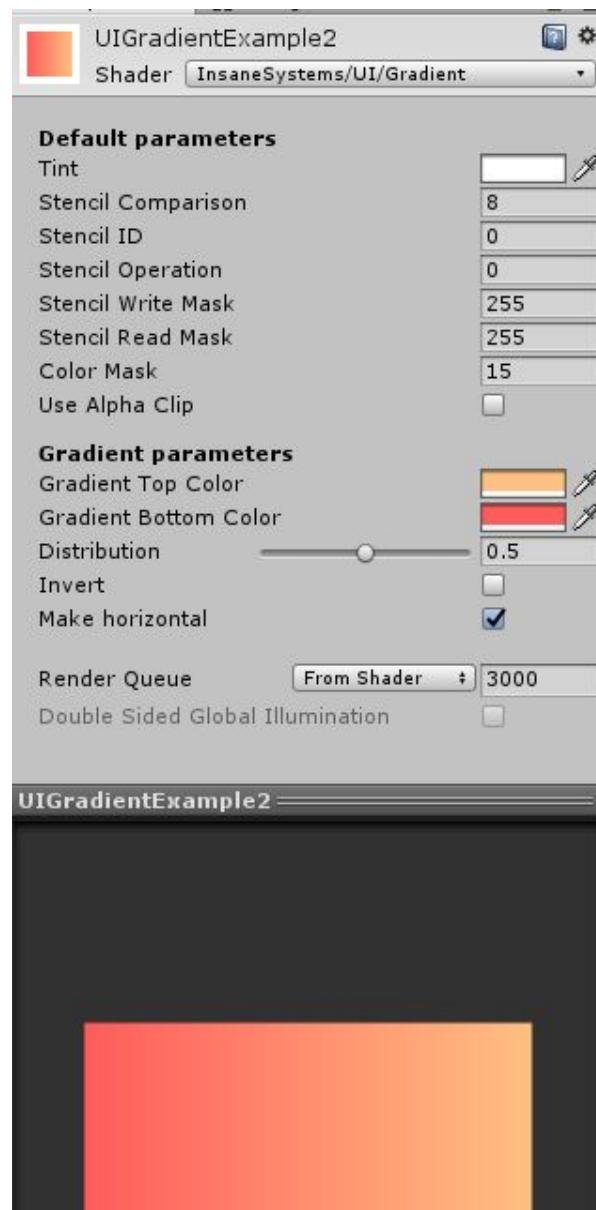
Quick Start

To create gradient, you need to create a new **Material** and select needed shader - **Sprite Gradient** for **SpriteRenderer** or **UI Gradient** for **UI Images**:

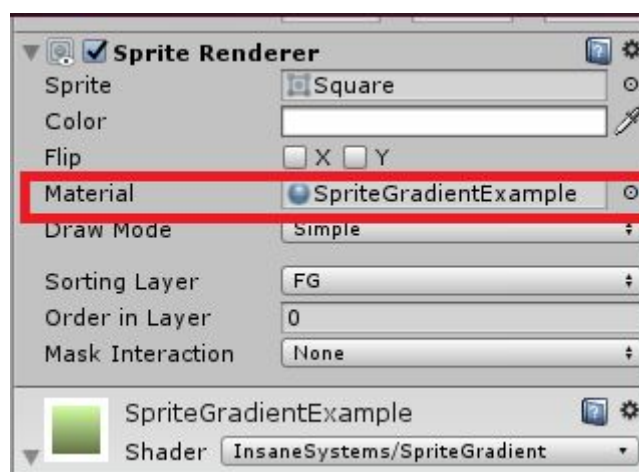




And now you can edit gradient material parameters:



To apply this material to **SpriteRenderer** or **UI Image**, just drag it into **Material** field:



After these steps it starts work. Now you can drag sliders to test how gradients works on your elements, and finally can use it as you need! :)

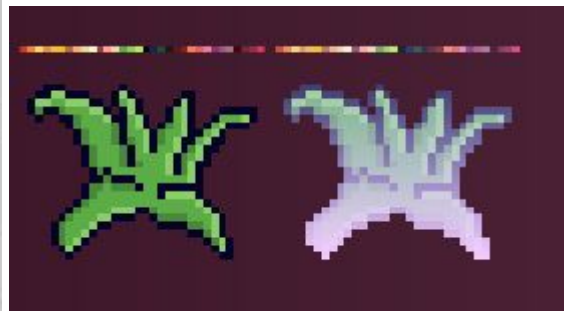
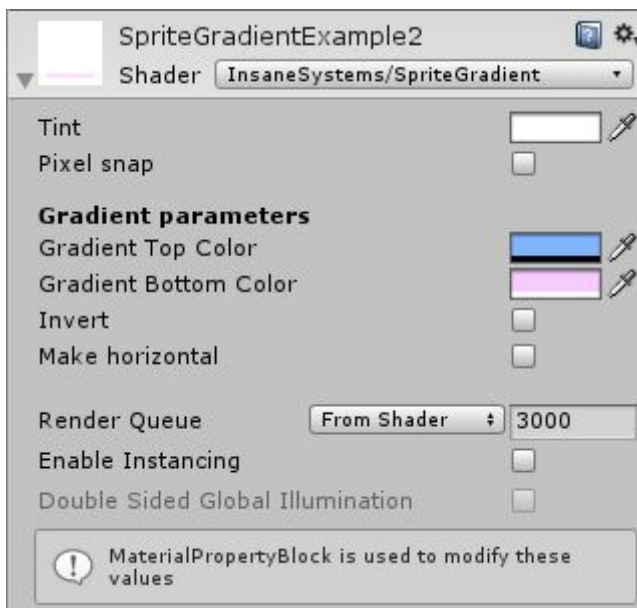
Note that gradient color alpha allows to make fade gradient overlay above your texture.

Detailed Guide

There will be not much additional info, because asset usage is very easy. But if you need more info, some asset features described below.

Gradients fade

By using gradient colors alpha you can create fade effects with your sprites:



It can be useful in some situations.

Blend script

Asset contains blend script, which allows you to smoothly change one gradient to another, calling its **StartBlend** method.

To use this script, add **GradientBlend** component to object with gradient material, which you want to blend during the game:

SCREEN

In this script you can setup colors of new gradient and time which needed to blend to this gradient when **StartBlend** method will be called.

Now you can call **StartBlend** method from your script:

```
var gradientBlend = GetComponent<InsaneSystems.UIAndSpriteGradients.GradientBlend>();  
gradientBlend.StartBlend();
```

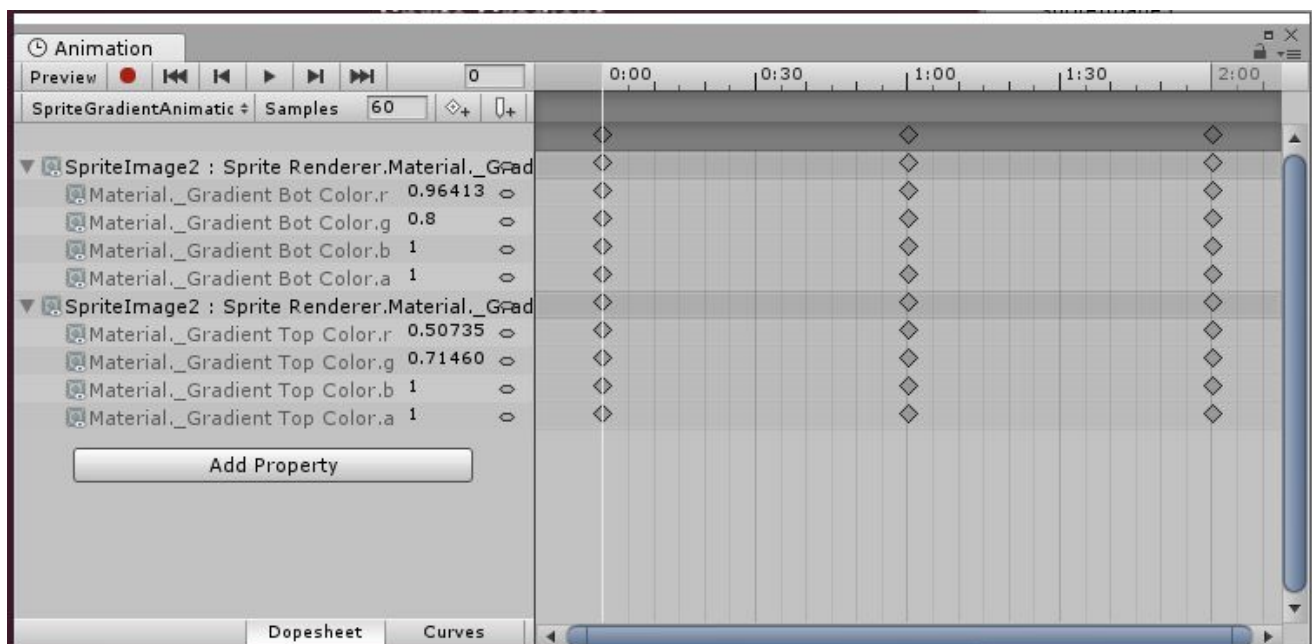
You can also blend from final gradient to default by calling **ResetBlend(false)** method. It will smoothly blend gradient back.

You also can directly access to material gradient parameters from your code like this:

```
editedMaterial.SetColor("_GradientTopColor", blendedTopColor);
editedMaterial.SetColor("_GradientBotColor", blendedBotColor);
```

Animation

You can animate gradients changes with Unity Animation tool by animating material parameters of SpriteRenderer or UI Image:



Example of animation can be found in **ExtraResources/Animations** folder.

Updates

In next updates we want to add angle value for better gradients position.

Contacts

You can ask your questions or send your suggestions to us. ☺

To contact use email **godlikeaurora@gmail.com**