Reflective UI Documentation

Quick Installation

Drag & Drop the *CameraRecorder* prefab that is located on "*Reflective UI / Prefabs*" to the scene. For the version 1.0, **TextMeshPro** and **Image** components are supported for **Screen-Space Canvas**

To make TextMeshPro's color reflective:

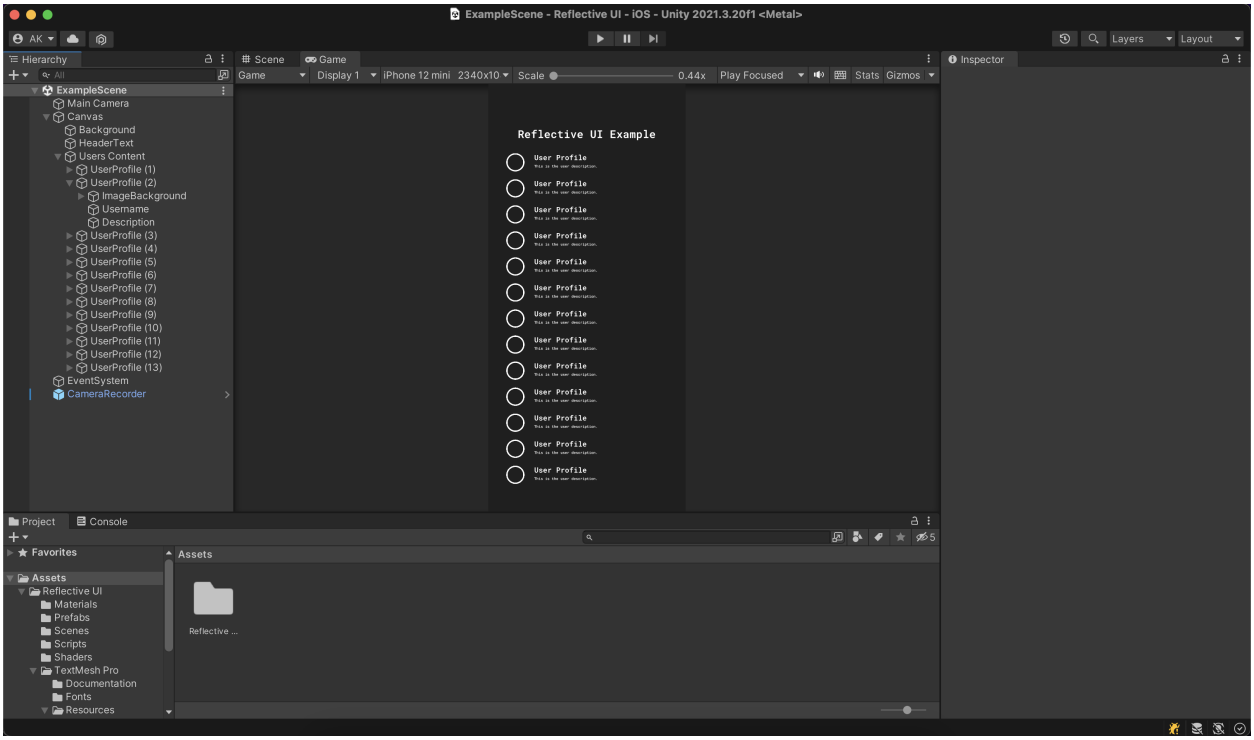
- Add "TMPColorHandler.cs" script to TextMeshPro's Gameobject.

To make Image's color reflective:

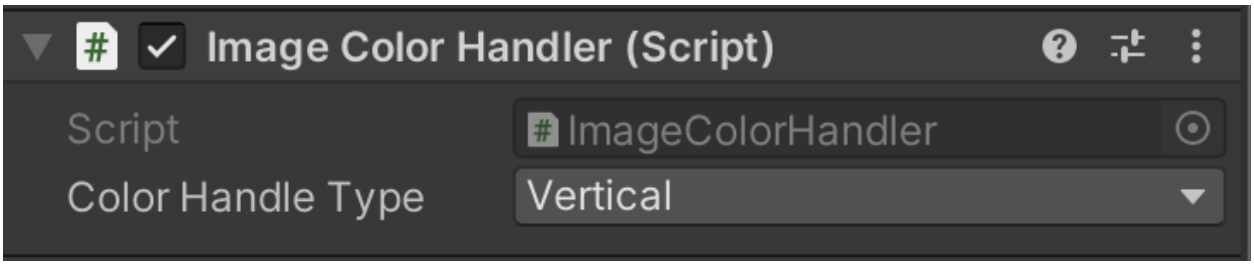
- Add "ImageColorHandles.cs" script to Image's Gameobject.
- Drag & drop the "Gradient Sprite" material that located in *"Reflective UI / Materials"* to the image's material.

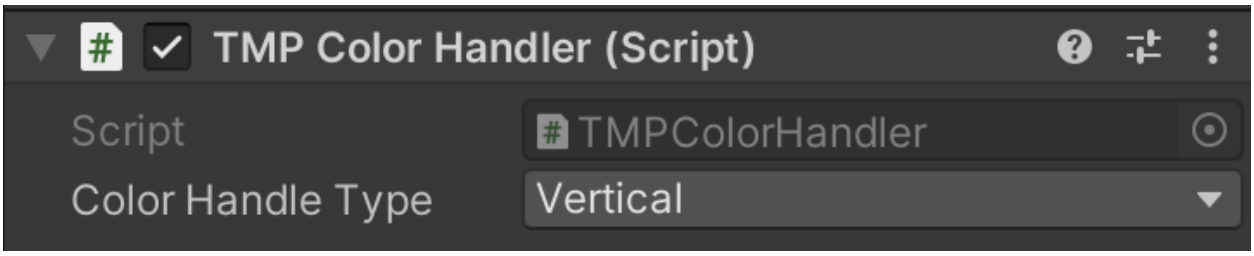
That's all. You're ready to go!

In addition, you can analyze “Example Scene” that’s located in the “*Reflective UI / Scenes*” folder.

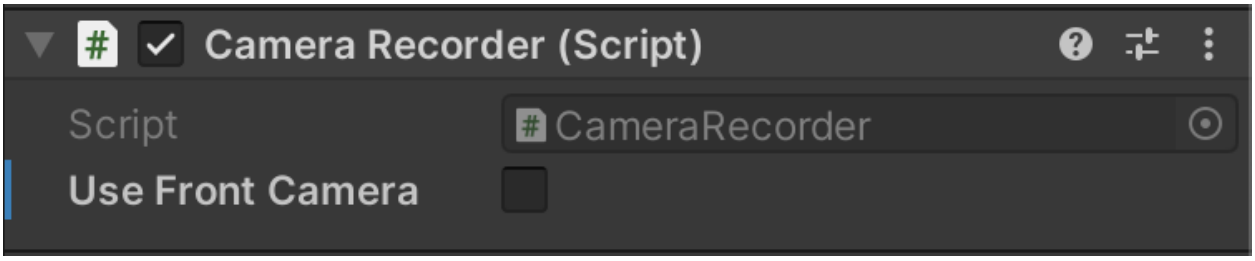


On the color handler components, you can select color handle type. Two types of handle are exist: Vertical and Horizontal. It just defines the direction of gradient.





On the Camera Recorder components's settings, you can select to use front camera or not (use back camera).



You can get color data from Camera Recorder component and use it anywhere you want in your application. Just call "`GetHorizontalColor(Vector2 screenPosition)`" or "`GetVerticalColor(Vector2 screenPosition)`" functions.