HealthEnergyBarTool_v0.5

HealthEnergyBarTool makes it easy to create dynamic state UI for your game, such as health bar, energy bar, hunger score, etc

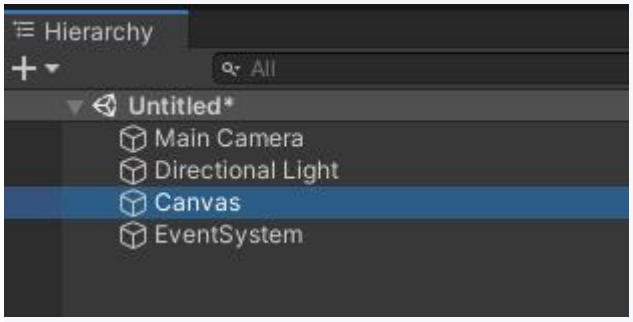
Not just a single bar of blood display, but a representation of many heart images like Zelda can be implemented quickly,or define different display images and colors in multiple states and without any code

Remember, there are many different ways to display, choose one or more of your favorites and join the game!

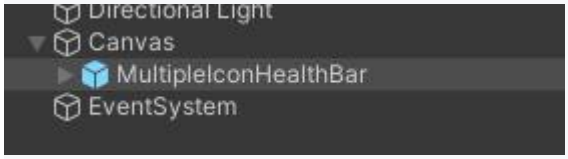
You can quickly see the effects with a demo scene

Quick start

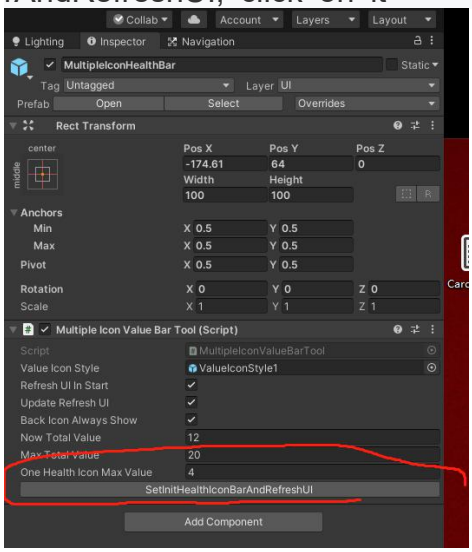
1. Right-click on the Hierarchy interface (or select GameObject from the menu) and create a Canvas through the menu "/UI/Canvas"
If you already have a Canvas in your scene, skip this step



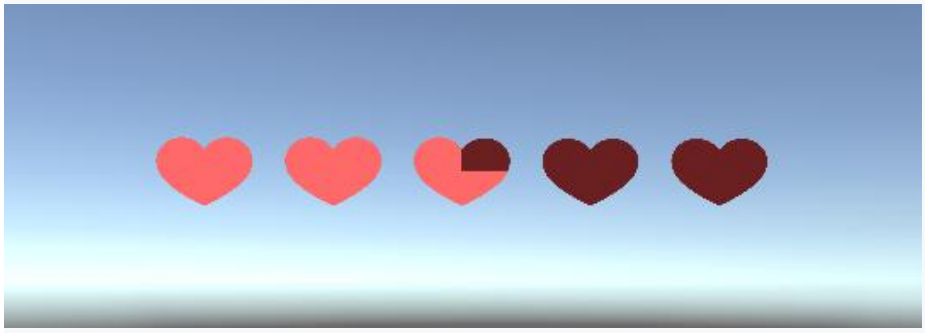
2. In the directory ”HealthEnergyBarTool/Prefabs”
pull (MultipleIconHealthBar.Prefab) to your game scenario, put him in the Canvas in children
just make sure the ROOT of MultipleIconHealthBar is Canvas



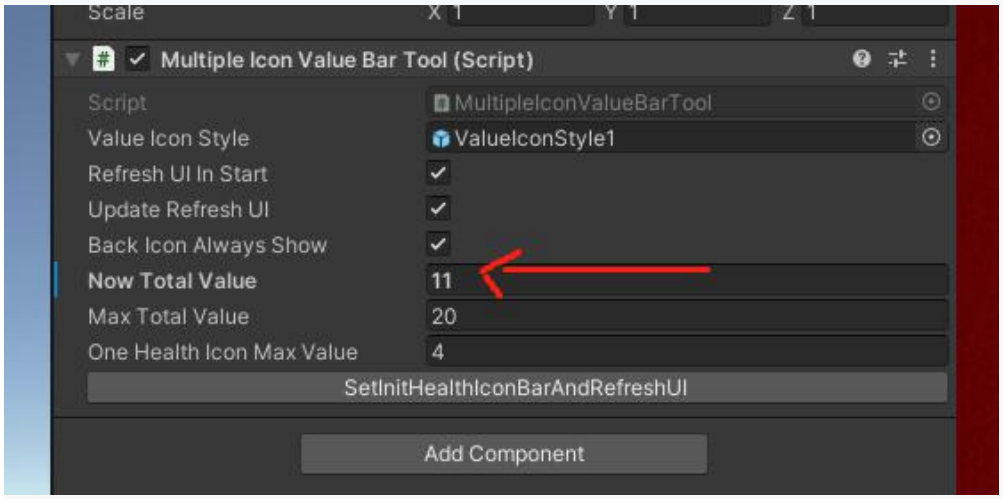
3. Selected MultipleIconHealthBar, see his Inspector panel, there is a button on the script component MultipleIconValueBarTool: SetInitHealthIconBarAndRefreshUI, click on it



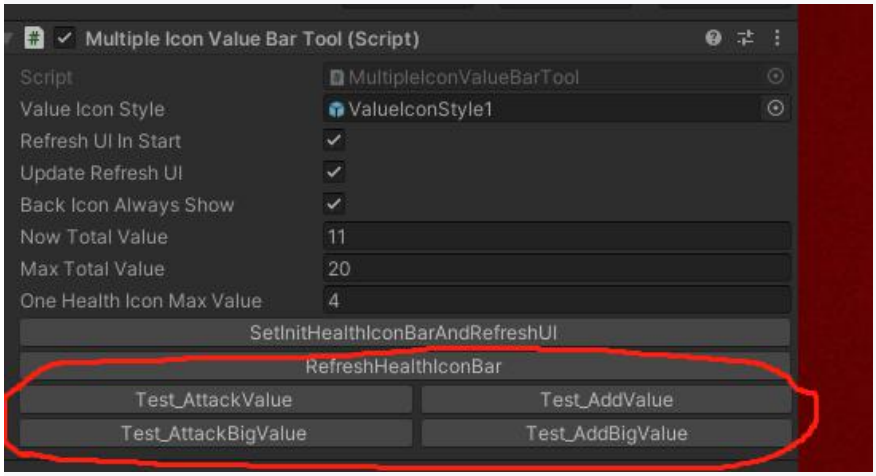
4.Now that you've generated a heart display in the scene similar to the Zelda Health UI,This UI represents a number as multiple ICONS if the location of the UI looks strange or invisible, adjust its position to match your game interface



Now you can play the game and use the mouse to change the nowTotalValue in the script. You can see the hearts change according to the value, and the animation is attached



While the game is playing, the script component may have four new test buttons that you can click to test the effect of your UI

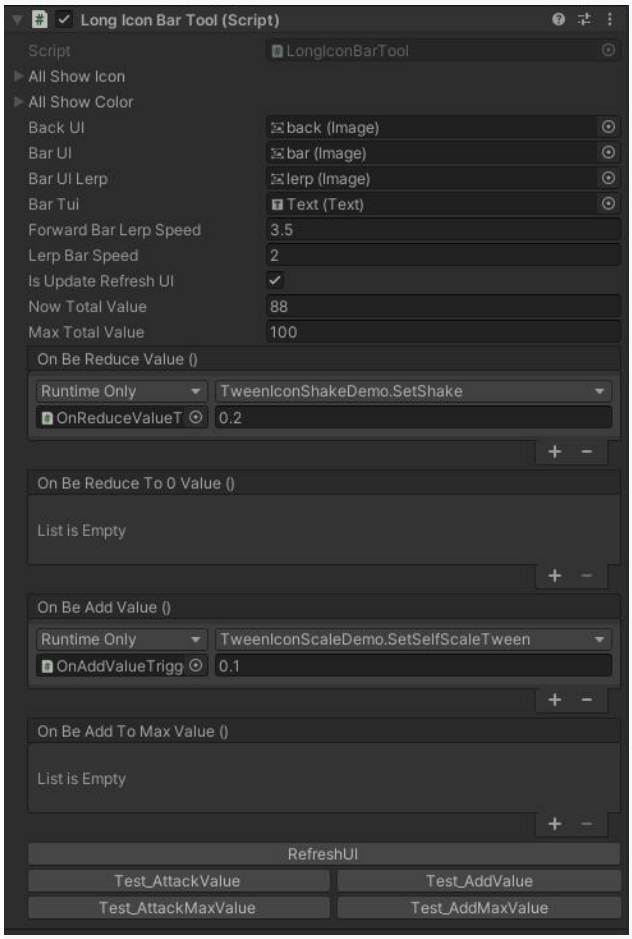


Introduction to Basic Styles

1. LongIconBar

This is the most common way to display, often used for health, energy and other values

Change the value dynamically by calling SetNowValue (value) and SetMaxValue (value).



AllShowIcon: When the array contains multiple images, when the value changes, the UI will display the image corresponding to the array according to the ratio of the current value to the maximum value(Reference Prefabs: OneLongIconBar_ManyStatelcon)

AllShowColor: When the array is set with multiple colors and the script value changes, the UI will change the color of the UIBar to the corresponding color in the array according to the ratio of the current value to the maximum value

BackUI: The background bar Image object for UIBar

BarUI: The main Image object for UIBar

BarUILerp: Image object of UIBar for delayed display, sandwiched between the back ground Bar and the main Bar(Can be null)

BarTui:Text hints for numeric values, text objects

ForwardBarLerpSpeed: The numerical synchronization delay speed of the main Bar. When the value is 0, it indicates no delay and instantaneous synchronization

LerpBarSpeed: The delay synchronization speed of BarUILerp

IsUpdateRefreshUI: Refresh the display state of the UI in real time.

If this option is enabled, there is no need to call any refresh function, only need to directly change NowTotalValue value to see the change of the UI in real time, but the performance cost is high

If not, you need to call the script's SetNowValue () or SetMaxValue () function in y our code to update the UI value

NowTotalValue: Current value

MaxTotalValue: Current maximum value

OnBeReduceValue: An event is triggered instantly when the value is reduced

OnBeReduceTo0Value: Triggered when the value is reduced to 0

OnBeAddValue: An event is triggered instantly when the value is increased

OnBeAddToMaxValue: The event that fires when the value is increased to the maxi mumvalue

RefreshUI(button): Updates the state of the object based on the parameter and var iable Settings of the current script

Test_AttackValue(button):A button for developers to test.Reduce a small number o f values

Test_AddValue(button):A button for developers to test.Reduce the maximum numb er of values

Test_AttackMaxValue(button):A button for developers to test.Add a small number of values

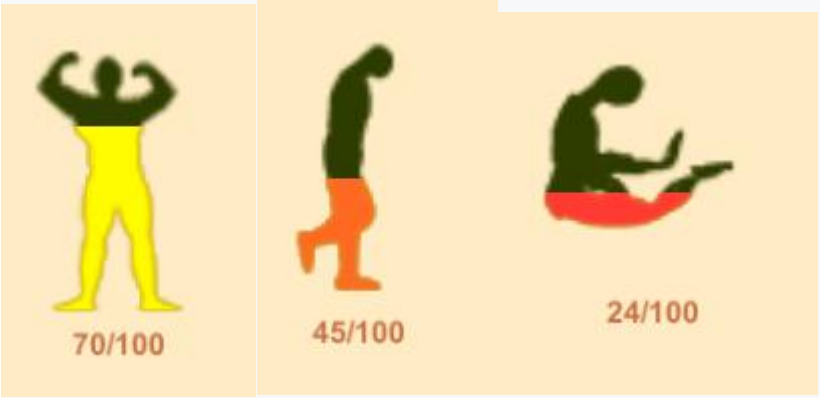
Test_AddMaxValue(button):A button for developers to test.Increase the maximum number of values

2. OneLongIconBar_ManyStateIcon

This style builds on OnLongIconBar and changes the current UI image based on th e ratio of the current value to the maximum value

It uses the same script as OnLongiconBar, see OnLongiconBar for details

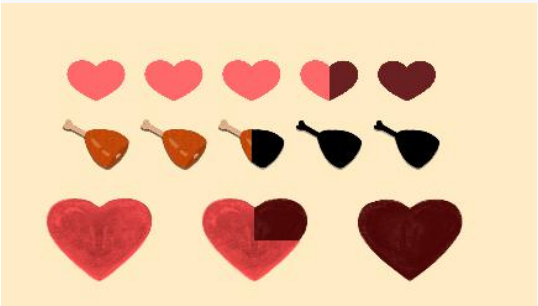
Change the value dynamically by calling SetNowValue (value) and SetMaxValue (value).

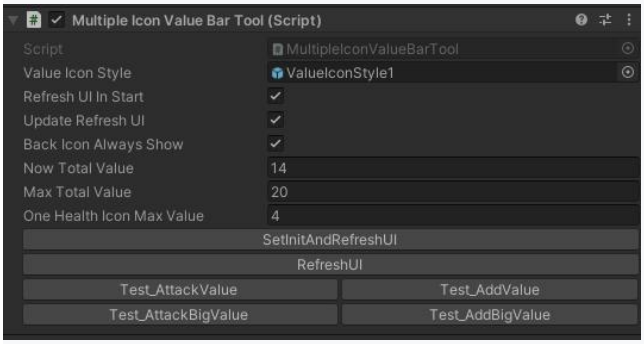


3. MultipleIconHealthBar

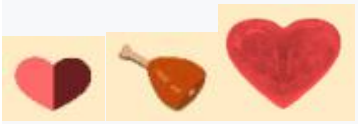
This style splits the value into paragraphs, each of which presents the value in a s eparate UI, similar to Zelda's Heart Health UI

Change the value dynamically by calling SetNowValue (value) and SetMaxValue (value). Update the effect of the UI by calling RefreshUI ()





ValueIconStyle: The style of each icon representing a numeric paragraph



RefreshUIInStart: When the game is playing, refresh the UI state in the start functi on

UpdateRefreshUI: Refresh the display state of the UI in real time.

If this option is enabled, there is no need to call any refresh function, only need to directly change NowTotalValue value to see the change of the UI in real time, but the performance cost is high

If not, you need to call the script's SetNowValue () or SetMaxValue () function in y our code to update the UI value,you also have to call RefreshUI() to refresh the UI 's effects

BackIconAlwaysShow: The background image of each icon is always displayed

If on: 

If off: 

If you don't need to set the background image at all, set the BackImage to NULL i n the paragraph style Prefab

NowTotalValue: Current value

MaxTotalValue: Current maximum value

OneHealthIconMaxValue： Evaluate the value that each paragraph represents

SetInitAndRefreshUI(button): SetInit the script and Updates the state of the object based on the parameter and variable Settings of the current script

RefreshUI(button): Updates the state of the object based on the parameter and var iable Settings of the current script

Test_AttackValue(button):A button for developers to test.Reduce a small number o f values

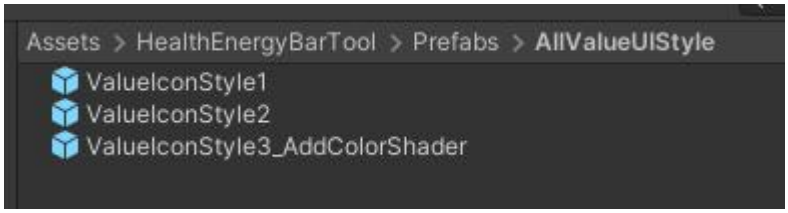
Test_AddValue(button):A button for developers to test.Reduce the maximum numb er of values

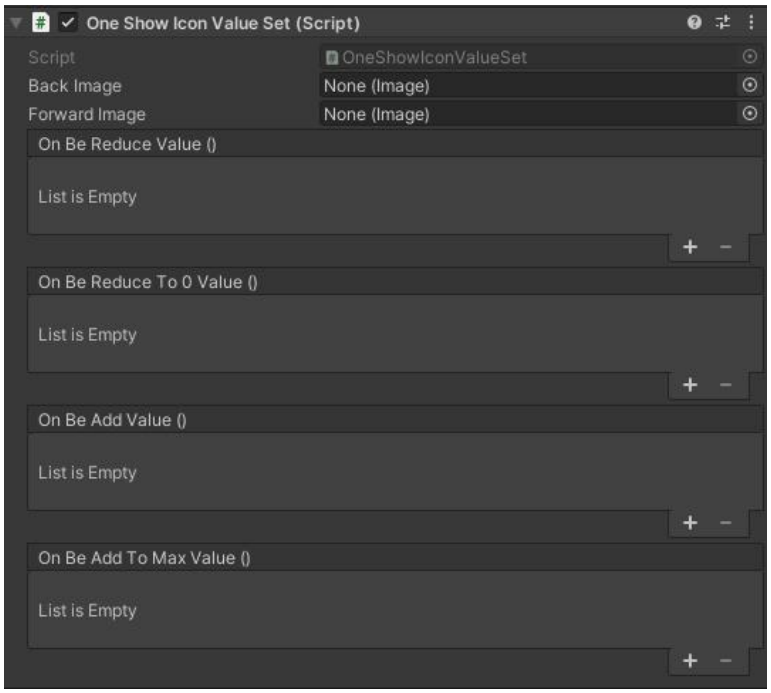
Test_AttackMaxValue(button):A button for developers to test.Add a small number of values

Test_AddMaxValue(button):A button for developers to test.Increase the maximum number of values

ValueIconStyle

MultipleIconHealthBar needs to set the style of each paragraph icon. We have built in 3 different styles for you to use directly. You can also customize your own style





BackImage: The background image object of the icon

ForwardImage: The main image object of the icon

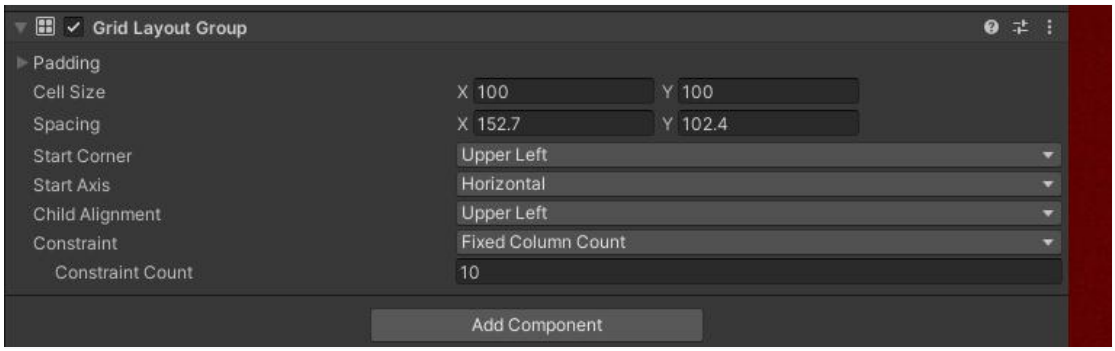
OnBeReduceValue: An event is triggered instantly when the value is reduced

OnBeReduceTo0Value: Triggered when the value is reduced to 0

OnBeAddValue: An event is triggered instantly when the value is increased

OnBeAddToMaxValue: The event that fires when the value is increased to the maximumvalue

On the parent of each icon, there is a UGUI component - the GridLayoutGroup,



Using this component, you can adjust the ordering of these ICONS, such as horizontal or vertical, you can adjust the spacing of each line, or the spacing of each icon, can be adjusted here, details can be seen in the UGUI official tutorial

TweenDemo

When the value decreases or increases, events trigger the UI to attach some animation effects to make it feel more like a game, such as the UI shaking when taking damage, and the UI zooming in and out when healing

`TweenIconShakeDemo:`This example code will make an object shiver for a while
`TweenIconScaleDemo:`The code in this example increases the zoom of an image by a multiple of an instant, and then gradients back to the original zoom at a certain speed
`TweenIconColorDemo:`This example code will change the color of an image to a certain color instantly, and then change it to the original color at a certain speed

These actions and animations can be customized with the event trigger provided. We have some simple animations built in and you can use them directly,

of course, you can also use Dotween and other more rich plugins