Attribute Menu

# Game Plan

Attributes show: Primaries, Secondaries, Vital on bars

+ Button for Primary, so that we can increase the primary when level up; the secondary will increase automatically with inherited values via modifiers.

A screenshot of a video game

Description automatically generated

Will require some design thought.

* Values are show with own frame so there’s a widget we can reuse
* FN to set value of numerical data represented
* Rows with text and framed value for each attribute
* Own widget
* Primaries are similar but also have button
* Row widget + Own widget for widget
* Reuse functionality from 2ndary?
* One big widget for menu itself
* How to construct?
* Not all 2nd attributes shown
* Scrollbar to move down to see additional values

A screenshot of a game menu

Description automatically generated

So:

* Framed Value widget for numbers
* Row widget contains text box and also framed value widget
* Primary is a row widget with and additional button widget
* Attribute menu widget contains many row widgets and scrollbar
* Health and mana own widget progress bars
* Close menu widget

A black background with yellow text

Description automatically generated

# Framed Value

New UI BP

Attribute menu an Overlay? New folder is tidier

WBP\_FramedValue based on AuraUserWidget

+SizeBox

FillScreen to Desired

Override Height and Width to arbirary values 80/45 to work with for now, change later if needed

Box Size (rename to Box Size Root) set to variable; in Graph set Height and Width of box as variable floats

A screenshot of a computer

Description automatically generated

Set category for W + H

Drag in Size Box Root and call set overrides from Event PreConstruct

A screen shot of a computer

Description automatically generated

Check defaults, currently at zero and collapse to Function



Size box now exists. To add to the menu:

* Background colour/image
* Border
* Text

Since things are going to be on top of one another will need to add an overlay that is a child of the size frame and set to fill it

A screenshot of a computer

Description automatically generated

Add an image of a child of the overlay, give it a value, either plain black RBG sliders or an asset:

A screenshot of a computer

Description automatically generated

This is an animated asset like the health/mana globes and has params we can set to preference:

A screenshot of a computer

Description automatically generated

Renamed the new elements and set as variable, then set in the graph:

A screenshot of a computer

Description automatically generated

Remember to set a default value for the brush

A screenshot of a computer

Description automatically generated

This way we can set variations, like colour, in child BPs if we want

Collapse to Fn

Add new image for the border

Fill H + V

For image pick desired border asset

A screen shot of a computer screen

Description automatically generated

Change Draw As from Image to Border and choose margin size – I went with 0.5 for all.

The size box can be changed in size and the border and background will stretch to match

A screen shot of a computer

Description automatically generated

For now I won’t parameterise the border, we can always change that later if we want!

Add Text block. Will set default as a 2 digit number for convenience

A screenshot of a computer

Description automatically generated

Set alignment (centre)

Set Justification (center)

Change font

A screenshot of a computer

Description automatically generated

Play with font size and outline settings for appearance

Set the text as a variable so we can set the value with Fn later

# Text Value Row

New Widget WBP\_TextValueRow

Add Sizebox, override W and H, as before with the frame

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

We’ll add elements left to right; first text, then the framed value widget

Add Horizontal Box and text to that box, horizonal align left, centre vertical

A screen shot of a computer

Description automatically generated

Set default text, font, fontsize outline and fontspacing to taste

A screenshot of a computer

Description automatically generated

Add FramedValue Widget, set to fill, right, center

A screenshot of a computer

Description automatically generated

Add a spacer after it to modify the size a little

A screenshot of a computer

Description automatically generated

Test with longest attribute names (intelligence, critical hit resistance)

A screen shot of a computer

Description automatically generated

Too long! Make the size box bigger!

A screen shot of a computer

Description automatically generated

Better!

The spacer works well, but remember we will also need to add a button in a child class

For this we can add a Named Slot

A screenshot of a computer

Description automatically generated

Slot will allow children to have more widgets on the named slot

Tinker a little for overall taste:

A screenshot of a computer

Description automatically generated

# Text Value Button Row

New widget based on the Text Value row

A screenshot of a computer

Description automatically generated

Includes all elements by default but only the named slot is accessible.

However, we will need to change the text and the numerical value for each instantiation, so we need to expose them!

For now, let’s add the button. Start with an overlay and an image for the button border/background

A screenshot of a computer

Description automatically generated

Center and draw as image, set size (I picked 45 \* 45 to match the value box size)

Add button

A screenshot of a computer

Description automatically generated

Now pick styles for Normal, Hovered, Pressed and Disabled

This project has existing assets to use out of the box

A screenshot of a computer

Description automatically generated

(set to draw as image)

We can add a single character of text, a +, so we know it’s for adding to the attribute!

Alight and justify as usual, set a font and appearance to taste

A black and grey plaid

Description automatically generated

We’ll need functions to set text and numerical value and make the button functional