Week 3 - UMG

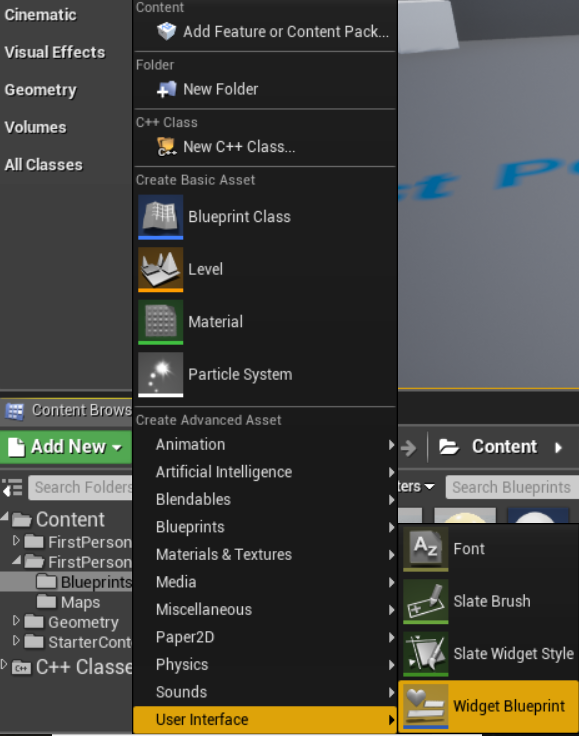
**Section 1 – Creating a new widget**

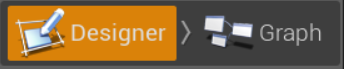
Think of widgets as self-contained canvases from Unity, they have their own blueprint editor for scripting functionality for that Widget

To create a widget navigate to:

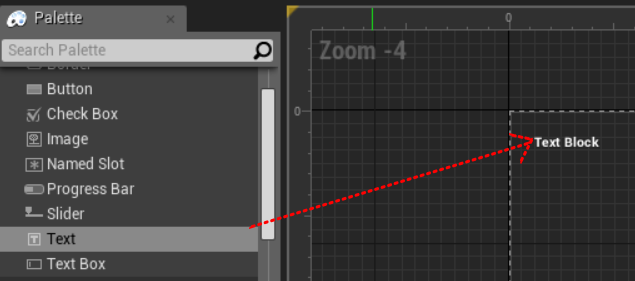
Add new -> User Interface -> Widget Blueprint

Name it “HUD\_BP” and open it up

****

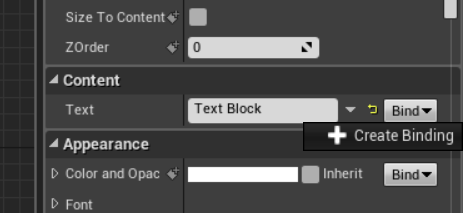
Make sure you are in the “Designer” mode in the top right corner, “Graph” is your blueprint editor.

Find the Palette tab on the left and drag a “Text” object into the top of the canvas

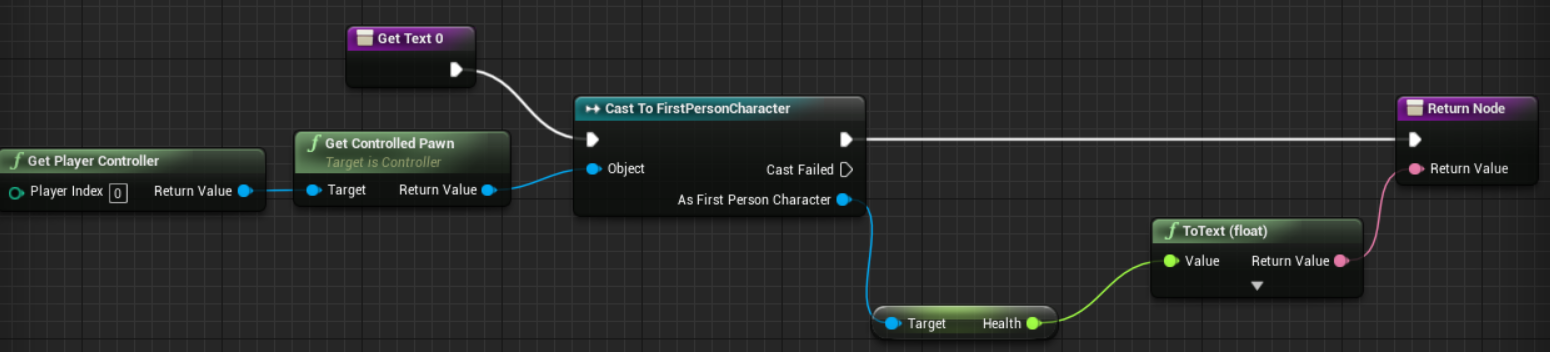


We are now going to bind this Text to the players health.

Make sure your new Text object is selected then navigate to the details panel on the left, find the “Content” header and select the “Bind” dropdown next to Text, the press “Create Binding”.



The blueprint graph for this new binding should open automatically. Recreate the blueprint code below:

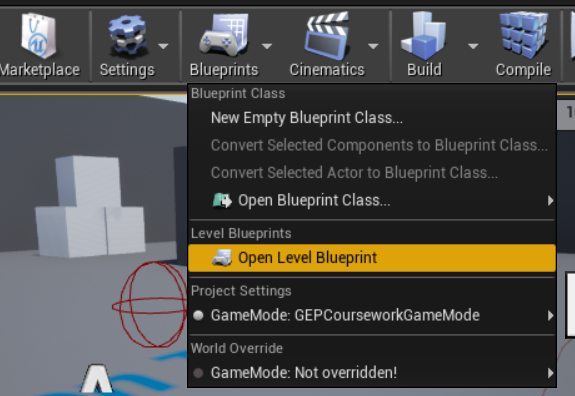


The Text in this widget should now automatically display what the health variable of the player is.

**Section 2 – Displaying the HUD widget**

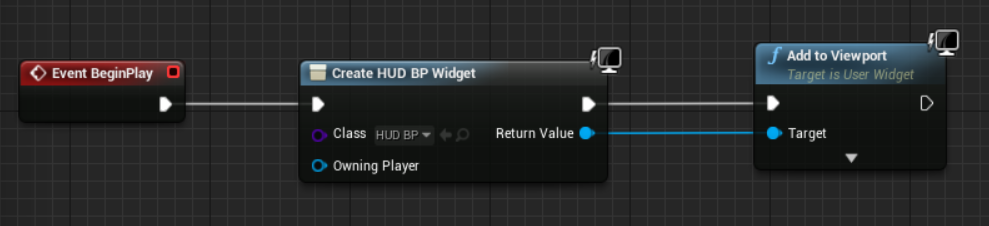
*For simplicity in demonstrating this we are going to use the Level Blueprint to spawn our widget, the Level blueprint is scripting for this specific level. Under normal circumstances it would be best to hook this code into the initialization chain of your Gamemode.*

Return to the main editor window and on the top bar you will find Blueprints->Open Level Blueprint.



Create a BeginPlay node then hook it up like this:

*Don’t forget to use the “Class” dropdown on Create Widget to select your HUD\_BP.*



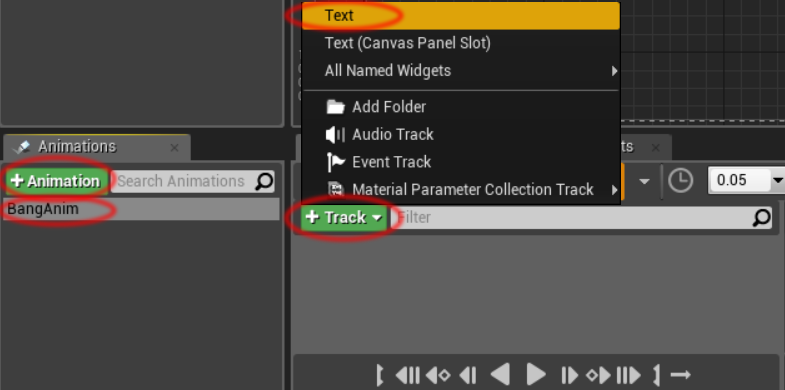
Play your game then walk over the pickup, the text should update.

**Section 3 – UMG Events**

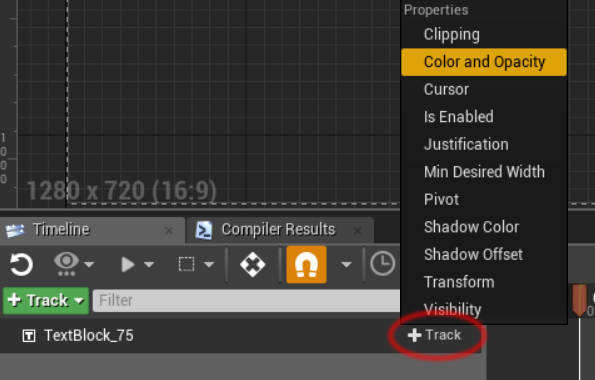
For this section we are going to make a basic UMG animation then create an event that plays the animation. When the player fires their weapon “bang..” will flash up on the screen.

Drag another text object from the palette into the center of the screen and rewrite the text to say “bang..”

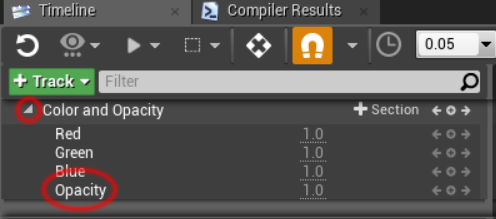
1. At the bottom left of the editor press the “+ Animation” button
2. Name it “BangAnim”
3. Press the “+ Track” Button
4. Select “Text”



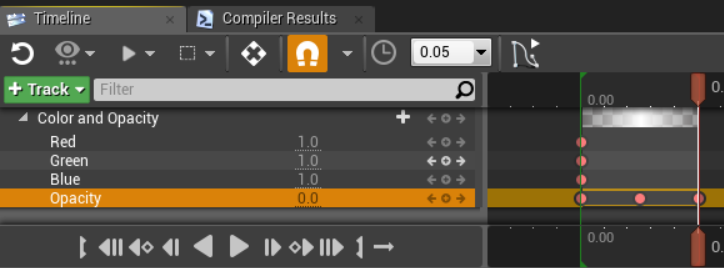
Press the “+ Track” button then select “Color and Opacity”



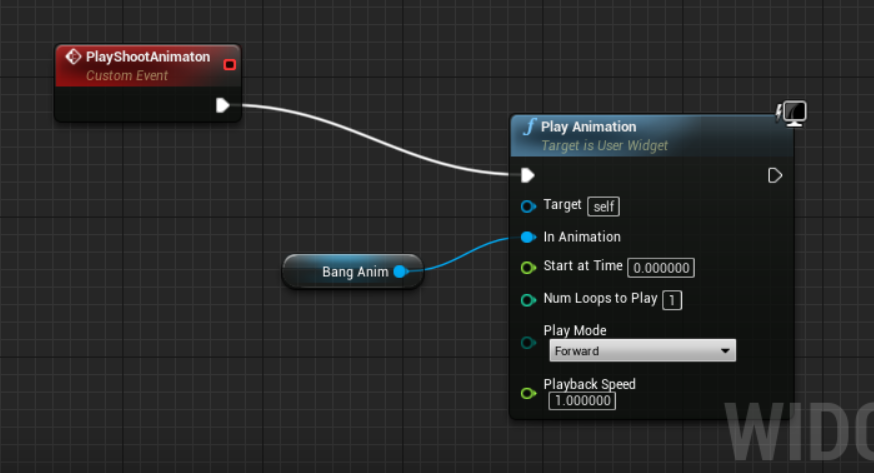
Expand the selection of “Color and Opacity” then select “Opacity”



1. Set the opacity to be 0.0
2. Drag the thin red end bar to 0.5 seconds.
3. Drag the marker to 0.25 then set opacity to be 1.0
4. Drag the marker to 0.5 then set to be 0.0
5. Press play and it should quickly fade out and back in.



Go the “Graph” mode in the top right and create a new custom event called “PlayShootAnimation”, drag your new animation from the Animations tab in the bottom left into the graph and drag out a Play Animation node.



**Section 5 – Calling the event**

*As the firing of the weapon is handled in C++ we are going to need to create a new Blueprint Implementable Event in our Character class. This will be called inside the OnFire function. We can then listen for this event in blueprint and call our UMG event.*

In the Character.h create a public or protected function definition:

UFUNCTION(BlueprintImplementableEvent, Category = "Weapon")

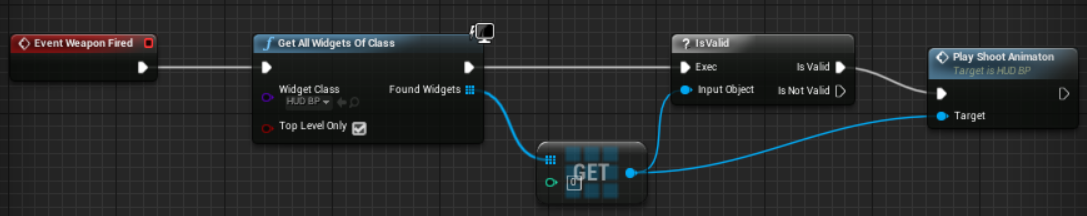
void WeaponFired();

In Character.cpp find the OnFire() function and insert where appropriate:

WeaponFired();

Compile then open up the FirstPersonCharacter Blueprint

Create a Weapon Fired event node then replicate as below:



When you play the text should flash up on screen.

**Challenges** (C++ recommended)

1.Give the character an Ammo variable and update it on screen.

2.Reload ammo with R key

3.Only allow the player to fire if they have ammo. *(C++ recommended)*

4.Create a power meter that charges over time, press space to activate it at 100%.

5.When the power is activated give the player a speed boost for 10 seconds.

6.At full charge animate the meter to pulse or flash.