Creating a HUD

Steven Harris

TABLE OF CONTENTS

1. UNREAL MOTION GRAPHICS

The recommended way to create a HUD is to use the Unreal Motion Graphics (UMG) system. The links below will take you to the key documentation pages to learn more about UMG.

UMG User Interface Guide: https://goo.gl/1KmHN7

UMG User Interface Designer Quick Start Guide: https://goo.gl/PySYWC

2. CREATING A HUD

2.1 OVERVIEW

This example will walk you through how to create a HUD element (called a widget) to display the current health of a player character, in the **2D SideScroller Tempalte game**. These steps can easily be changed to show any other type of data, from any other blueprint.

This involves four basic steps:

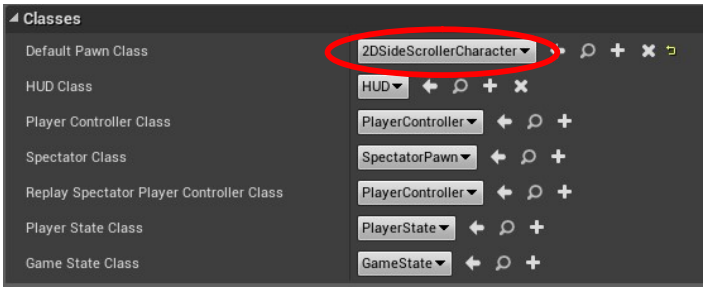
1. Add a variable to your player character to store the current health value.
2. Creating a layout of widgets (text, progress bars etc).
3. Linking those widgets to a specific variable in a blueprint so it can display the data.
4. Drawing those widgets on the screen

NOTE: As Unreal Engine 4 is constantly under development, menus locations and names often change. This tutorial was created with version 4.13.1.

2.2 TUTORIAL

The above steps are explained in more detail here. To Create a HUD:

1. Discover which blueprint class has been setup as the player character class (also called the PLAYER PAWN), ie which blueprint the player is using to control the game character. Once you know this class, create player health variables in that Blueprint.
    a. Go to the GameMode Blueprint by selecting BLUEPRINTS menu | GAMEMODE EDIT *<your_game>*GAMEMODE | EDIT *<your_game>*GAMEMODE. Check to see which class is setup as the DEFAULT PAWN, see example below, this is the class which the player is controlling and you should therefore add the health variables there.

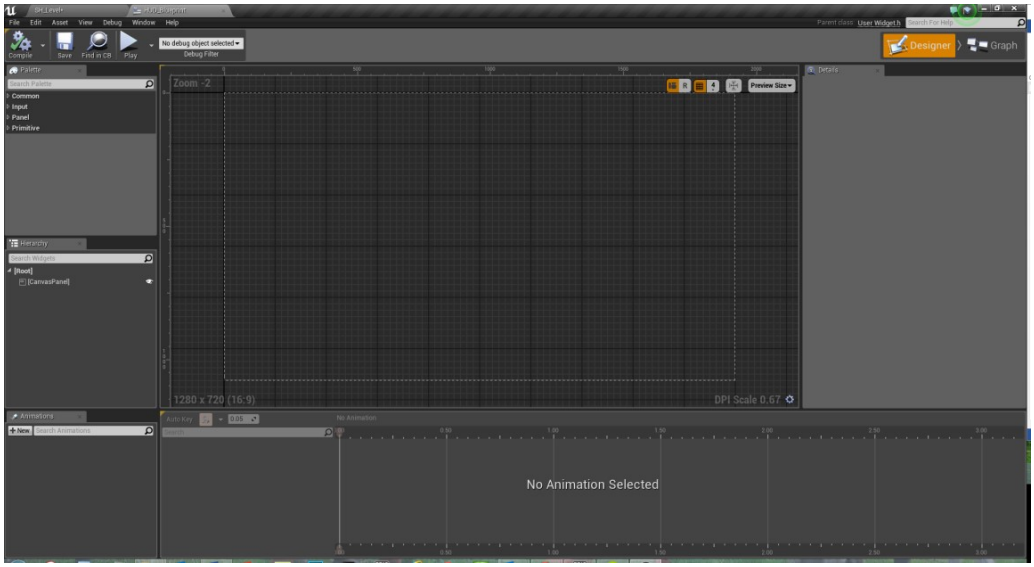


b. In the required class blueprint (see previous step), add a float variable for the current (HealthCurrent) and give it a default value of 1.

    i. When you first add a variable, there is no box in the DETAILS tab to add a default value. To show this DEFAULT VALUE box you must first COMPILE the blueprint, after which the DEFAULT VALUE box appears in the DETAILS tab.
    ii. It will become clear later why the health variable is a floats and set to a value of 1.0.

2. Create the HUD Blueprint

    a. In the CONTENT BROWSER, create a folder called 'HUD'.
    b. In that folder right click and select NEW ASSET | USER INTERFACE | WIDGET BLUEPRINT. Give it a name of: myHUD
    c. Double click on the newly created HUD Blueprint to display a screen as below.



    d. In the top right of the HUD Blueprint screen, you will see the DESIGNER and GRAPH buttons. The DESIGNER will be selected, as below.

    e. The DESIGNER is where you create the layout of the various HUD widgets. The white dashed line in the main workspace, represents the current screen dimensions and is caller the LAYOUT AREA. Any widgets displayed in this area will appear at that same position in the game.
    f. The GRAPH tab is where you add behaviours to link the widgets to data.
    g. In the top left, in the PALETTE tab, expand the COMMON list and drag a PROGRESS BAR onto the middle bottom of the LAYOUT AREA. Resize the progress bar to make it slightly longer

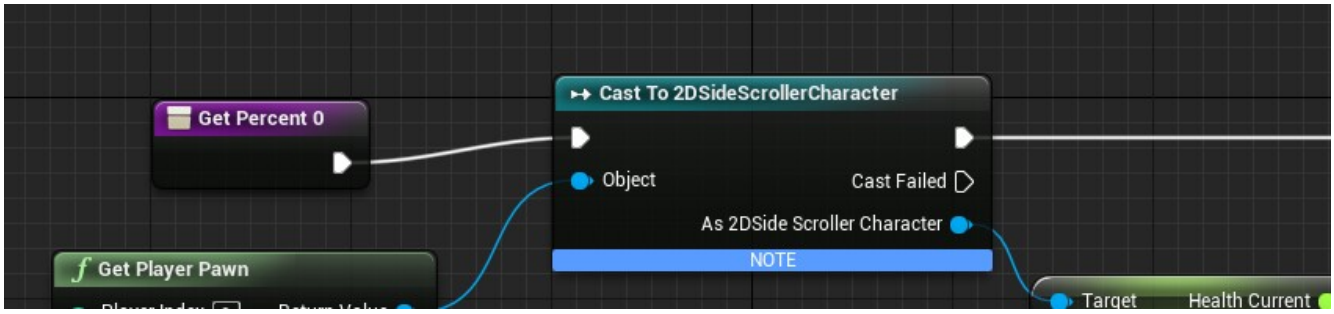
h. Then drag a TEXT item from the PALETTE and place it to the left of the progress bar. In the DETAILS tab, change the contents of the TEXT property to: Health
Your health bar should now look like the example below.

i. Select the health bar. In the DETAILS tab to the right of the screen look at the PROGRESS section. The PERCENT number controls how much the progress bar is filled. This entry ONLY ACCEPTS VALUES FROM 0 to 1. It does NOT work with percentages. It is easier if your Blueprints work with values from 0.0 to 1.0 rather than 0 to 100, as there will be no need to convert your health values to the range 0.0 - 1.0 before displaying it on the progress bar.
    i. This is why we set the health variables to floats and gave them a value of 1.0, as this will be the maximum value for the progress bar.
    ii. It is possible to have your game work with percentages from 0 to 100. However, you will have to write some blueprint code to convert your percentages from 0 to 100, back to 0.0 to 1.0 before displaying them on the HUD. You do this my dividing your percentage health by 100, for example 97% ÷ 100 = 0.97

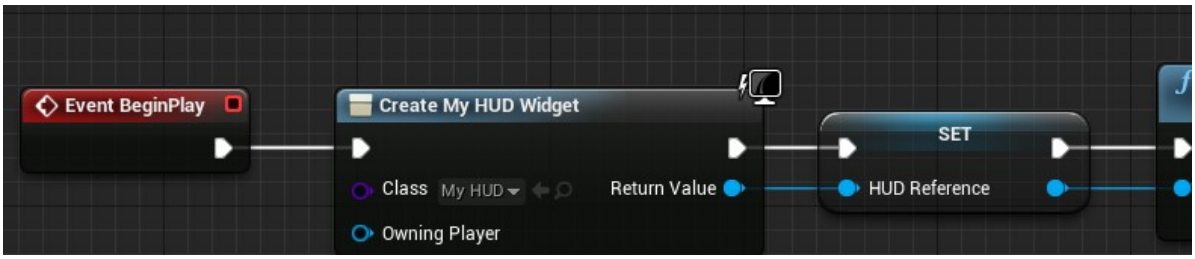
j. We now need to link the progress bar to the variable controlling the player health. To do this, select the progress bar and in the PROGRESS section of it's DETAIL window click the BIND button next to PERCENT, then CREATE BIND. This opens a blueprint window in which we can create the blueprint code to extract the health value from the required player character class and link it to this progress bar.

k. In this new Blueprint, create a sequence as below. Note that the CAST TO command requires a PLAYER PAWN connection. The GAME MODE screen defines 2DSideScrollerCharacter as the player pawn, *see the very first step in this process.*

l. You have now created the HUD widget and added some blueprint to the health bar to link itself to the value of the 'score' variable in the player character blueprint.



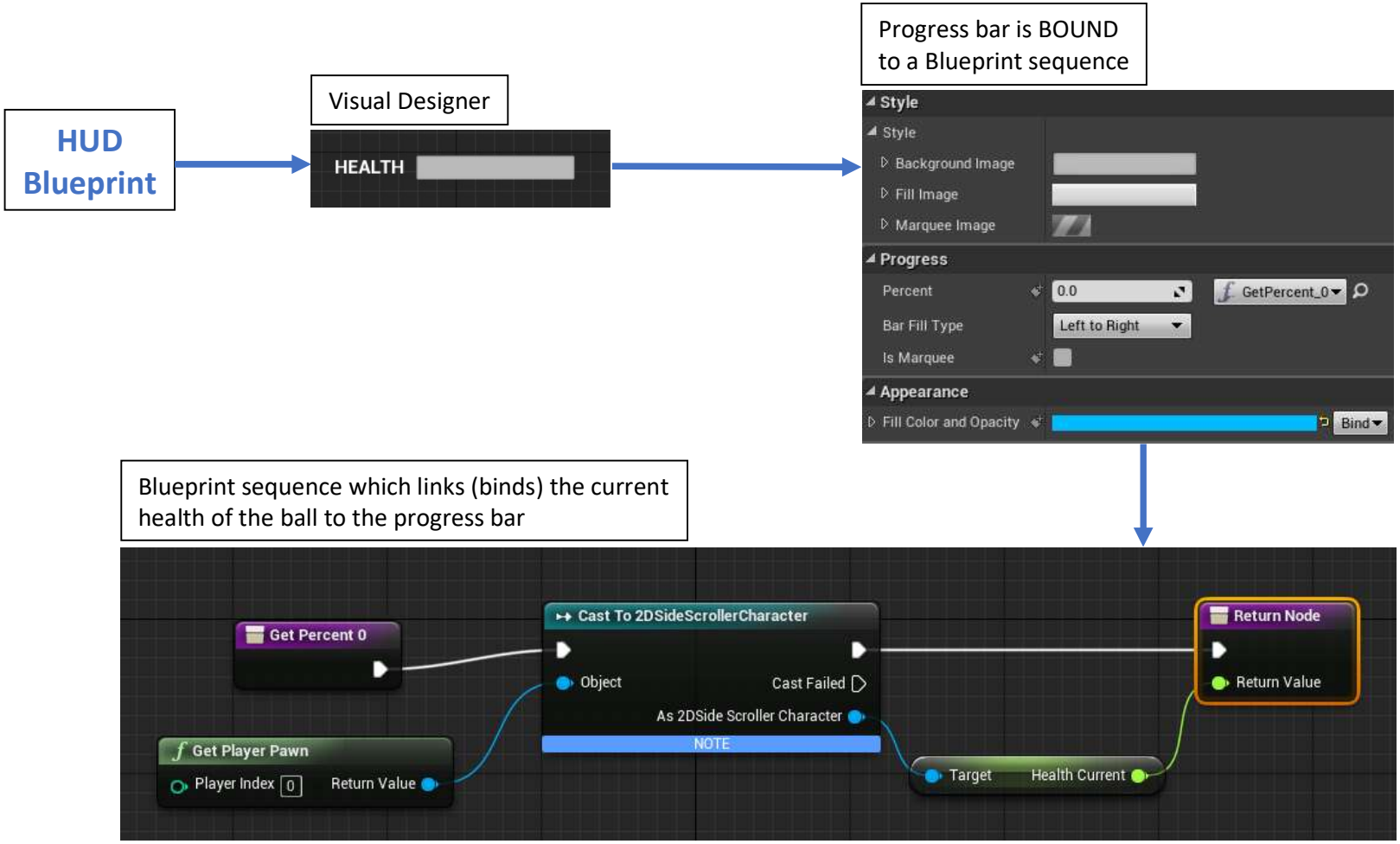
2. The next step is to display this 'widget' on the screen.
    a. Go back to the 2DSideScrollerCharacter blueprint where you added the health variable. In the EVENT GRAPH tab of the blueprint, find the EVENT BEGINPLAY node. If this node is already attached to another node, then insert the code on the next page immediately after the event node, and make sure it is linked to any existing code when you are finished. See example on next page.



b. In the above code, make sure the CLASS pin of CREATE MY HUD WIDGET is set to your own HUD blueprint you have created. When dragging out from the RETURN VALUE pin, select the PROMOTE TO VARIABLE option as this will create the SET node for you. You should also rename the variable in the SET node to HUD Reference, just so the code is easier to read.
    i. This code will display the widget on the screen.

c. You should now create an actor which you can place into the level which contains blueprint code to reduce the value of the score variable in the player character blueprint.
d. Your HUD is now setup and when you run the game you will see the health bar reduce when you touch the actor which is setup to reduce the score variable.

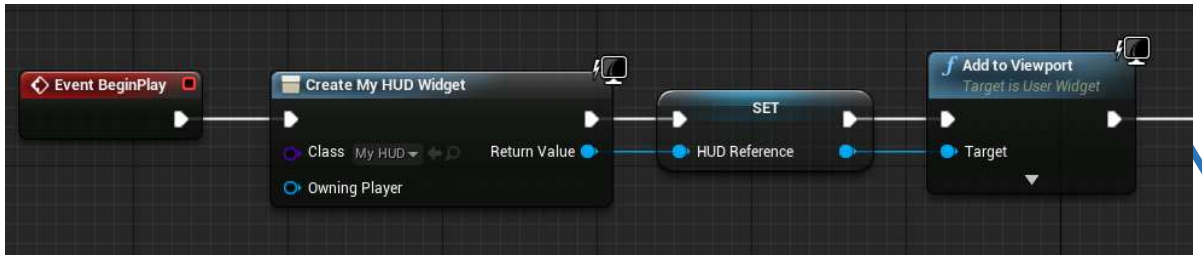
3. EXAMPLE: HUD ELEMENTS TO DISPLAY PLAYER HEALTH



Blueprints | Project Settings | Gamemode | Edit …….*mode*

Check which Blueprint is the Pawn (player character)

In the player character, display the HUD



A Take Damage FUNCTION inside the Pawns Blueprint which can be called to a