

14. Volumes

Volumes são áreas tridimensionais, dentro de seus níveis e cada um deles serve a um propósito específico. Alguns desses propósitos incluem:

- Causar dano ao Jogador.
- Bloquear o Player de entrar no volume, agindo como uma superfície de colisão.
- Alterar a forma como um nível é calculado para iluminação ou visibilidade.

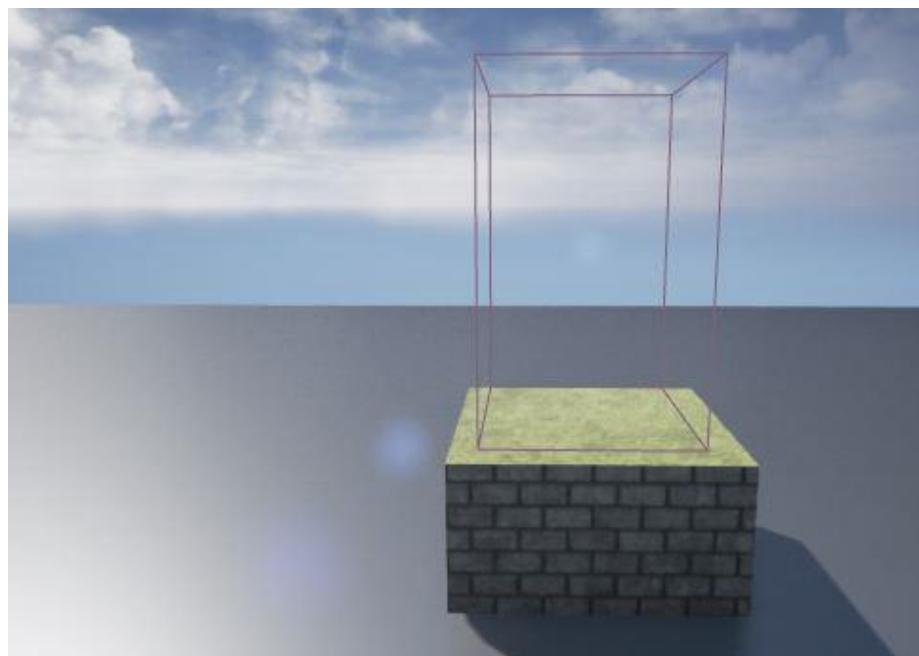
14.1. Usando volumes

Um aspecto sobre os volumes, que é fundamental para os designers de novo nível, é a importância de usá-los sabiamente em um nível. Isso se resume a dois pontos-chave:

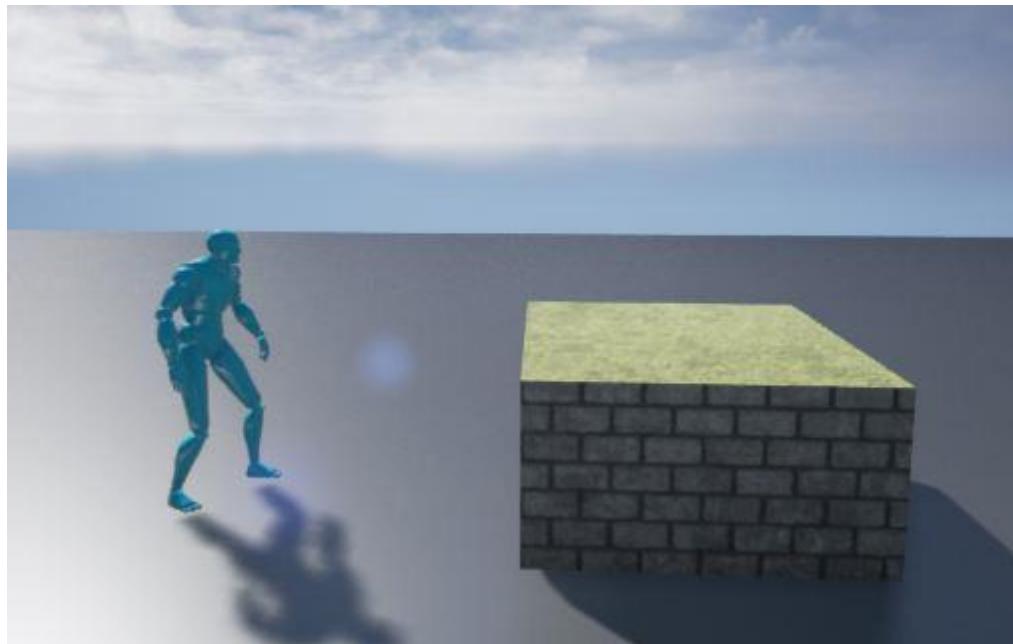
- Os volumes são invisíveis durante a jogabilidade; os jogadores não podem vê-los!
- Devido ao primeiro ponto, raramente você vai usar um volume por si só para qualquer efeito.

Como os volumes são invisíveis, os jogadores geralmente não sabem que inseriram um, não a menos que você, como o designer de nível, ajude-os um pouco. Os volumes devem ser normalmente considerados como componentes de um efeito maior, com algum outro elemento de nível, assim como devem agir como uma sugestão visual, para que se possa ter entendimento e completar o efeito.

Vejamos o exemplo a seguir:

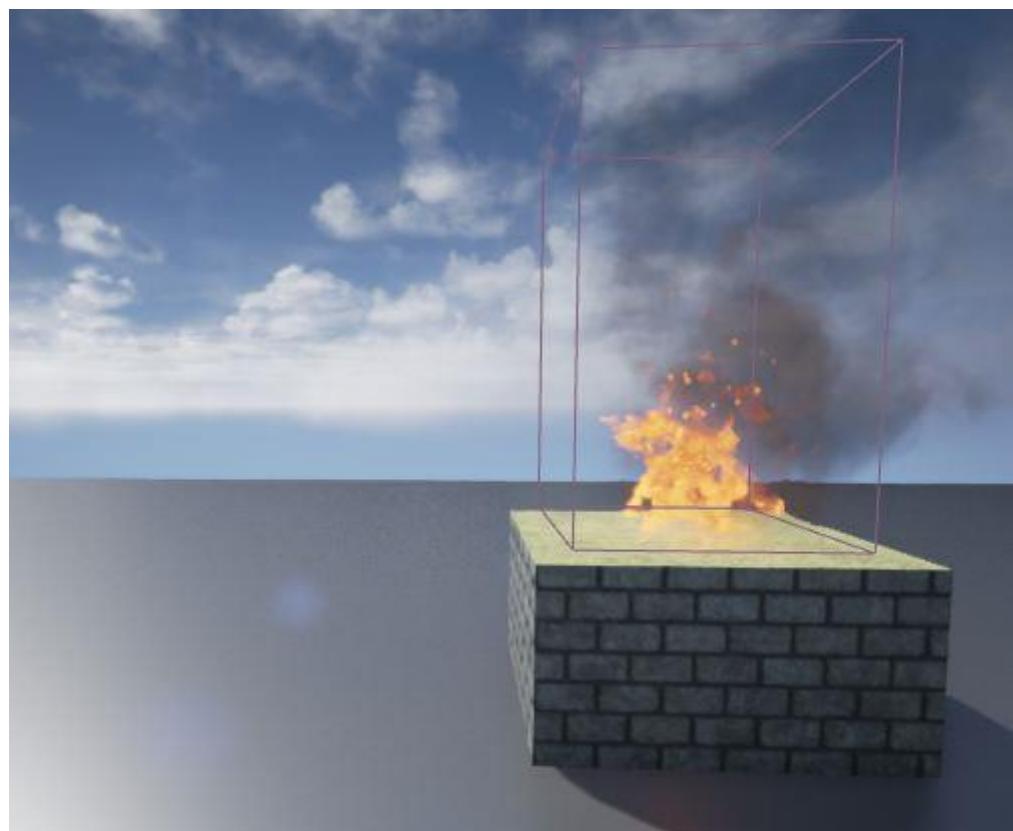


Colocação do volume de dano



O que o jogador verá

Caso o jogador subisse, ele não entenderia porque fica de repentinamente ferido assim que entra na área. No entanto, se houvesse um efeito de partícula ardente no meio do volume:



Colocação do volume

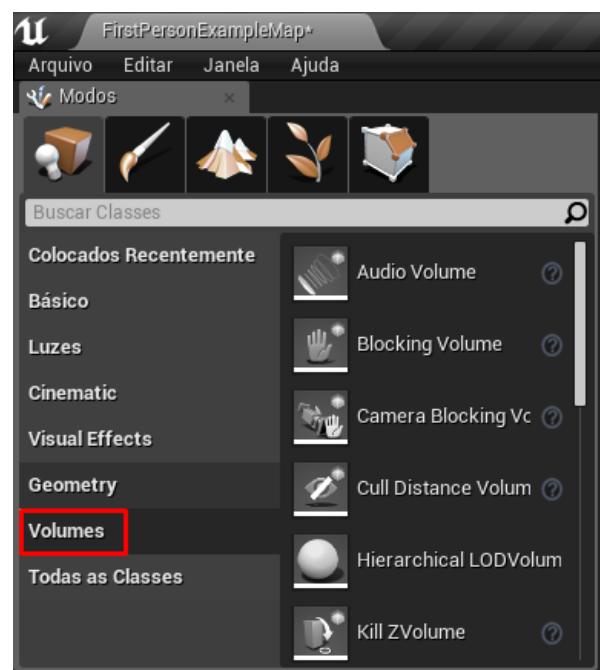


O que o jogador verá

Agora ele entenderia o porquê!

14.2. Criando volumes

Os volumes podem ser criados, selecionando na guia Volumes, do menu Modos, e arrastando-os e soltando-os na janela de nível:



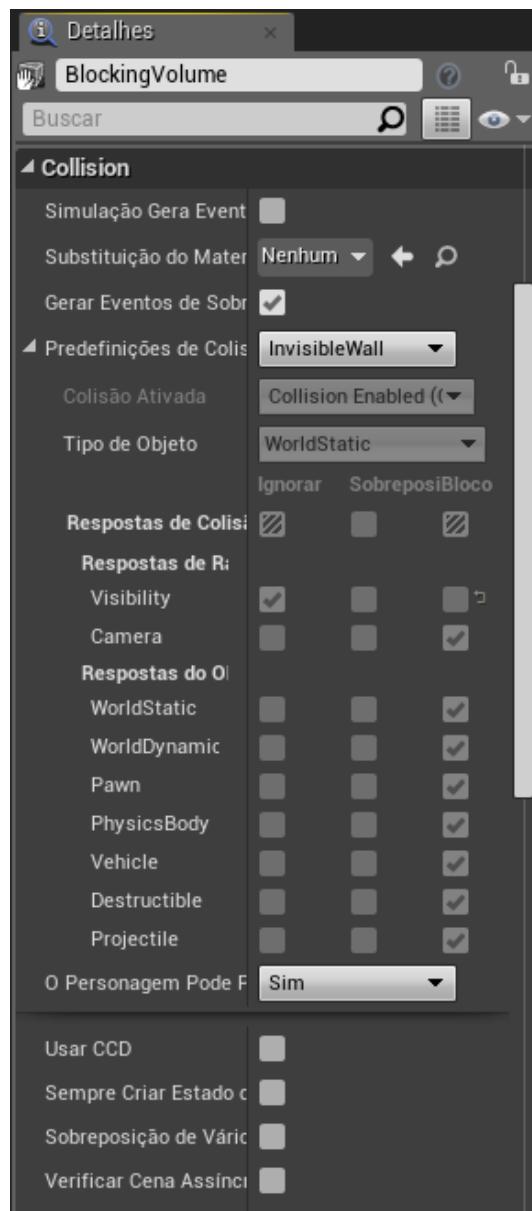
Uma vez que o volume foi colocado no nível, você pode então redimensionar e (ou) repositionar o volume, como você vê o ajuste.

14.3. Tipos de Volume

Abaixo veremos alguns dos volumes mais usados:

14.3.1. Volume de Bloqueio

Um volume de bloqueio serve como uma forma de colisão, através da qual os objetos não são destinados a passar. Ajustando os canais de colisão no volume (ilustrado abaixo), você pode controlar quais os tipos de objetos que serão permitidos a passar, bem como se quaisquer objetos também devem gerar eventos de sobreposição ao passarem pelo volume.



Os volumes de bloqueio podem ser usados no lugar de superfícies de colisão em malhas estáticas, particularmente no caso de paredes em estruturas. Quando usados dessa forma, eles geralmente têm algumas vantagens sobre superfícies de colisão de malha. A primeira é que eles são mais fáceis de calcular do que os modelos de colisão mais complexos. O segundo é que, uma vez que a natureza de um volume é saber quais objetos estão dentro dele, um volume de bloqueio pode ser mais útil ao colidir com objetos de física de alta velocidade.

14.3.2. Volume de bloqueio da câmara

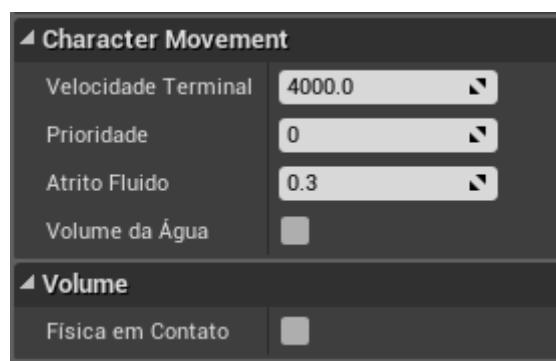
Os volumes de bloqueio da câmara são semelhantes em natureza a Volumes de bloqueio, na medida em que são utilizados para definir um espaço onde um ator de câmara não se destina a passar. Um exemplo disso seria em uma perspectiva de terceira pessoa, quando você quiser manter a câmera do seu jogador confinado ao seu espaço de jogo e não permitir que eles girem a câmera fora do mundo.

As propriedades do volume de bloqueio da câmara podem ser ajustadas da mesma forma que um volume de bloqueio, ajustando os seus canais de colisão.

14.3.3. Volume de Física

Volume de Física são volumes que possuem uma configuração física que afeta caracteres e que pode ser manipulada. Um uso comum para eles é no que se refere à criação de ambientes aquosos, onde o jogador precisa nadar. Mas você poderia facilmente usar um para criar uma área com baixa ou nenhuma gravidade e afetar a velocidade terminal dos objetos que passam através dele.

Existem algumas propriedades designadas para esse volume, que podem ser ajustadas a partir do painel detalhes, ilustradas abaixo.



Velocidade terminal: Determina a velocidade terminal dos peões, usando movimento de personagem ao cair.

Prioridade: determina qual Volume de Física tem precedência se eles se sobrepõem.

Fricção de fluidos: determina a quantidade de atrito aplicada pelo volume, à medida que os peões que usam Movimento de Personagem se movem através dele. Quanto maior esse valor, mais difícil será mover-se através do volume.

Volume de água: determina se o volume contém um fluido, como água.

Física em contato: determina se o ator é afetado pelo volume ao tocá-lo (por padrão, um Ator deve estar dentro do volume para que ele o afete).

14.3.4. Volume causador de dor

Os Volumes causadores de dor são realmente volumes de Física com a capacidade adicional de infligir dano aos jogadores. Esses são úteis quando você tem áreas óbvias de um nível que o jogador não deve ir, como um poço de lava, uma nuvem de gás tóxico, e assim por diante.

Além das propriedades que podem ser atribuídas a partir de um volume de Física, o volume causador de dor também tem seu próprio conjunto de opções específicas, delineado abaixo.



Causa da dor: se o volume atualmente causa dano ou não.

Danos por segundo: dano causado por segundo aos Atores no volume quando a Causa de Dor é ativada.

Tipo de dano: isso determina o tipo de dano causado ao Ator.

Intervalo de dor: essa é a quantidade de tempo, em segundos, entre os danos aplicados quando a Causa de dor está ativada.

Dor de entrada: quando ativada, causa dano ao entrar em contato com o volume mais o dano por segundo.

14.3.5. Kill Z Volume

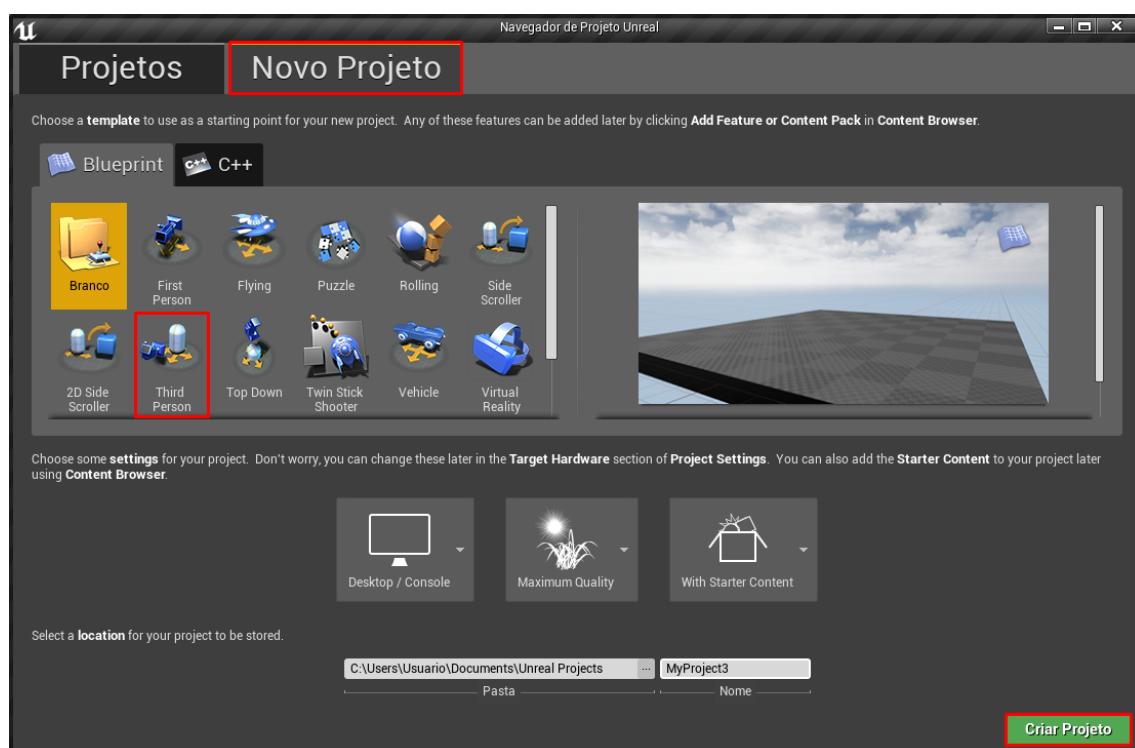
Mas o que é death zone?

Death zone (zona de morte) é um local que, quando o personagem entra em contato, ele morre. Esse recurso é muito utilizado nos games arcades, para definir buracos ou objetos que podem matar o personagem.

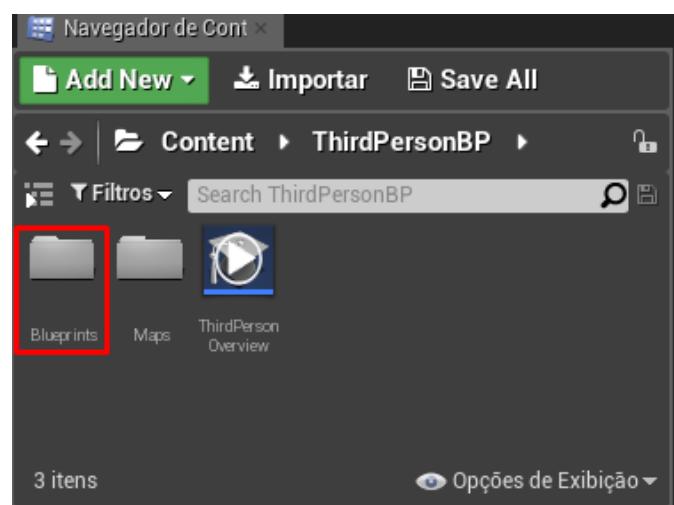
Dentro das configurações do death zone, podemos escolher o seu formato, deixando ele da forma que preferirmos, os dois mais utilizados são os de esfera e cubo.

Exercício de Conteúdo

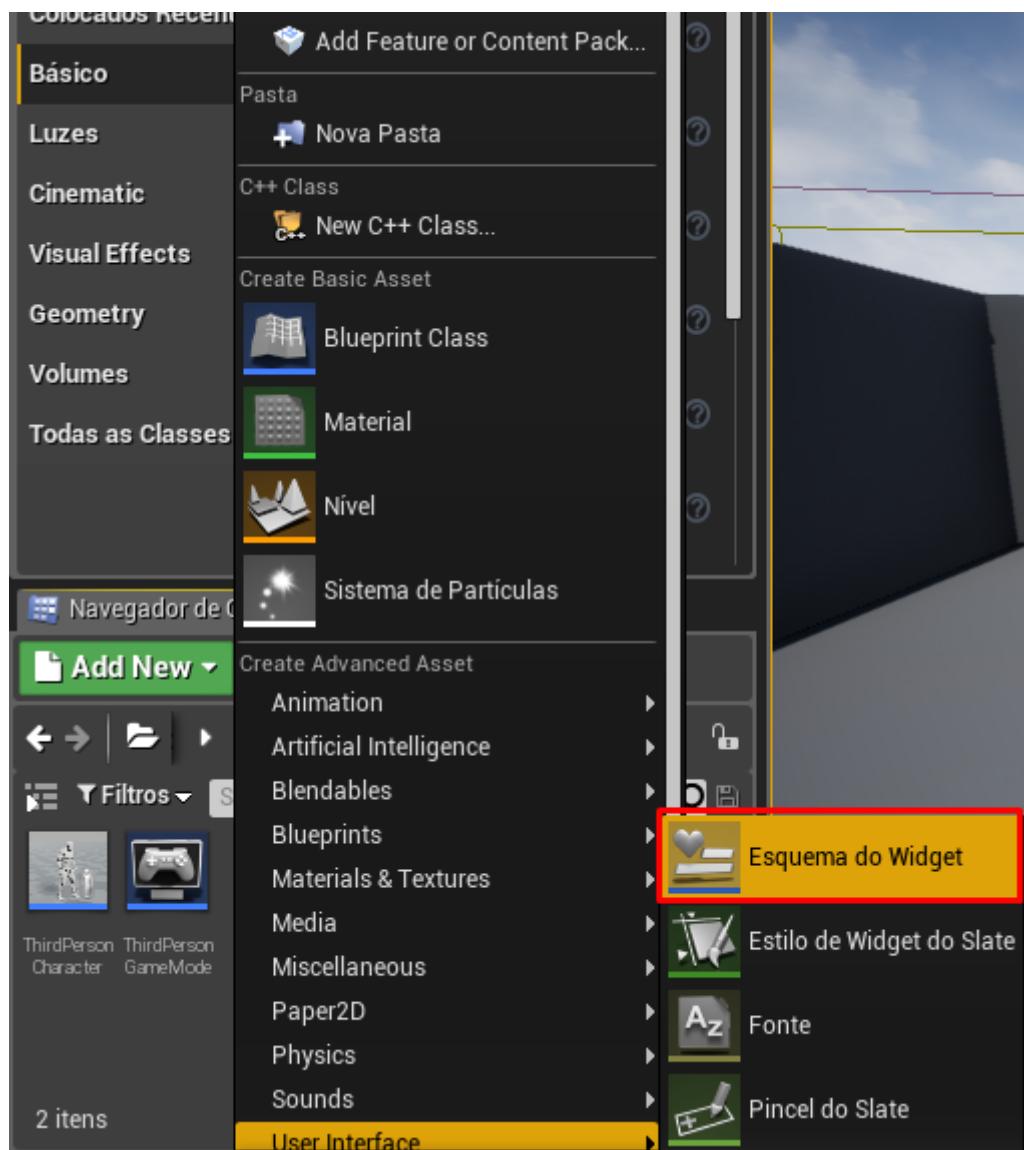
- 1- Abra a Unreal Engine 4
- 2- Selecione o projeto que você salvou com seu nome na aula 13 (sobre Áudio) e clique em abrir.
- 3- Nesse exercício, vamos criar uma barra de vida para o personagem, utilizando os blueprints e adicionar um volume causador de dor, para ver a barra de vida decaindo.
- 4- Para começar, abra a Unreal Engine e clique em novo projeto. Depois, escolha a opção ThirdPerson e, por último, clique em criar projeto:



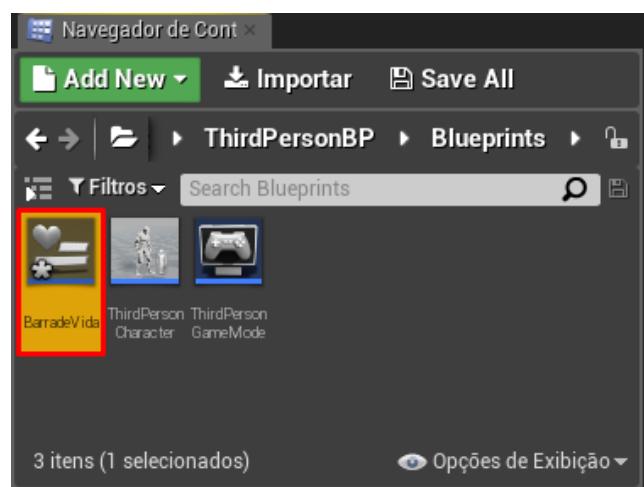
- 5- No navegador de conteúdo, entre na pasta blueprints:



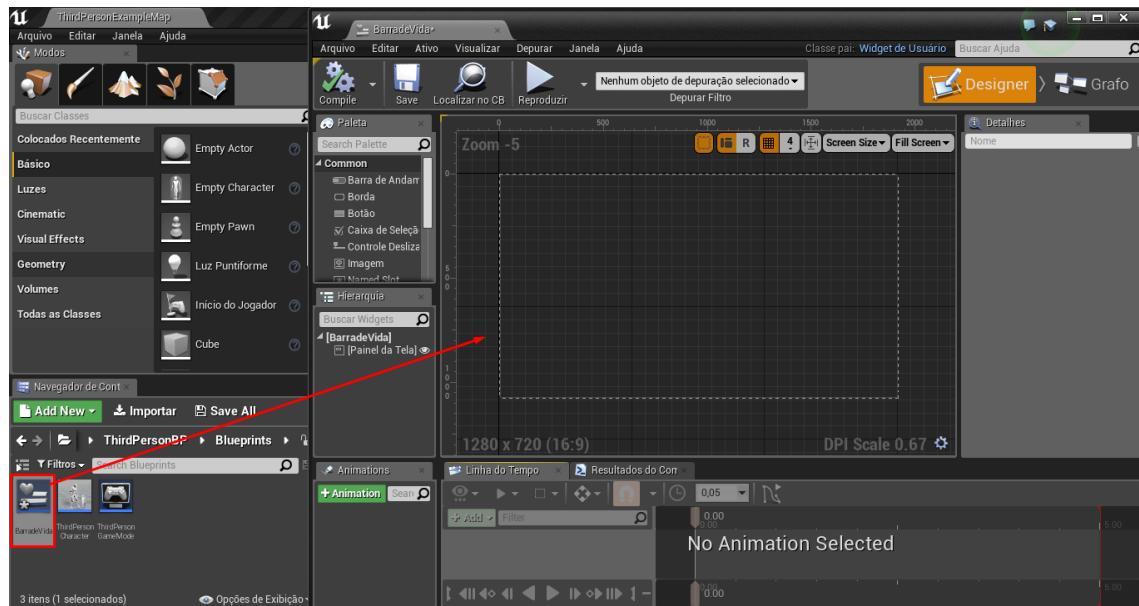
6- Clique em AddNew, vá até a opção User Interface e clique em Esquema do Widget:



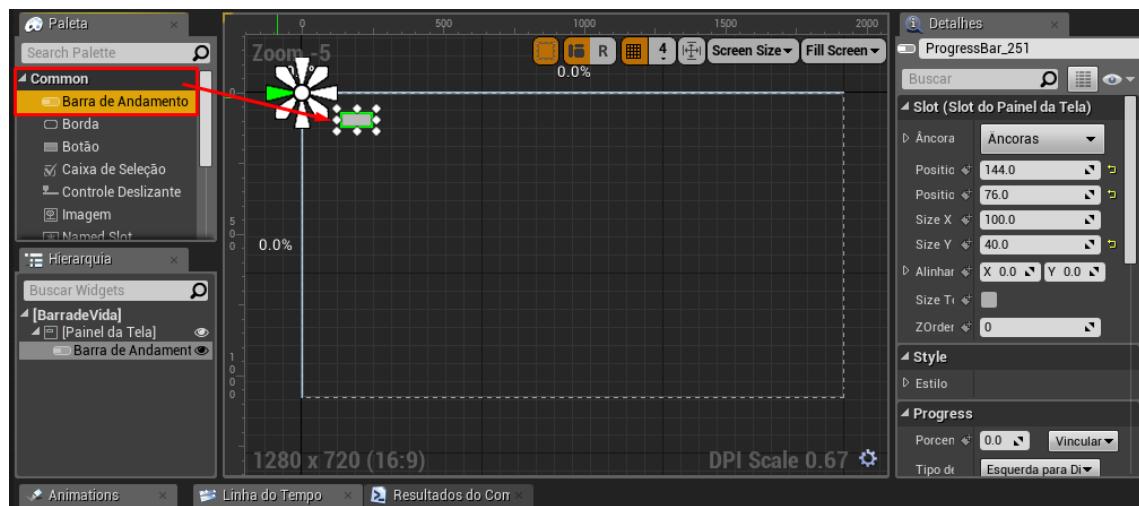
7- Dê o nome de BarradeVida:



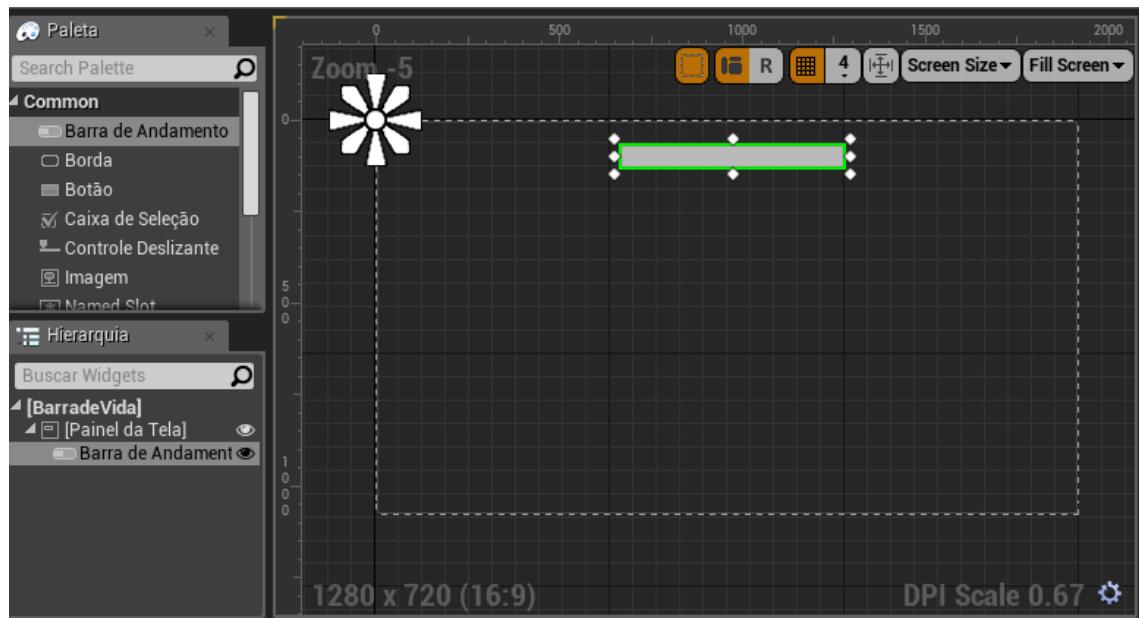
8- Dê um duplo clique, para abrir o editor do nosso widget:



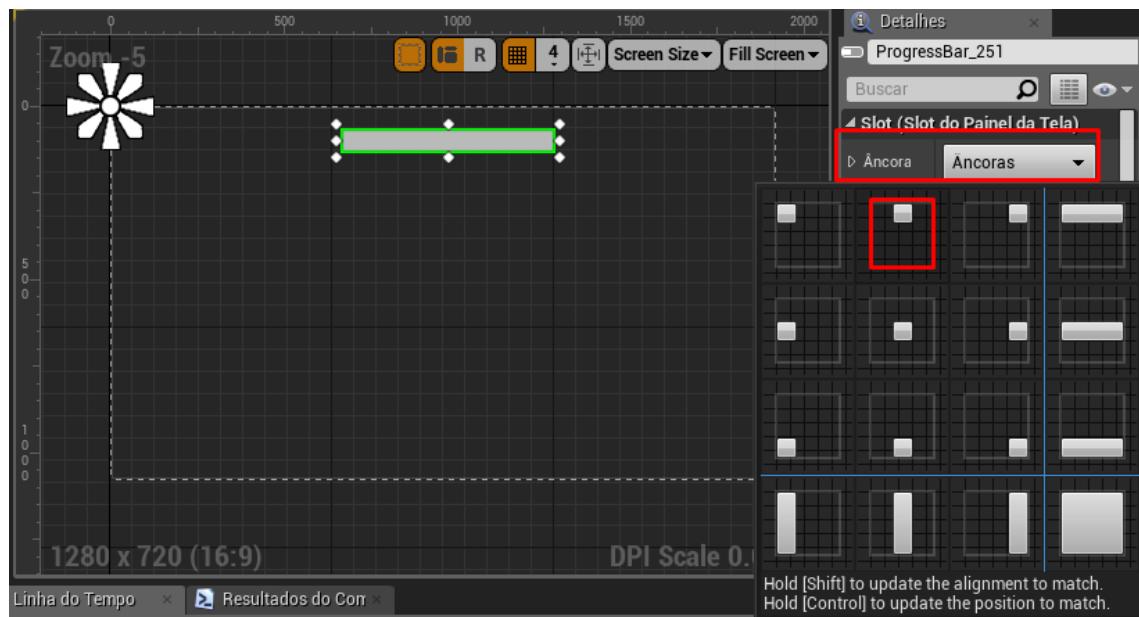
9- Esse retângulo pontilhado representa nossa tela. No menu lateral esquerdo, na aba Common, arraste para dentro da tela a opção “Barra de Andamento”:



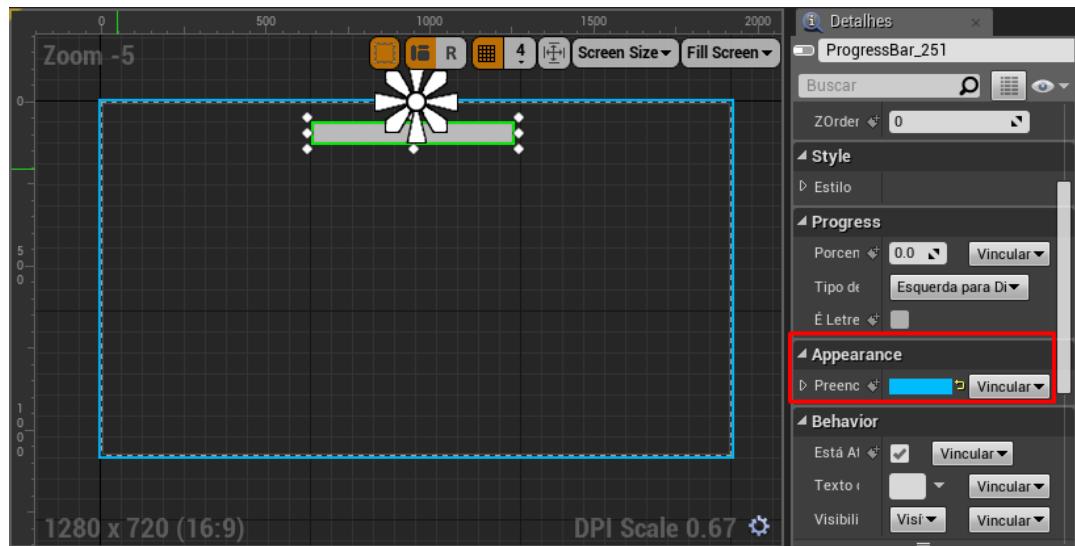
10- Ajuste, manualmente, a barra até ela ficar mais ou menos como a representação da imagem abaixo:



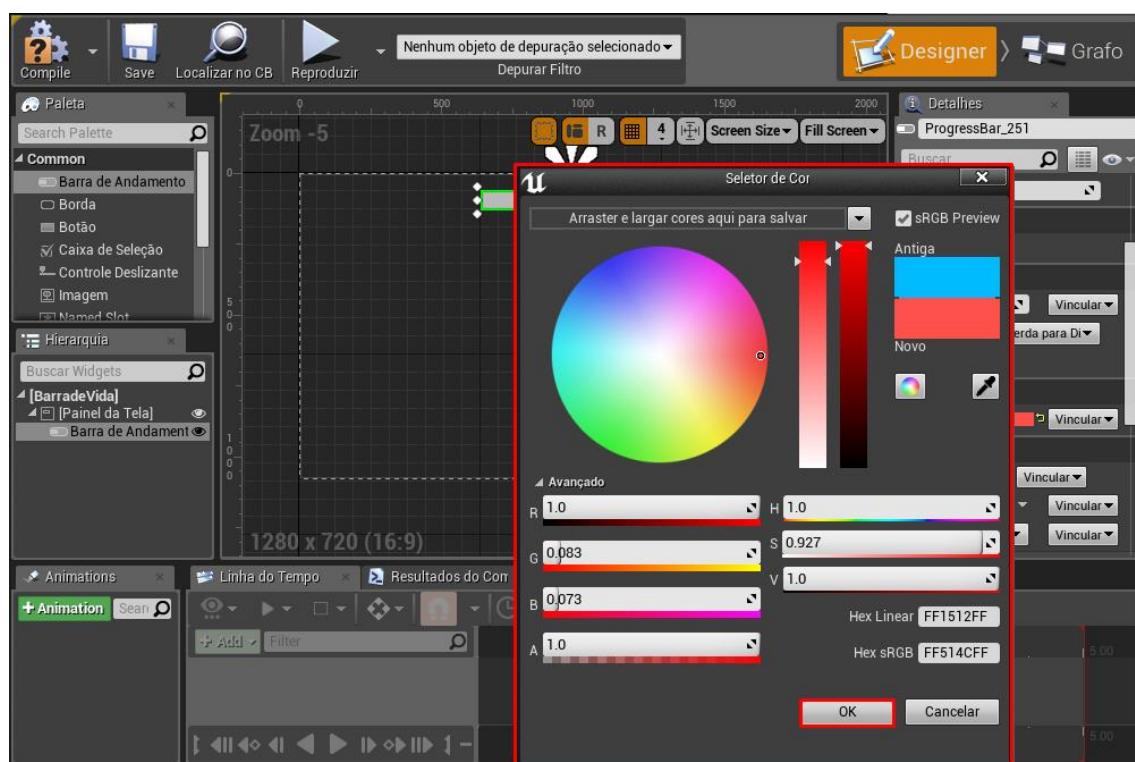
11- No menu detalhes, clique em âncora e escolha a opção selecionada abaixo:



12- Essa flor serve para ajudar no posicionamento do Widget. Ainda no menu detalhes, desça a barra lateral até chegar na aba appearance e dê um duplo clique sobre a cor azul:



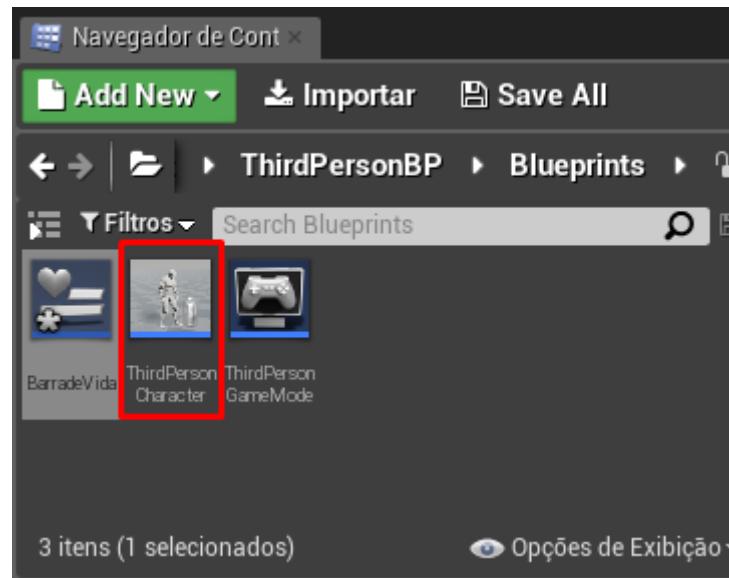
13- Vai abrir uma nova janela para escolhermos a cor da barra de vida. Selecione a cor vermelha e clique em ok:



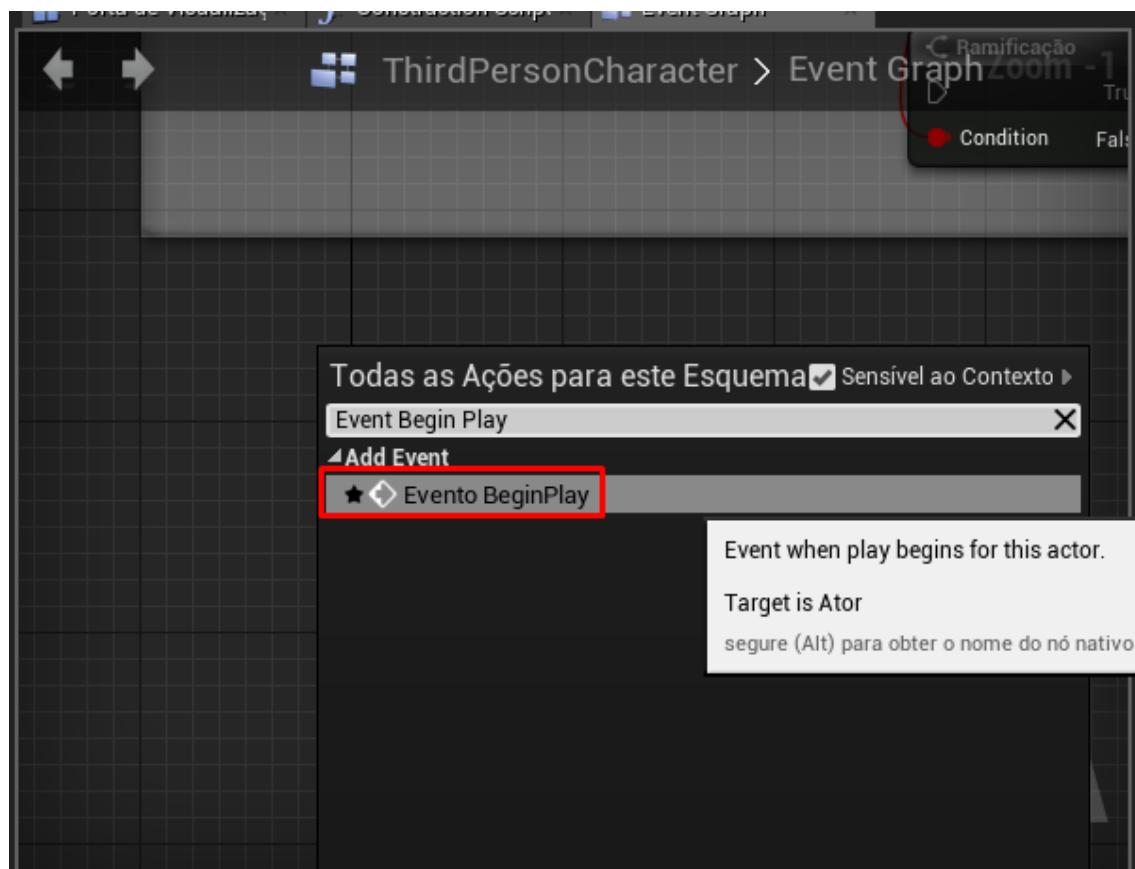
14- Clique em Compile e clique em save:



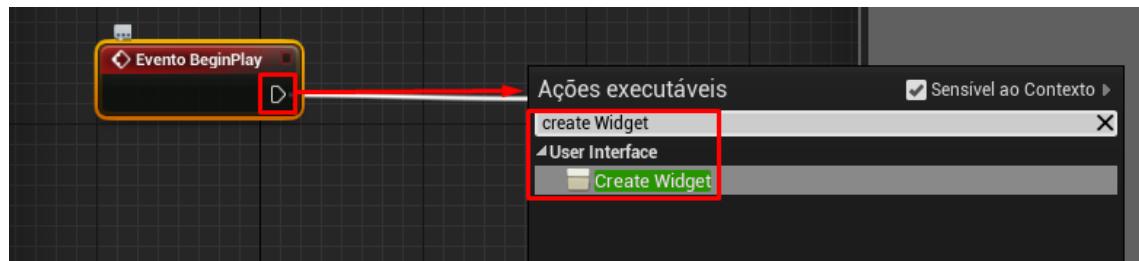
15- Com nossa barra posicionada, precisamos fazê-la aparecer na tela do jogador. Para isso, abra a blueprint do personagem, dando um duplo clique sobre ela:



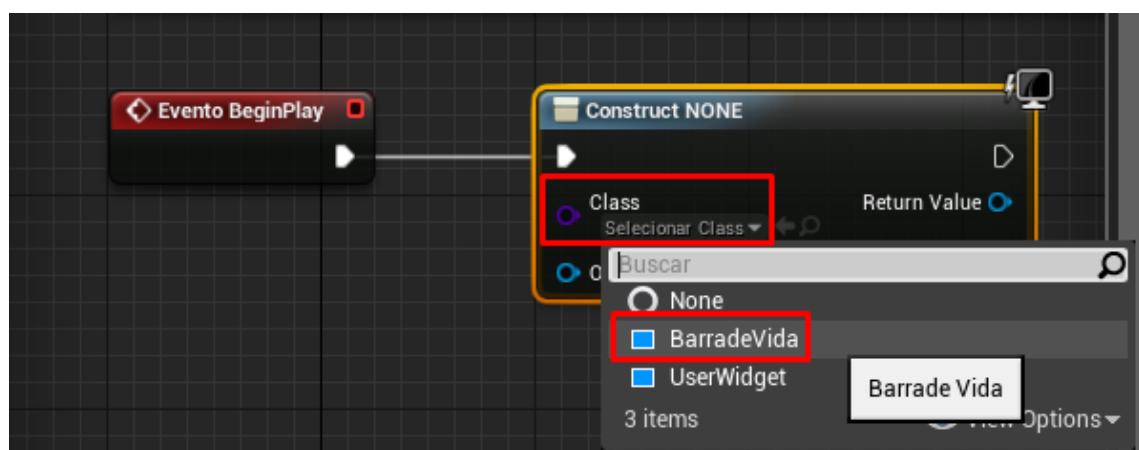
16- Nessa nova janela que abrir, você vai reparar que tem todas as ações que o personagem faz, criadas através do blueprint. Mas abaixo, numa área vazia, clique com o botão direito do mouse e pesquise Event Begin Play:



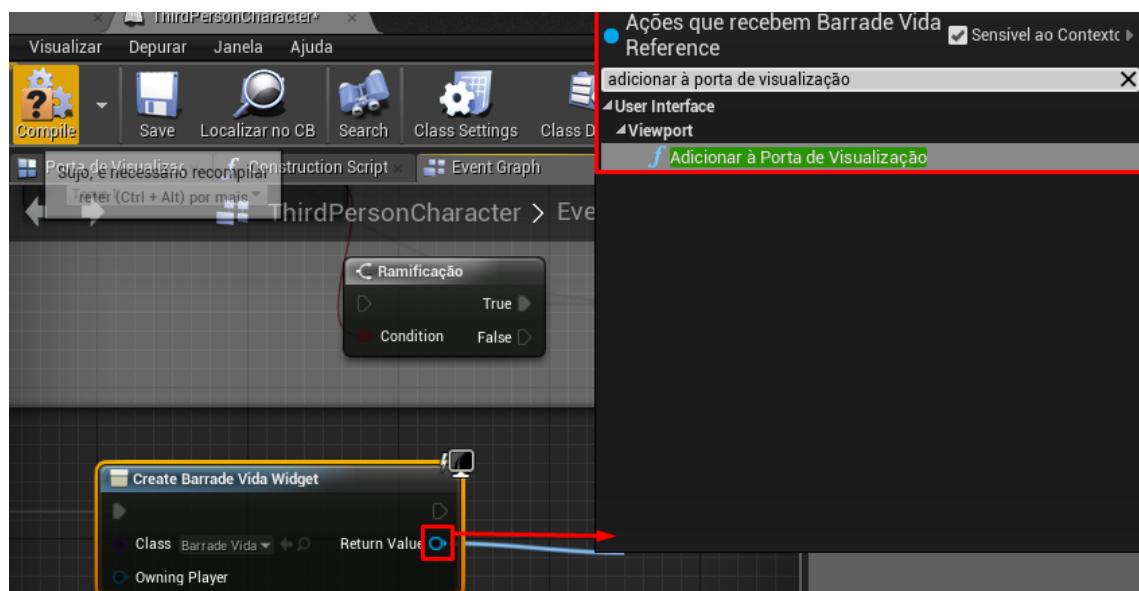
17- Clique no canal do nosso evento e arraste para o lado, vai abrir novamente a janela de ações, digite Create Widget:



18- Em Class, selecione a BarradeVida:



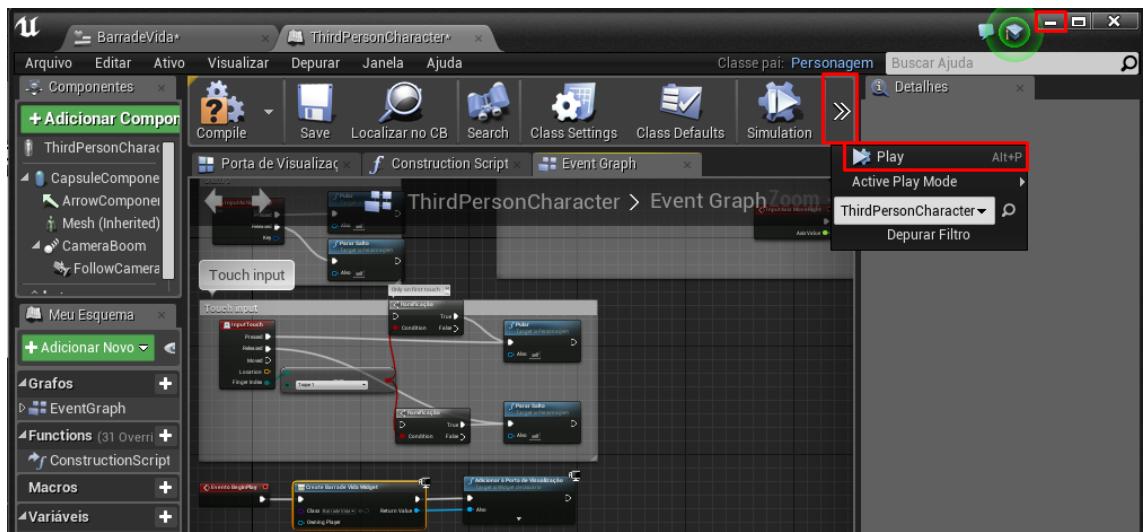
19- Agora no canal azul, Return Value, clique e arraste para adicionarmos outra ação.
Quando abrir a nova janela, digite Adicionar à Porta de Visualização:



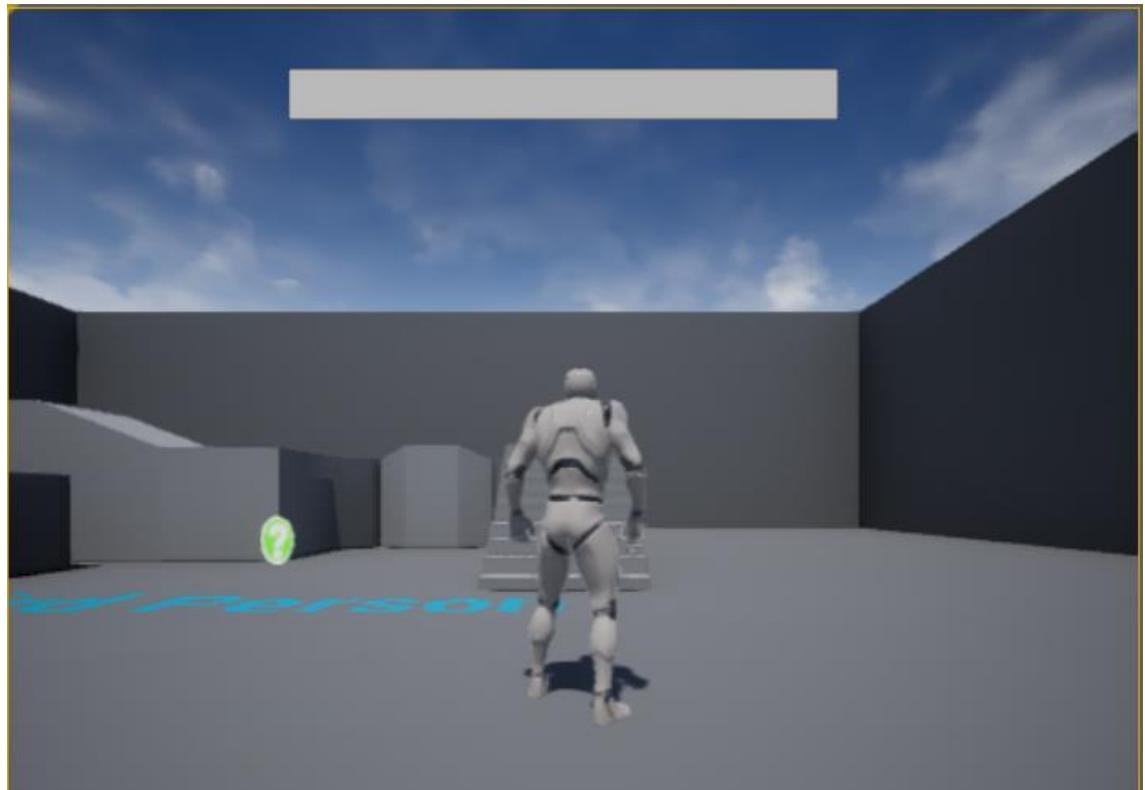
20- Clique em compile e depois em save:



21- Vamos ver se está aparecendo na nossa tela agora. Clique em Play e depois minimize o blueprint do personagem:



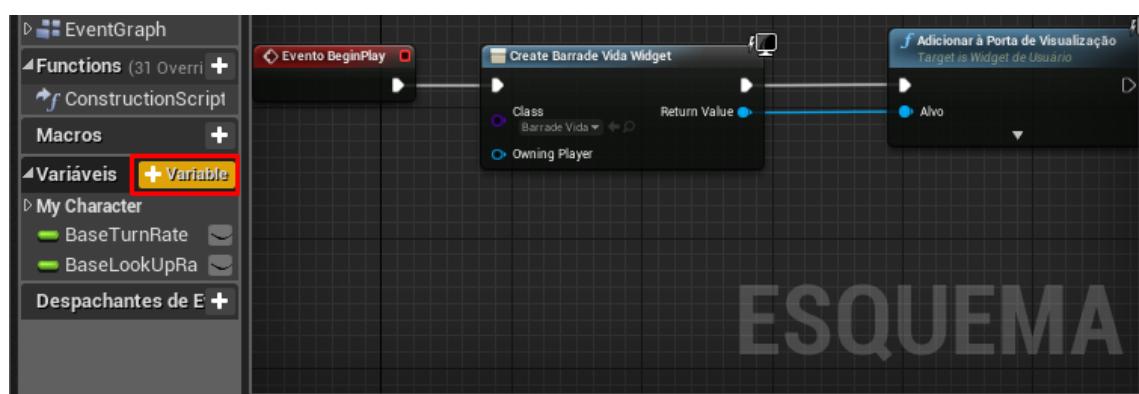
22- Perceba que deu certo:



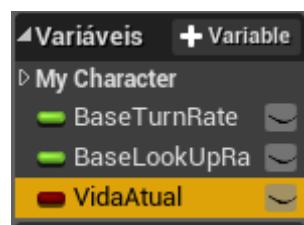
23- Repare que a barra está cinza. Para resolver isso, precisamos criar algumas variáveis, volte para a blueprint do personagem e dê um stop na simulação:



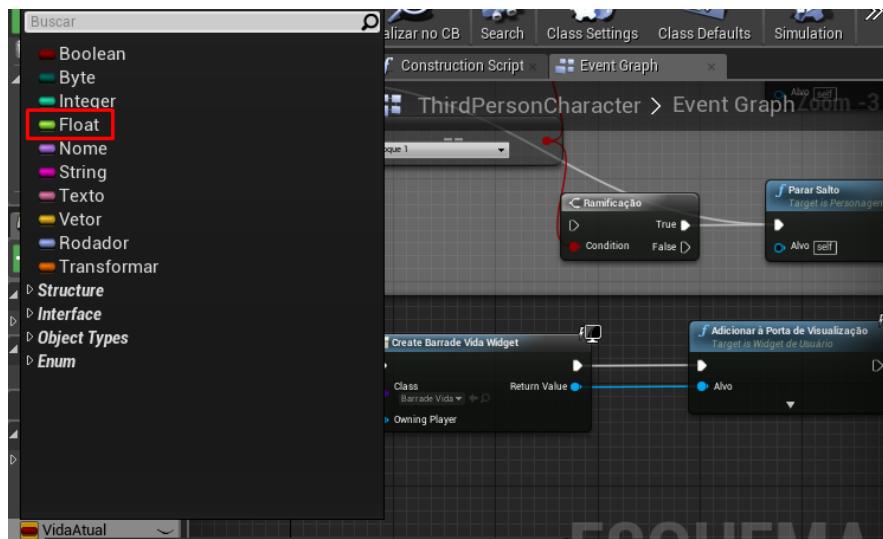
24- No menu lateral esquerdo, onde diz Variáveis, clique em +:



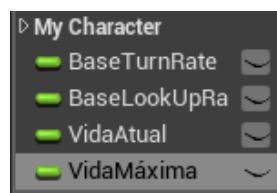
25- Nomeie a Variável de VidaAtual:



26- Clique sobre o ícone vermelho, para que possamos mudar o tipo de variável e escolha a opção Float:



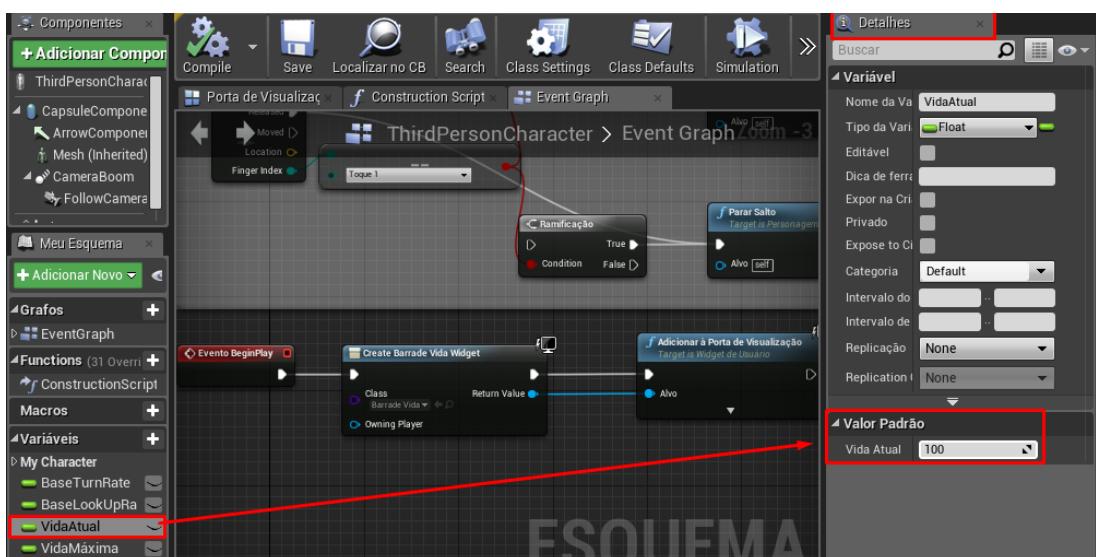
27- Adicione mais uma variável e a nomeie de VidaMáxima:



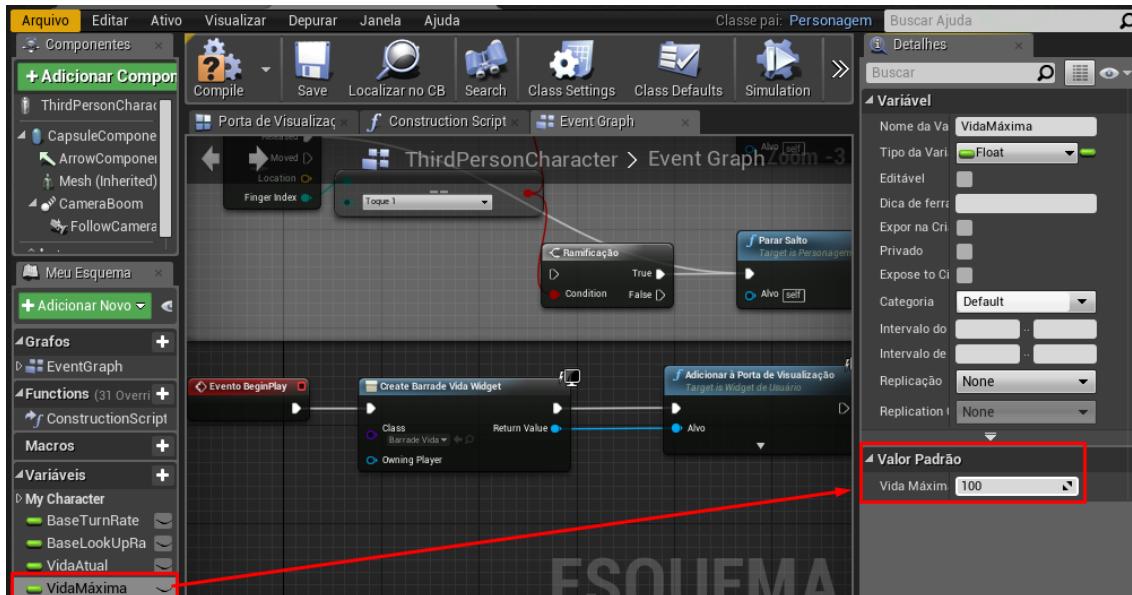
28- Clique em compile e depois em save:



29- Agora, selecione a variável VidaAtual e, no menu Detalhes, vamos mudar seu valor padrão, digitando 100:



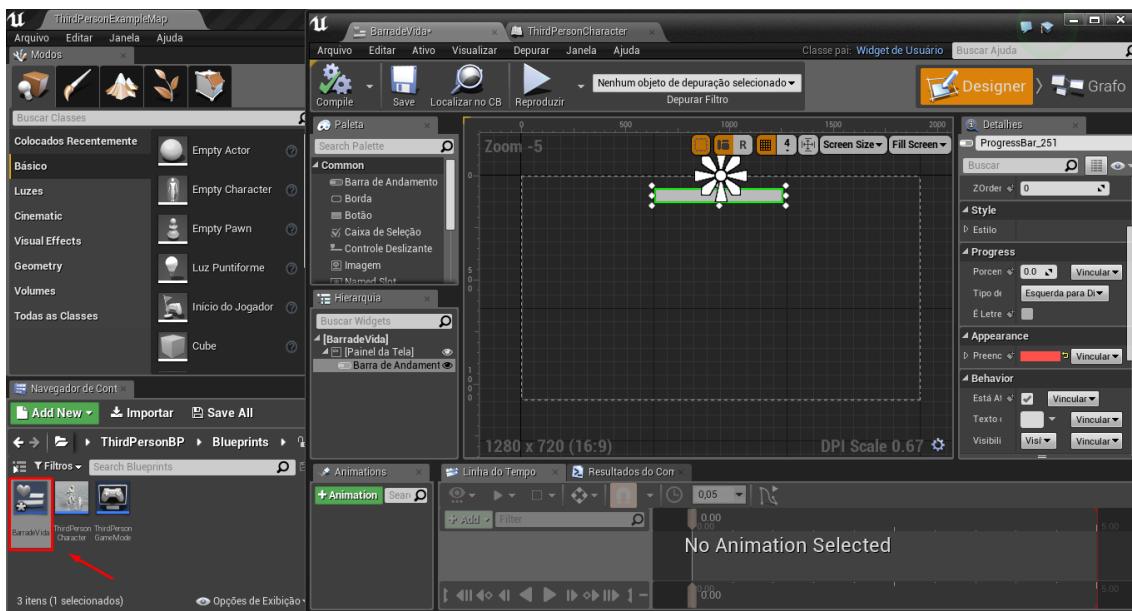
30- Faça a mesma coisa com a variável VidaMáxima:



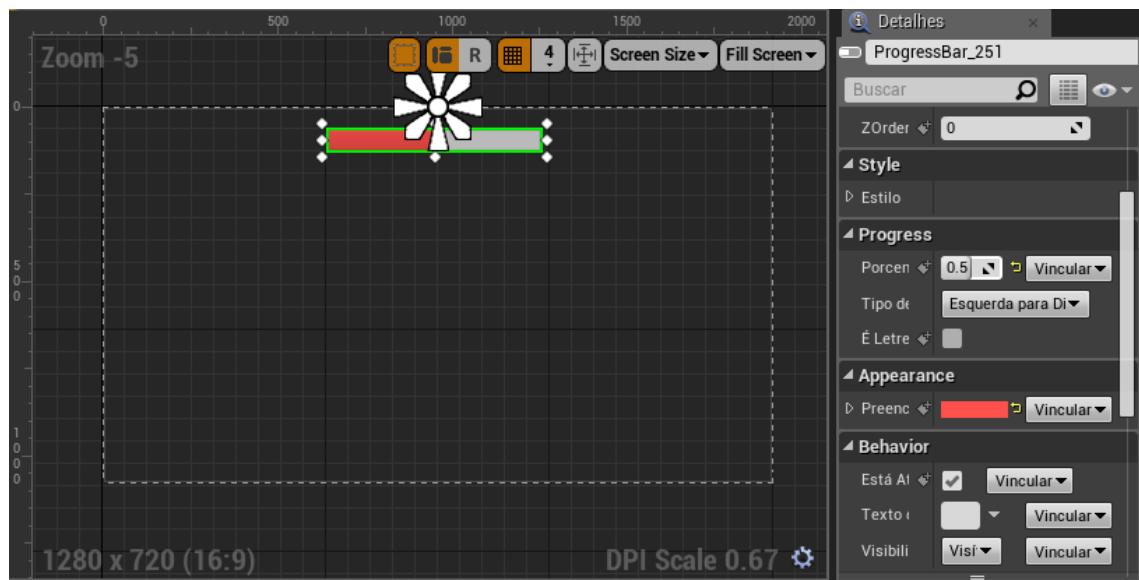
31- Clique em compile e de save novamente:



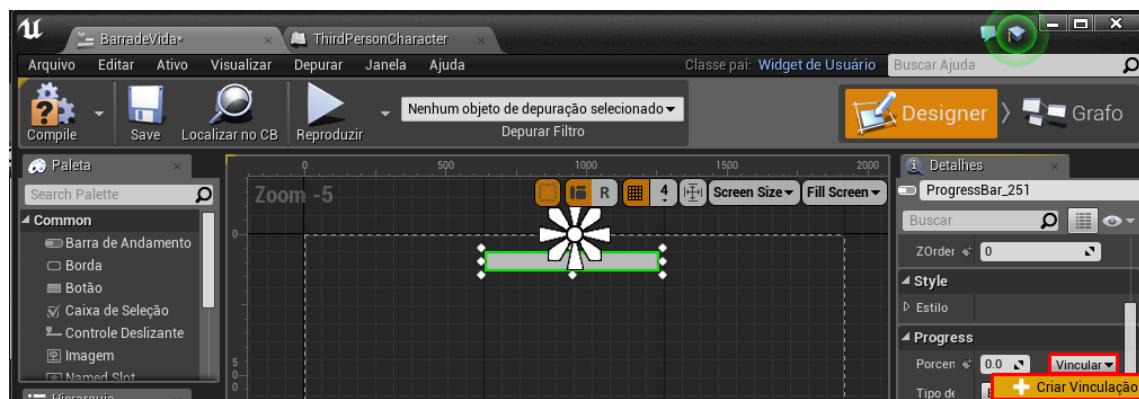
32- Agora que temos as variáveis, precisamos adicioná-las a nossa barra de vida. Dê um duplo clique sobre a widget BarradeVida, para que se possa abrir o editor:



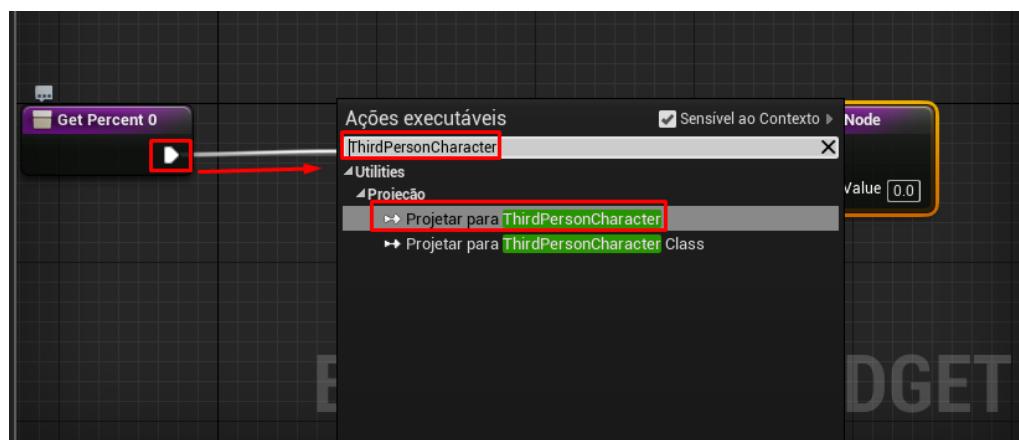
33- Em detalhes, na aba progress, tem um valor de 0. Se você aumentar esse valor, vai perceber que a barra de vida enche, por exemplo, coloque 0.5 no valor:



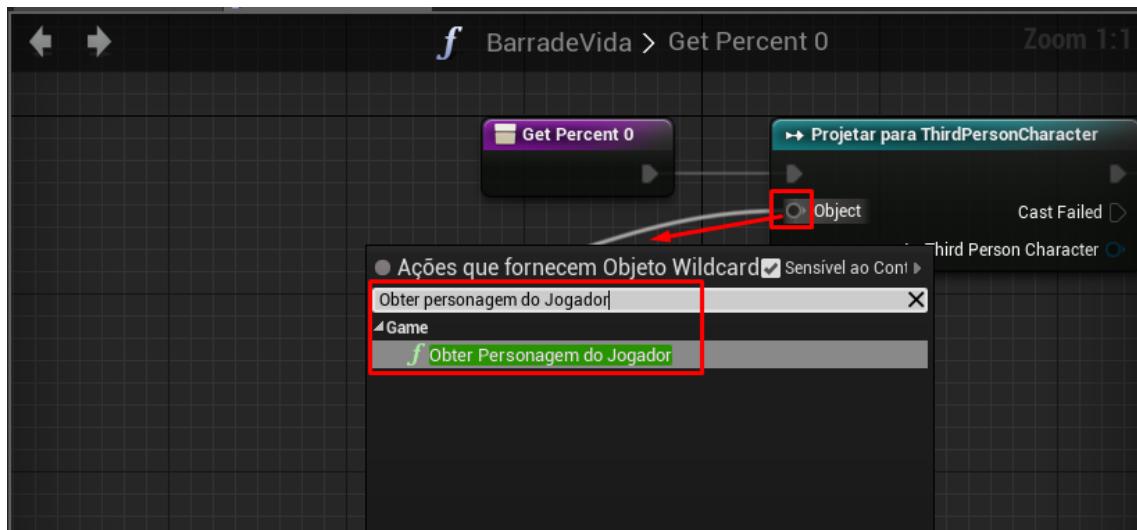
34- Essa é a variável que temos que manipular. Clique em vincular ao lado e, depois, clique em criar vinculação:



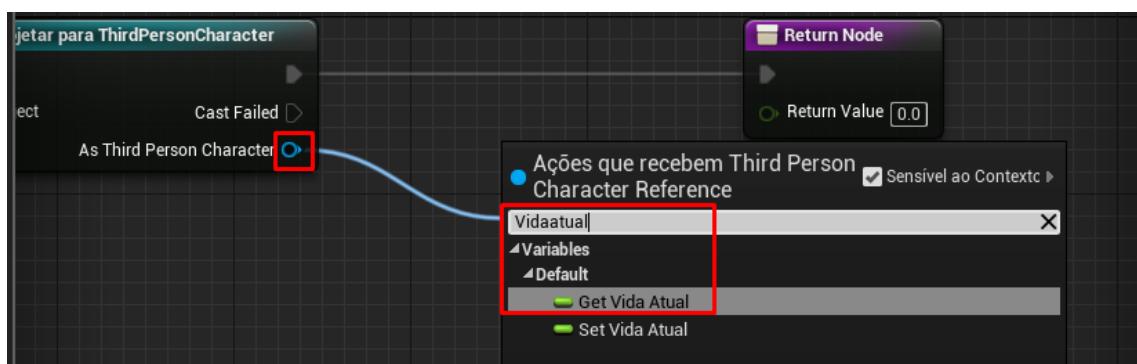
35- Uma nova janela se abrirá. Aqui, adicionaremos as ações para fazer a barra de vida se movimentar. Para começar, clique no canal do Get Percent e arraste para o lado. Pesquise por ThirdPersonCharacter e escolha a primeira opção:



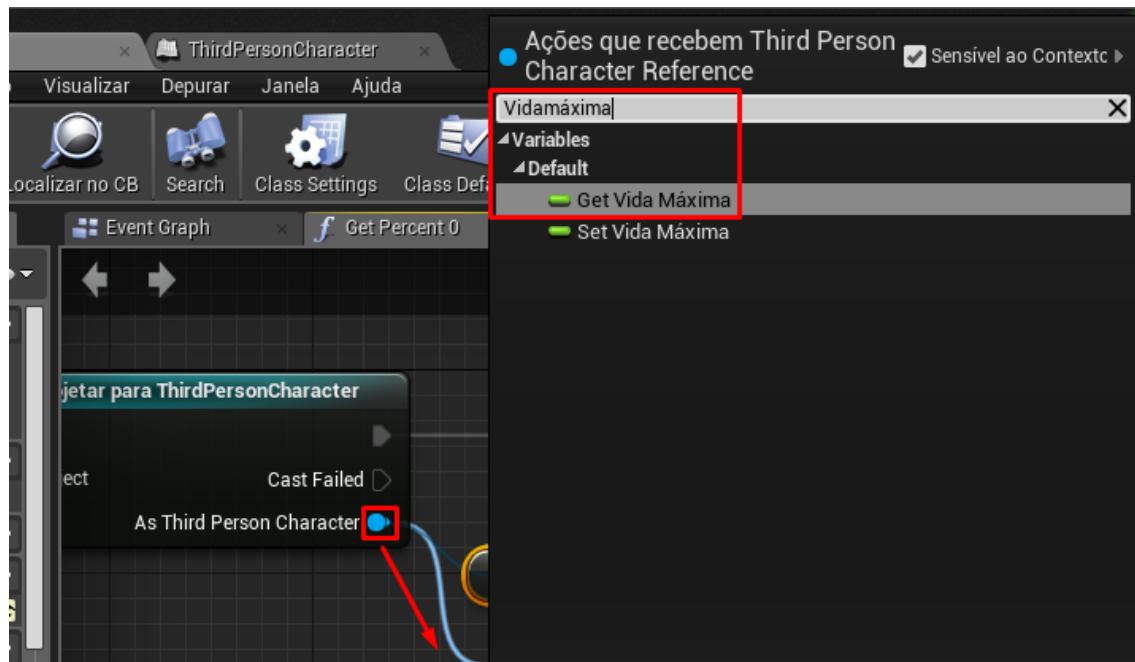
36- Na nova função adicionada, clique no canal Object e arraste para o lado. Pesquise por Obter Personagem do Jogador:



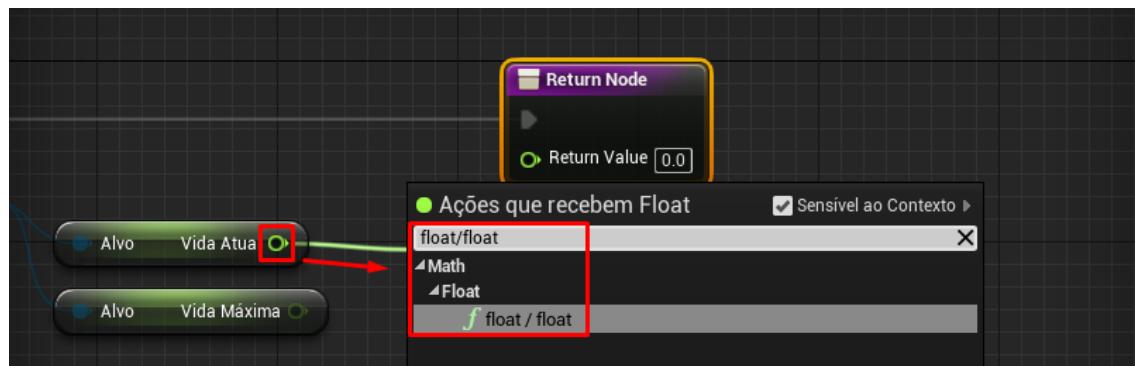
37- Agora, clique no canal as Third Person Character e arraste para adicionarmos as variáveis. Pesquise por VidaAtual e selecione a primeira opção:



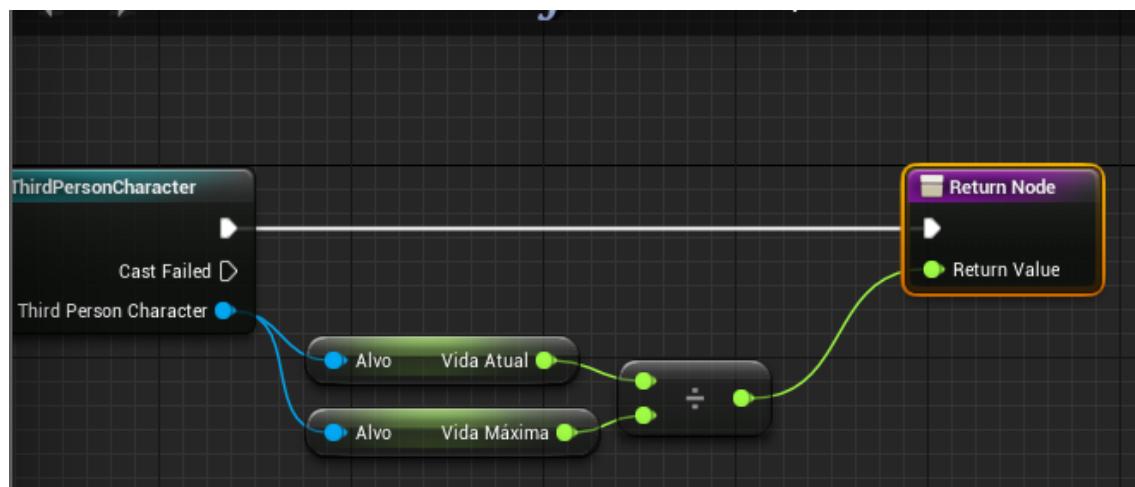
38- Repita o processo anterior, mas, dessa vez, pesquise por VidaMáxima:



39- Agora, em Vida Atual, clique sobre o canal verde e arraste para o lado. Pesquise por float/float (A barra / significa divisão, ou seja, uma variável dividida pela outra):



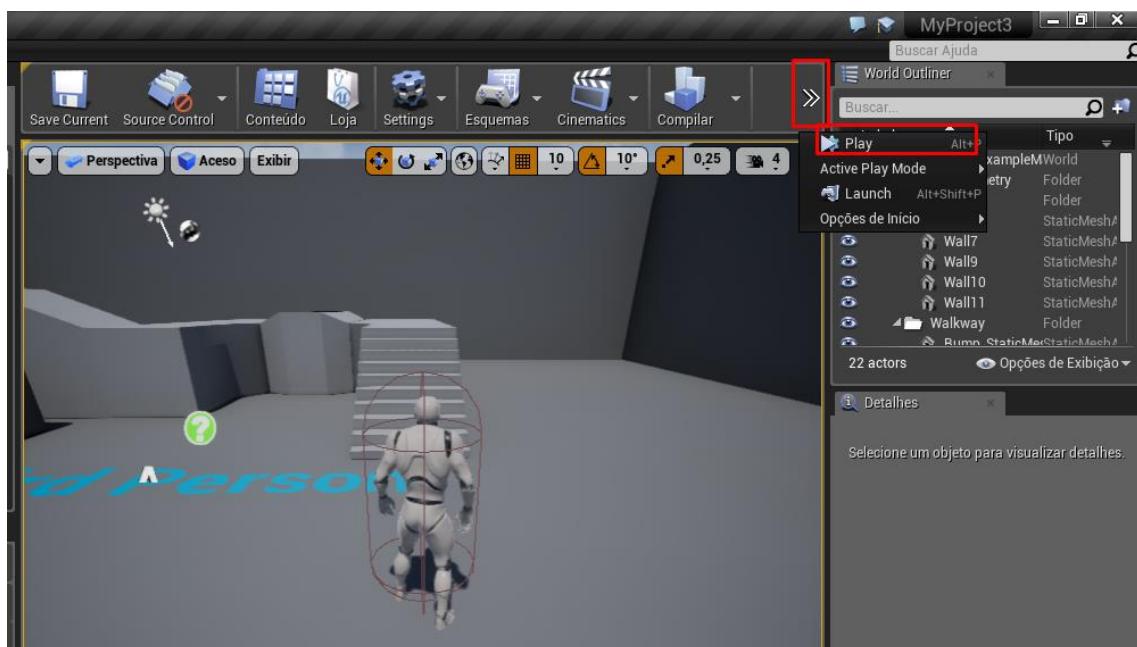
40- Ligue os canais, de acordo com a imagem abaixo:



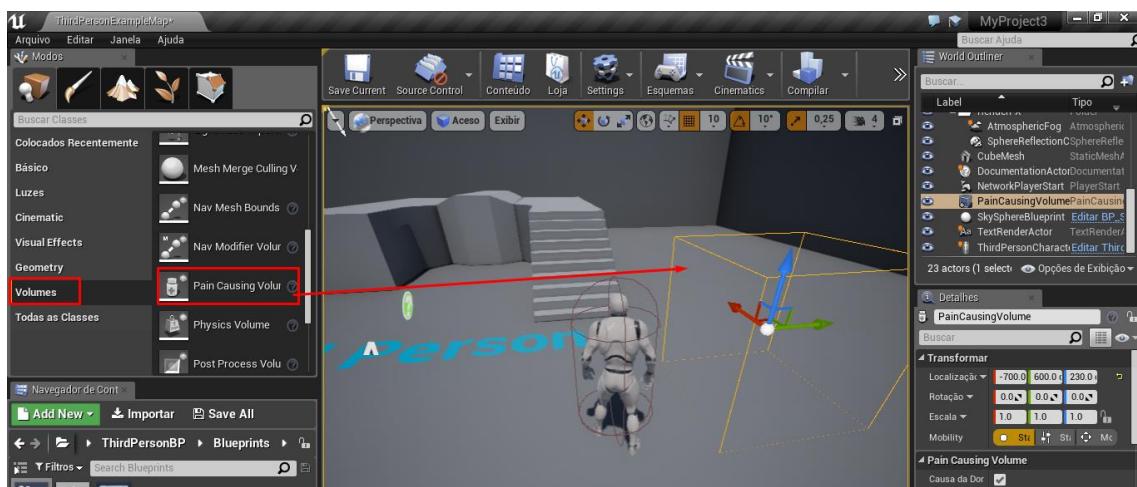
41- Clique em Compile e depois em save:



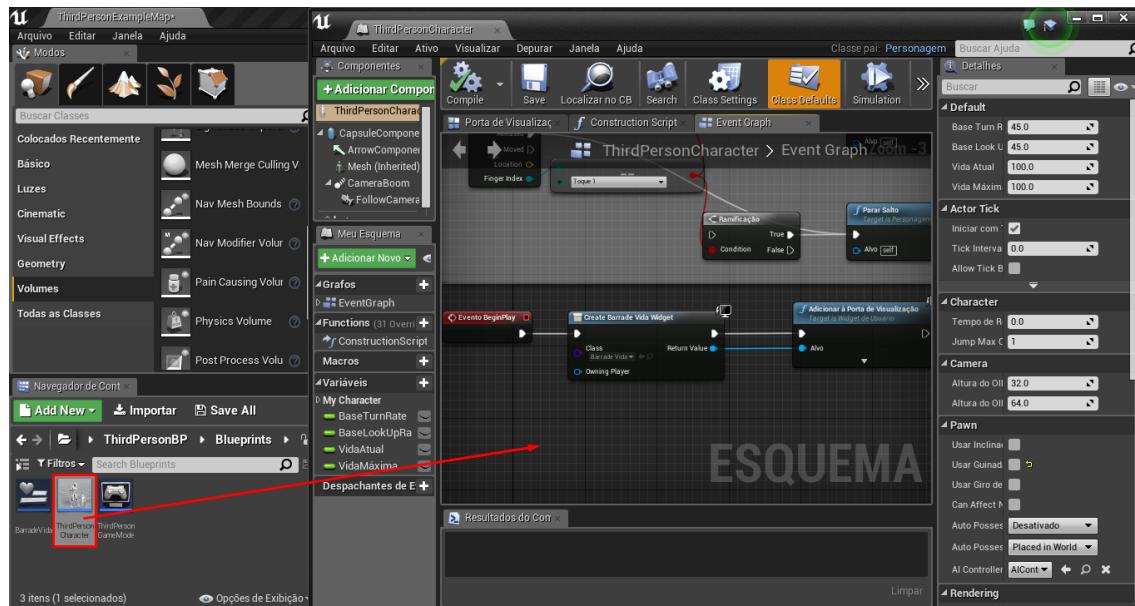
42- Minimize o editor e clique em play:



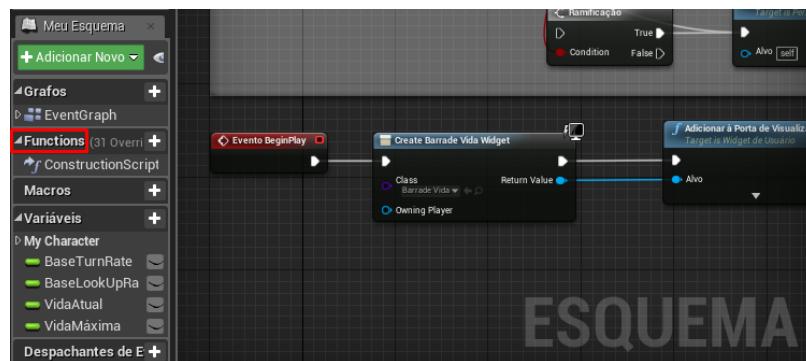
43- Ok, deu tudo certo com nossa barra de vida. Agora, precisamos adicionar alguma coisa para dar dano, que nesse caso será o Pain Causing Volume. Clique em volume e depois arraste o Pain Causing Volume para dentro da cena:



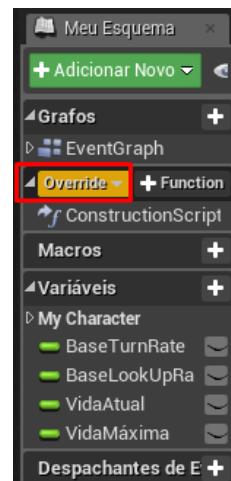
44- Para que esse volume funcione sobre nossa barra de vida, precisamos adicionar mais alguns eventos ao nosso personagem. Então, abra novamente a blueprint do personagem:



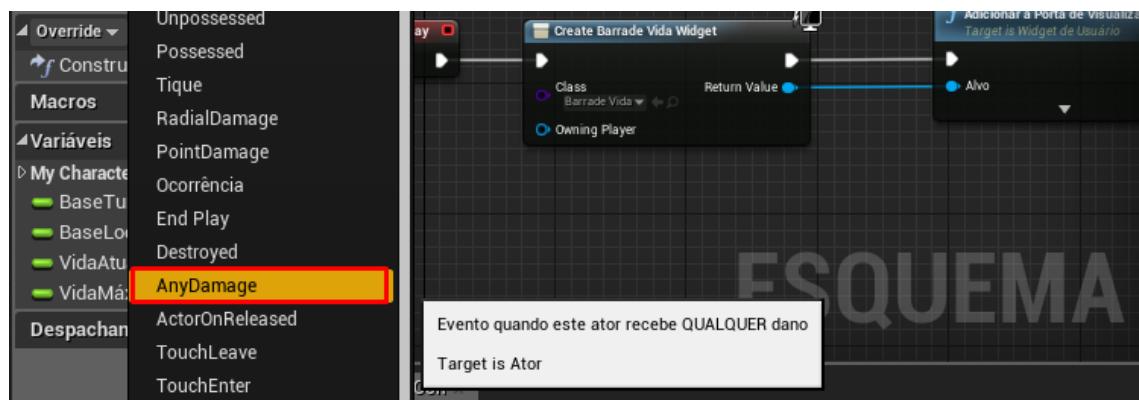
45- No menu lateral “Meu esquema”, coloque o mouse sobre a opção functions:



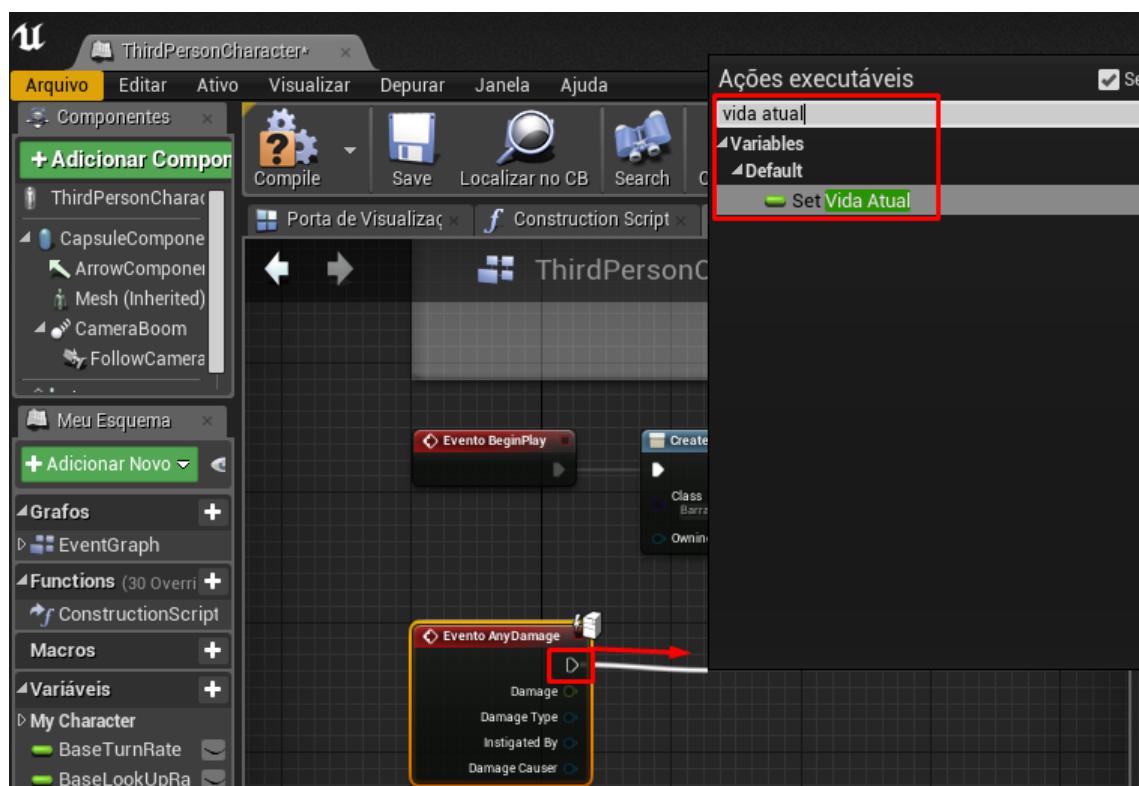
46- Ele deverá mudar para Override. Quando isso ocorrer, dê um clique sobre ele:



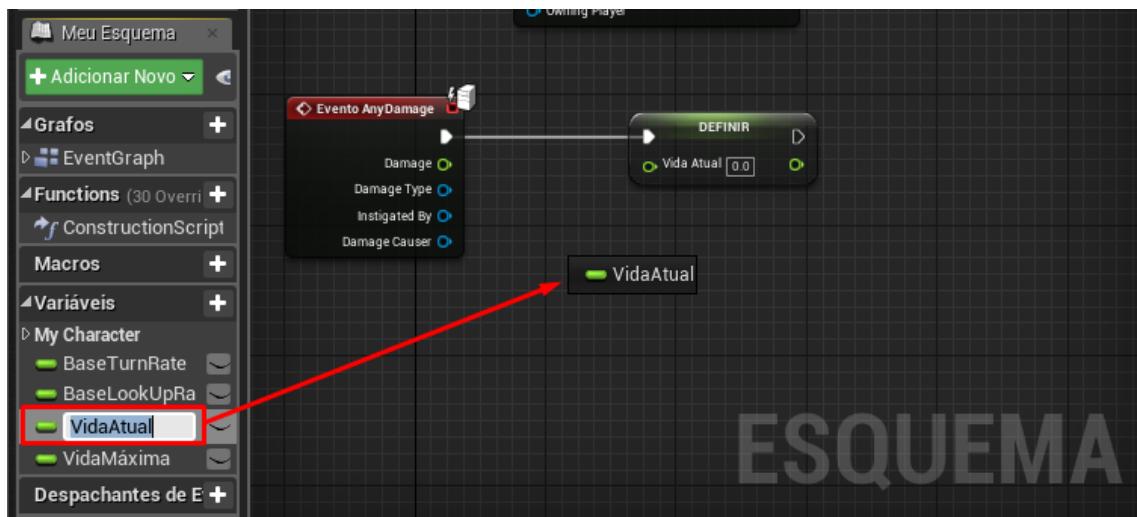
47- Abrirá uma aba com várias funções, escolha a opção AnyDamage, que é o evento para quando o ator receber qualquer tipo de dano:



48- Clique sobre o primeiro canal do evento que foi recém adicionado e arraste para o lado. Pesquise por Vida atual:



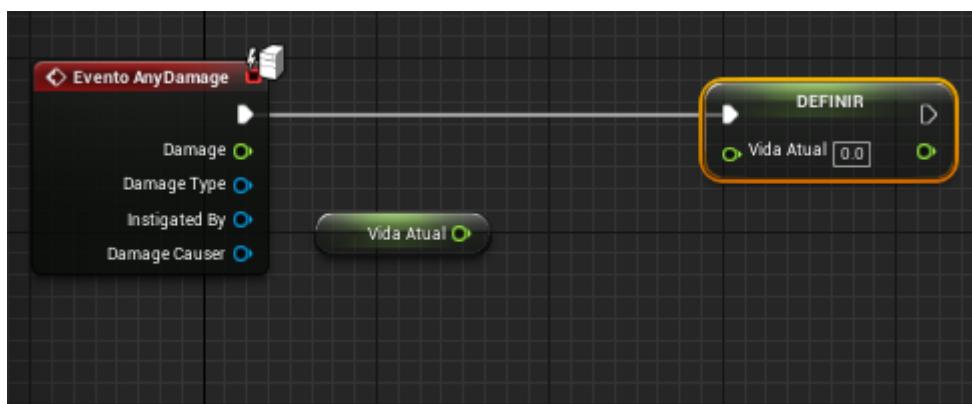
49- No menu lateral esquerdo, na aba Variáveis, clique em VidaAtual e arraste para dentro do seu esquema:



50- Quando você soltar, aparecerá o seguinte menu:



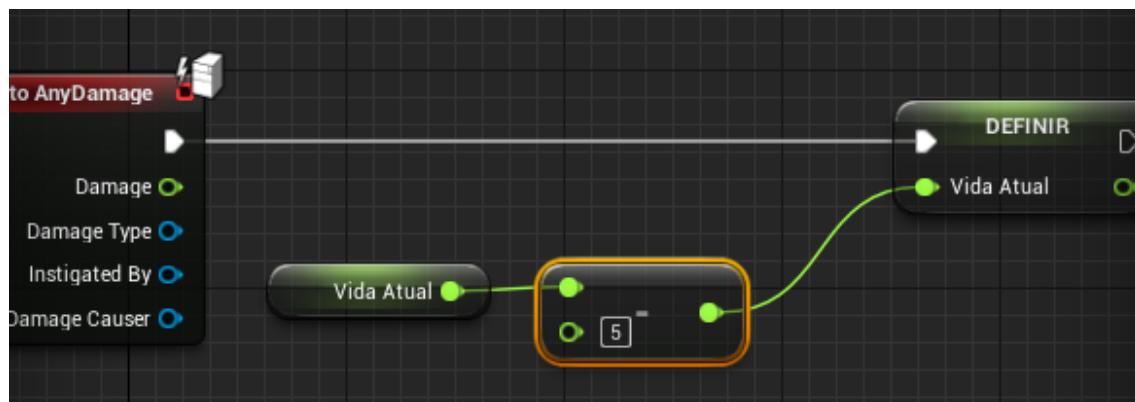
51- Clique em Adquirir:



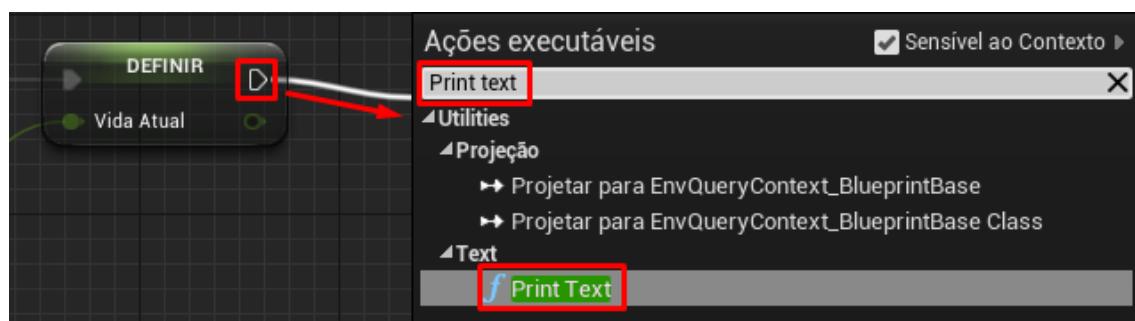
52- Agora, clique no canal dessa nova ação e arraste para o lado. Pesquise por float – float (o traço é referente a operação matemática subtração) e clique para selecionar a seguinte opção:



53- Na nova ação, podemos adicionar um valor de dano. Digite 5 e ligue o outro canal, de acordo com a imagem abaixo:



54- E para finalizar essa parte, clique e arraste no canal superior da ação DEFINIR, para adicionar um texto à tela quando você tomar dano. Pesquise por Print Text e selecione a opção abaixo:



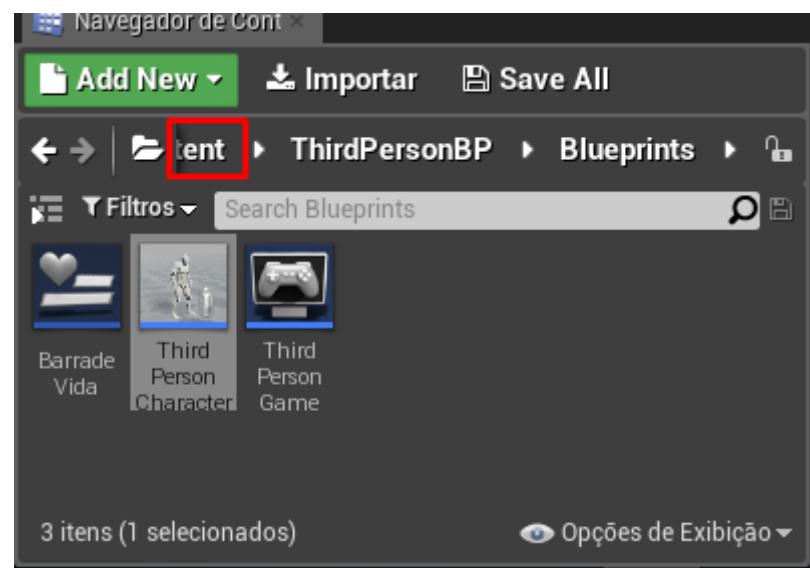
55- Clique sobre onde diz “hello” e adicione o valor 5, que é o dano que vamos tomar ao entrar na área:



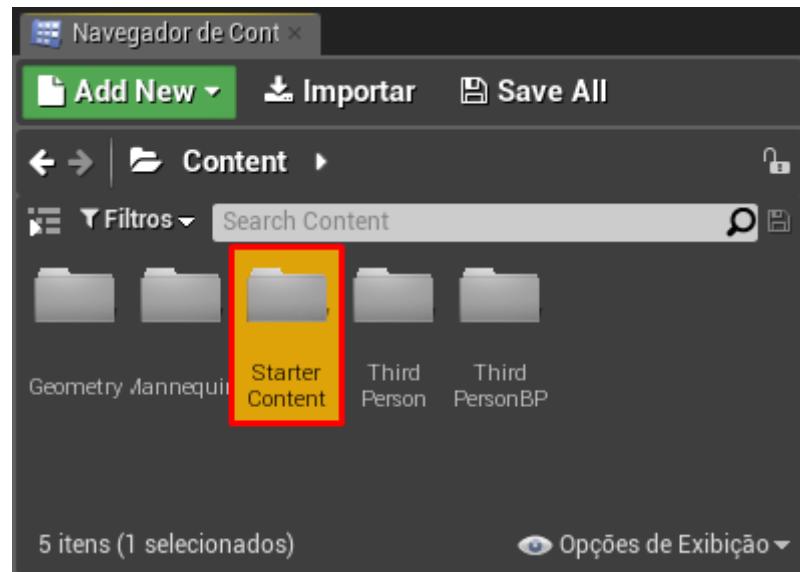
56- Clique em Compile e depois salve:



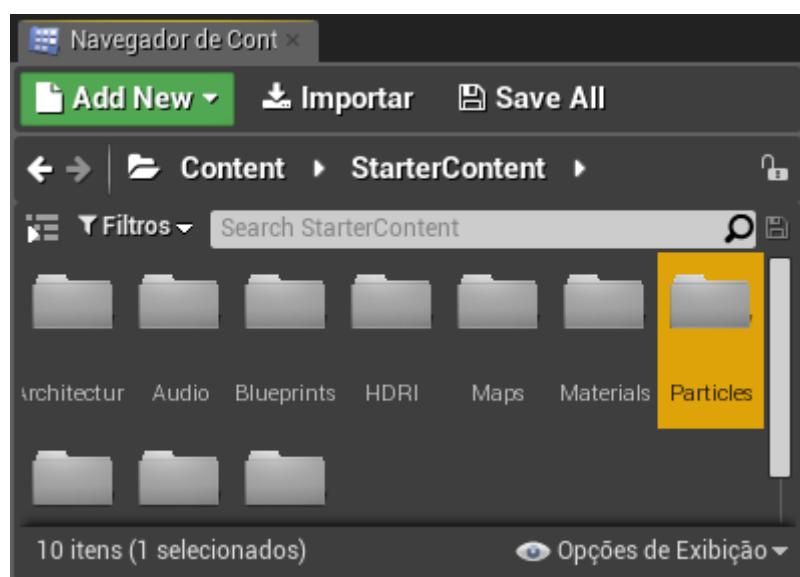
57- Voltando para o game, precisamos adicionar alguma coisa a esse volume, pois em jogo ele é invisível. Para isso, vamos usar o navegador de conteúdo. Clique em Content:



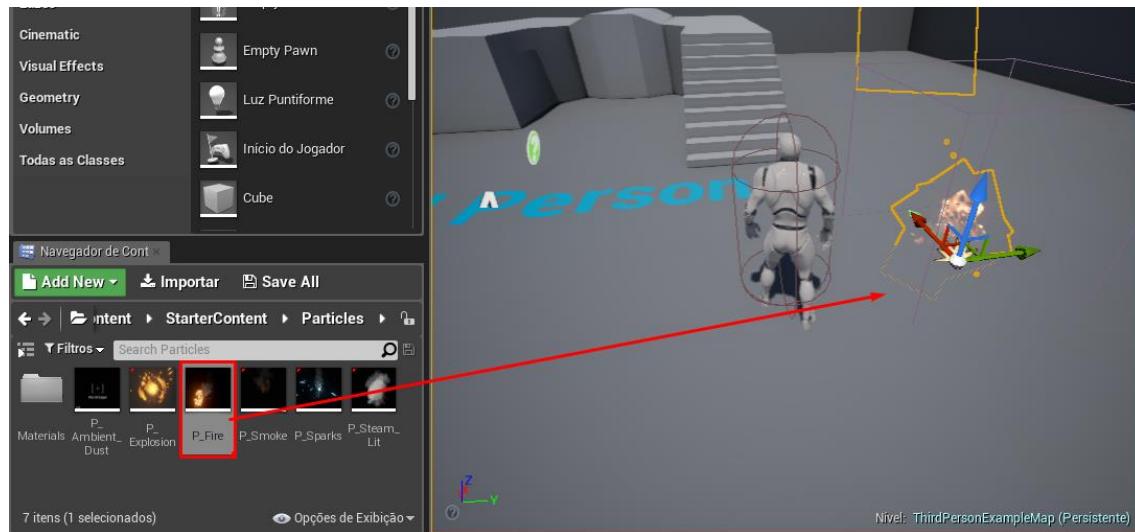
58- Dê um duplo clique na pasta StartContent:



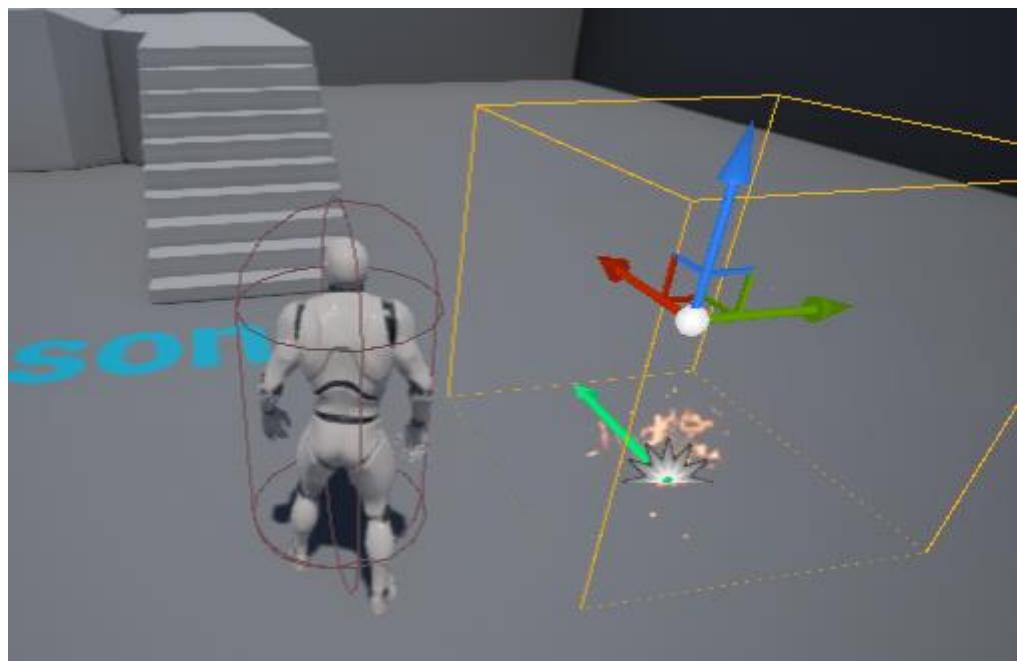
59- Dê um duplo clique na pasta Particles:



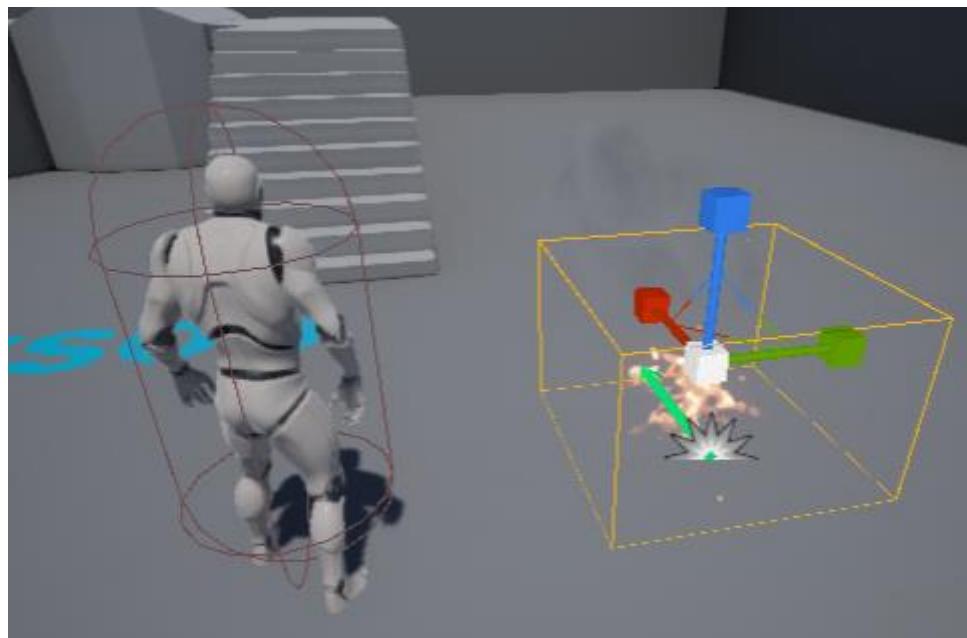
60- Selecione a partícula de fogo, P_Fire e arraste para dentro do volume:



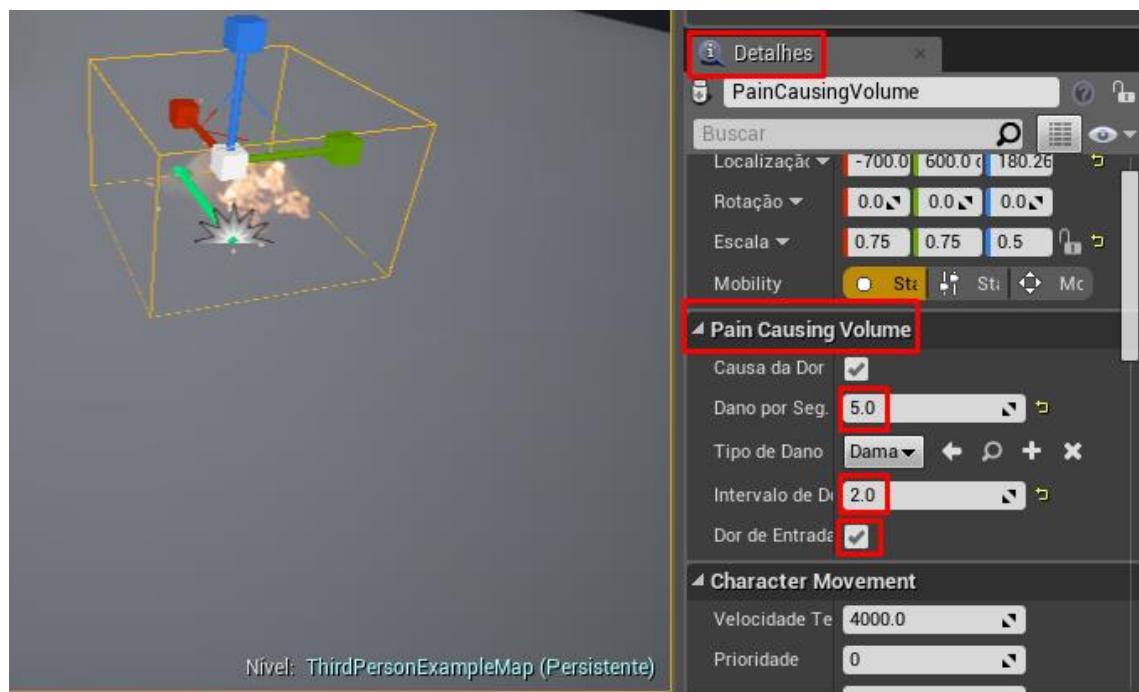
61- Agora, vamos ajustar o tamanho do volume e o dano causado por ele. Selecione o volume:



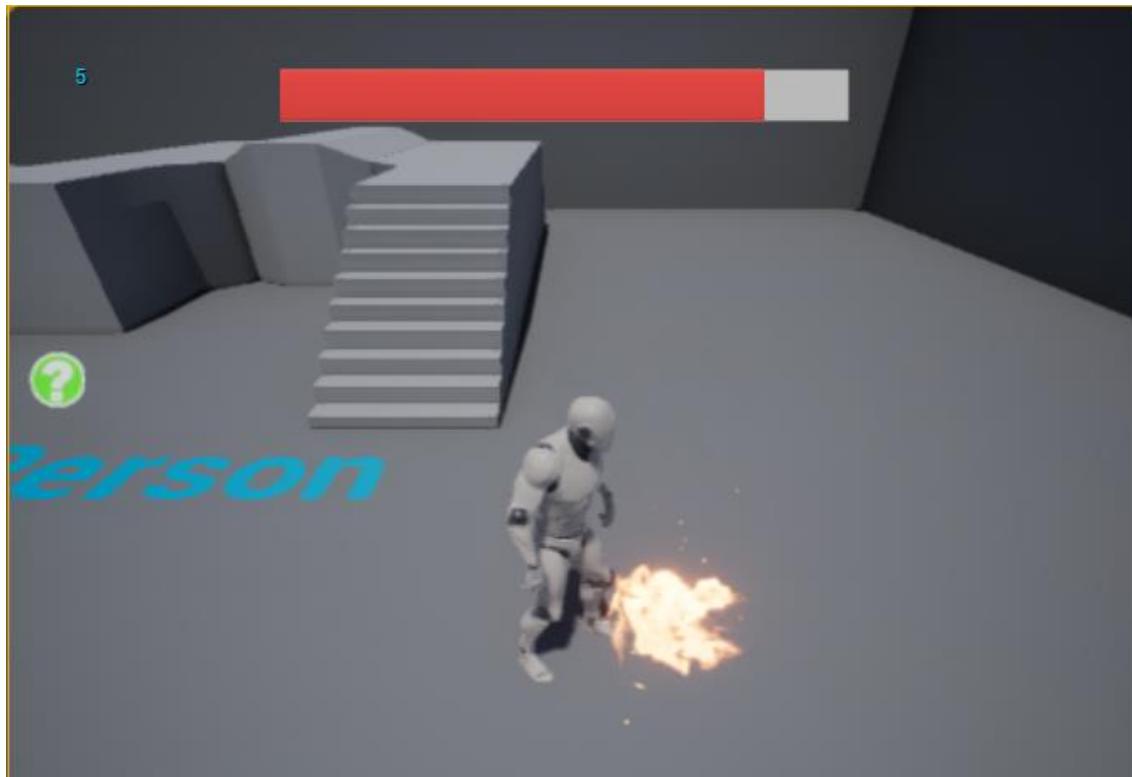
62- Altere as setas de movimentação para as setas de modificação de tamanho e ajuste o volume o mais próximo possível da imagem abaixo:



63- Agora, no menu detalhes, visualize a aba Pain Causing Volume e altere os seguintes valores:



64- Dê play e entre na área do fogo:



65- Observe a barra de vida caindo de acordo como você toma dano do fogo. Na lateral superior esquerda, é informado a você o valor. Esse exercício chegou ao fim. Mostre ao seu instrutor. Não é necessário salvar.