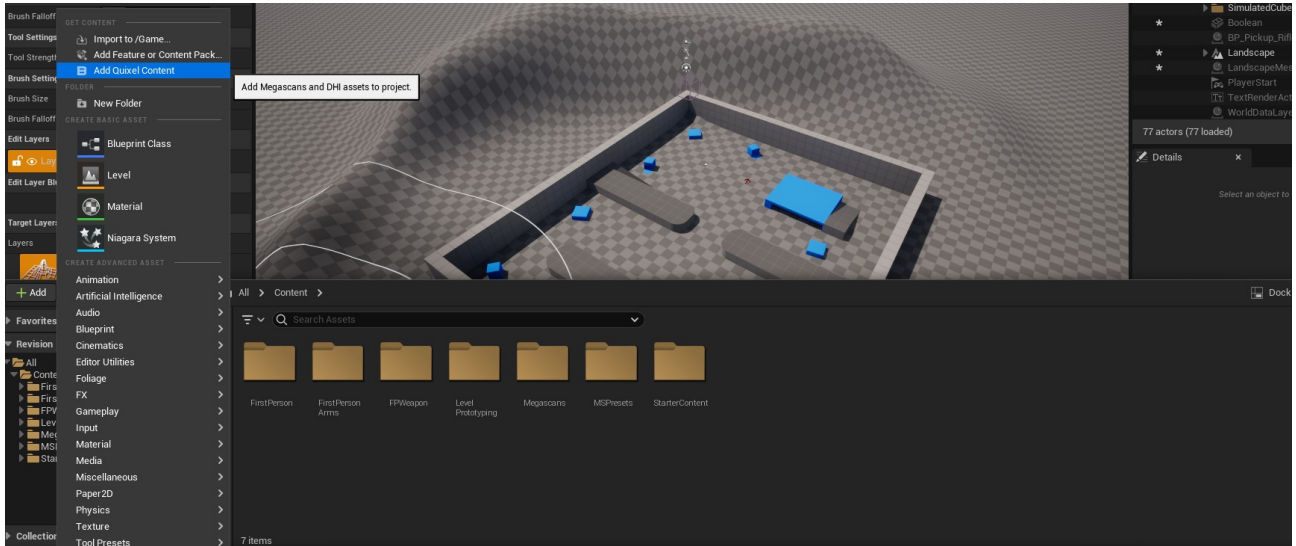


# Révision UNREAL

Ajouter des assets : ctrl+espace → add → Add Quixel

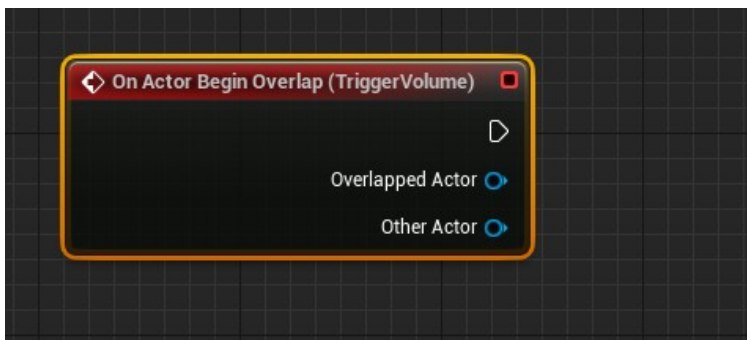
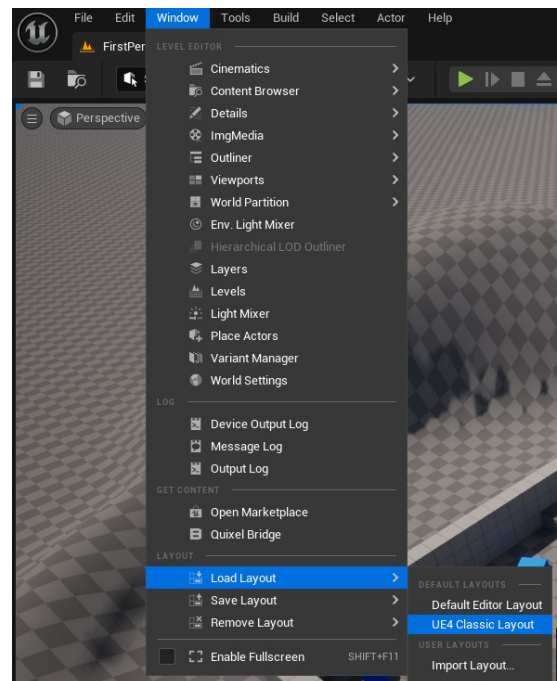


Pour utiliser l'interface de UE 4 :

Pour détecter l'entrée ou la sortie du personnage, utiliser un *TriggerVolume*

Pour gérer l'évènement, sélectionner le trigger volume, aller dans le level blueprint et chercher l'évènement « onActorBeginOverlap »

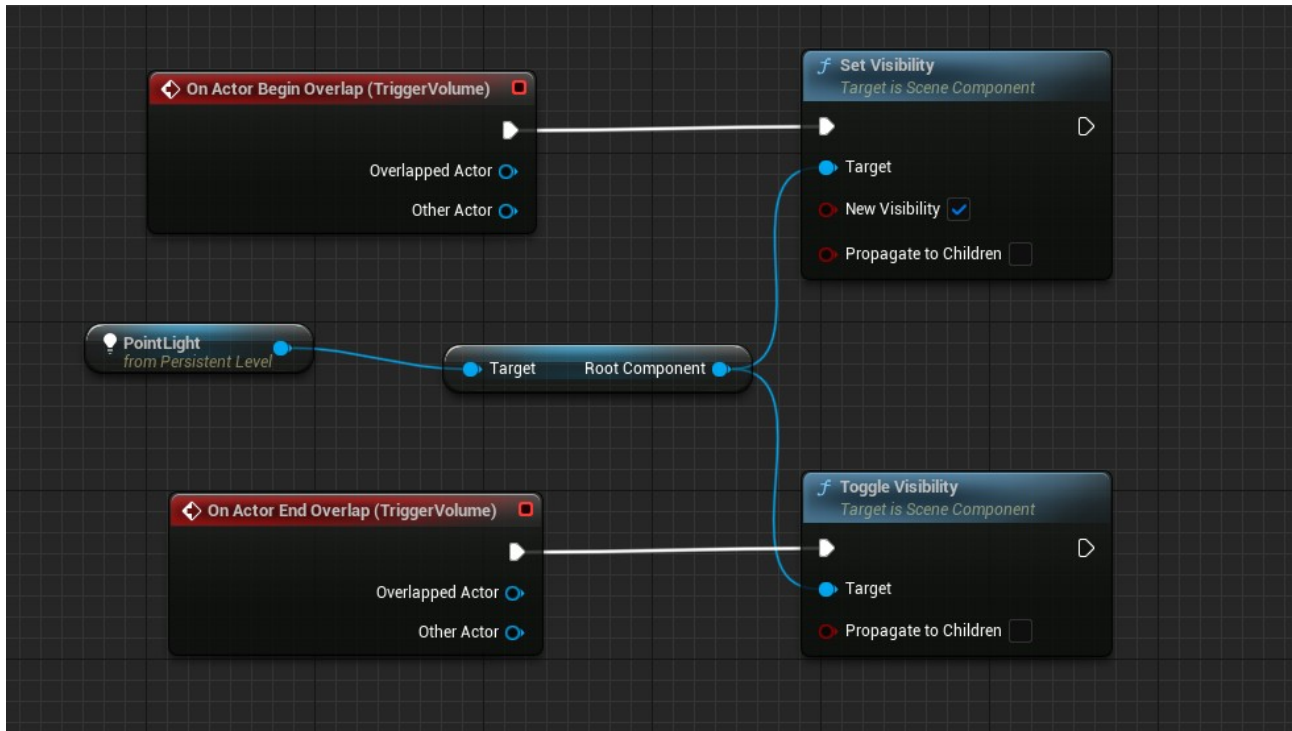
/!\ bien vérifier le « triggervolume »



Allumer ou éteindre une lampe :

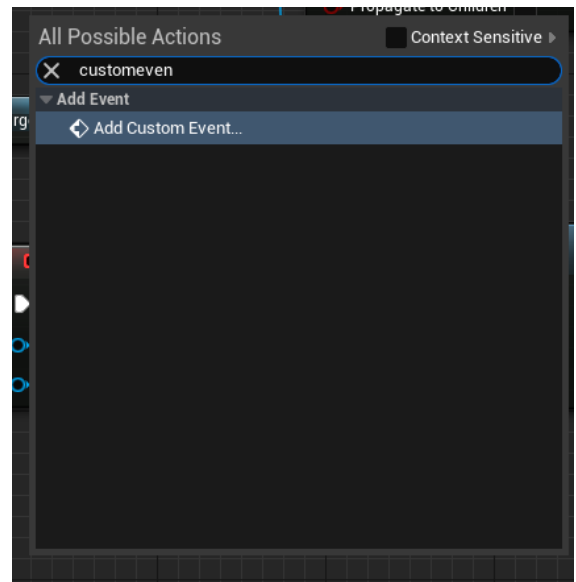
SetVisibility = on peut dire true ou false

ToogleVisibility = si la visibilité est à true, ça passe à false lors de l'appel, et inversement

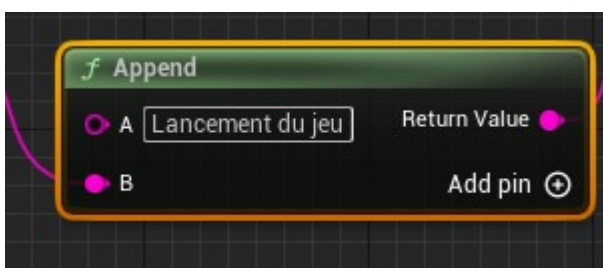


Il est possible de créer ses propres évènements avec  
« custom event »

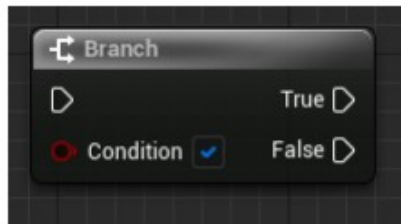
Pour appeler ces évènements en console (pendant une  
run) faire :  
ce <nom de l'évènement>



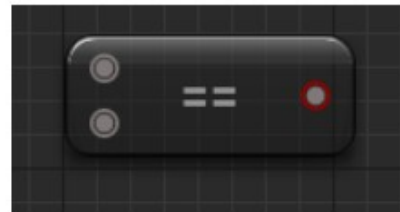
Pour concatener des chaines de caracteres, utiliser Append



if et ==



bloc de test conditionnel



opérateur ==

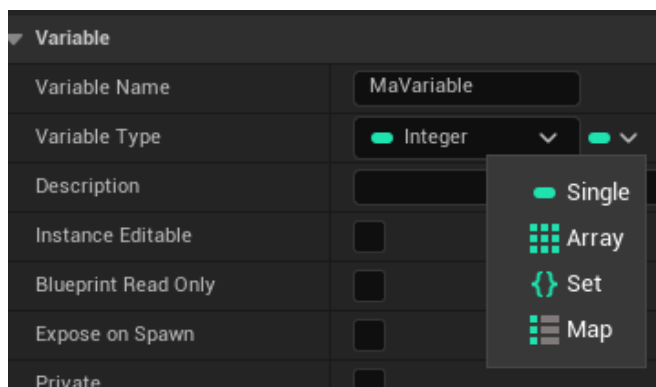
Pour faire un commentaire : Sélectionner les nodes voulu puis cliquer sur « c »

Pour faire une fonction : Sélectionner les nodes voulues → clique droit → *Collapse to function*

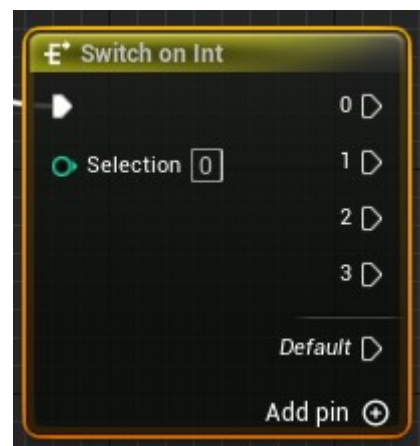
Random



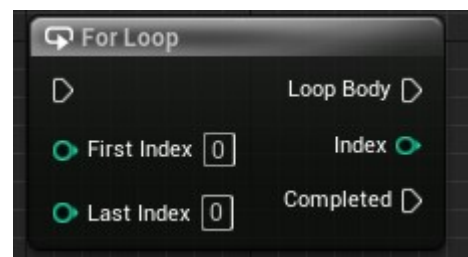
Une liste d'entier



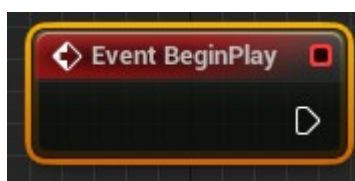
Switch



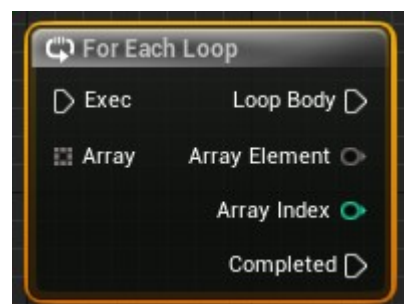
For



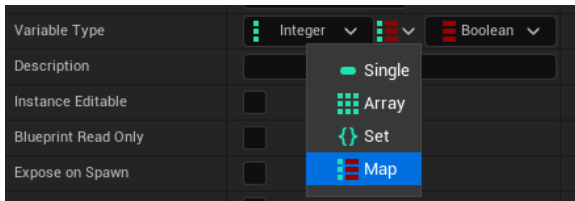
OnBeginPlay = équivaut au Start dans Unity



Foreach



## Map

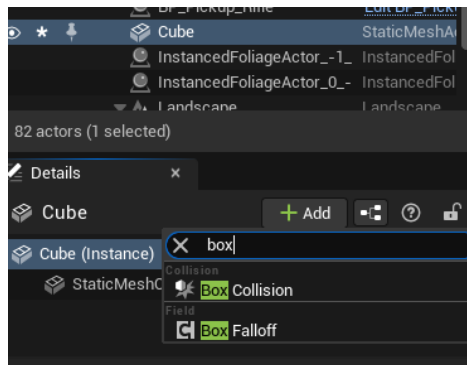


## Rechercher dans une map, utiliser *find*



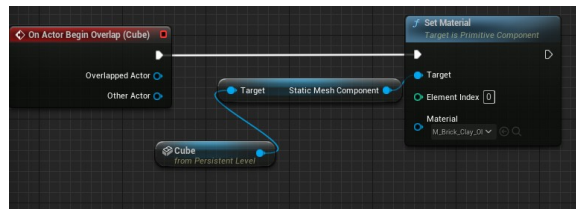
## Box Collision : sélectionner l'objet → Details → Add → Box Collision

!/\ ne pas hésiter à l'agrandir elle est peut être cachée dans l'objet

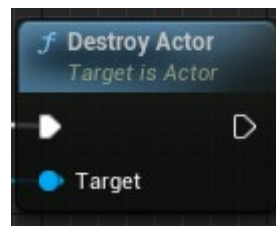


## Changer le material d'un objet : drag and drop l'objet depuis l'éditeur

!/\ il faut peut être enlever le sensitive contexte pour trouver set material



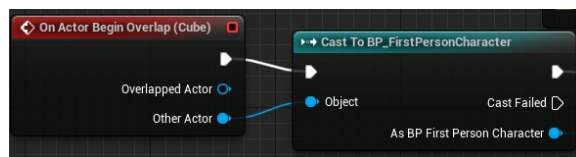
## Destroy un objet, détruire



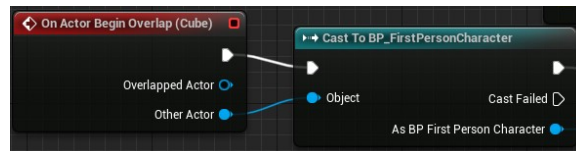
## Téléportation :



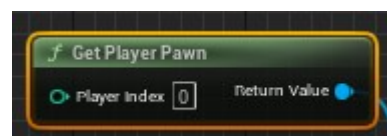
## Pour récupérer le personnage :



## Pour récupérer le personnage :



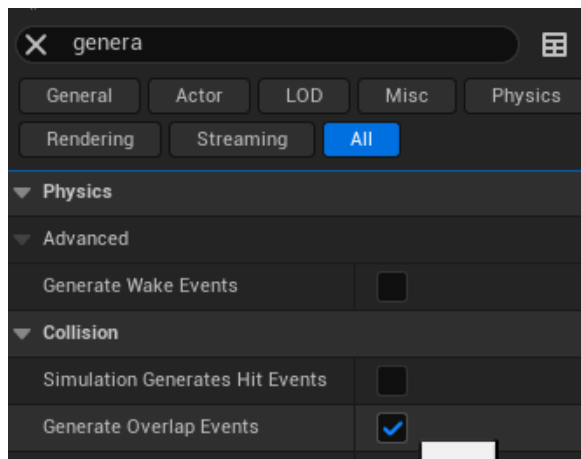
ou bien



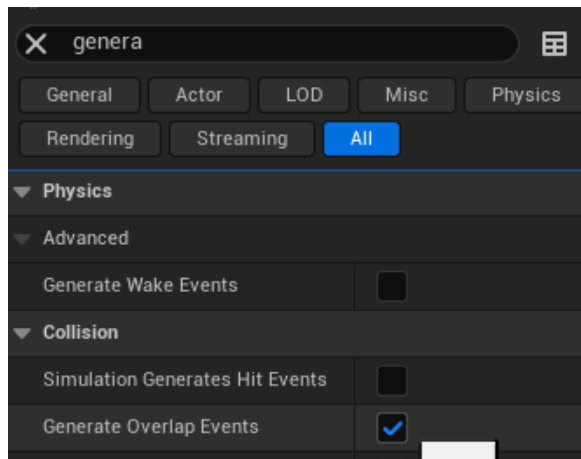
## Pour récupérer la position d'un objet :



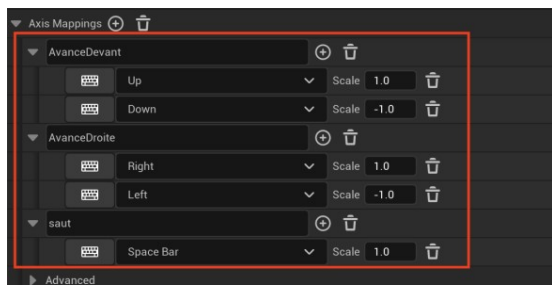
Pour que les objets subissent aussi les évènements faire :



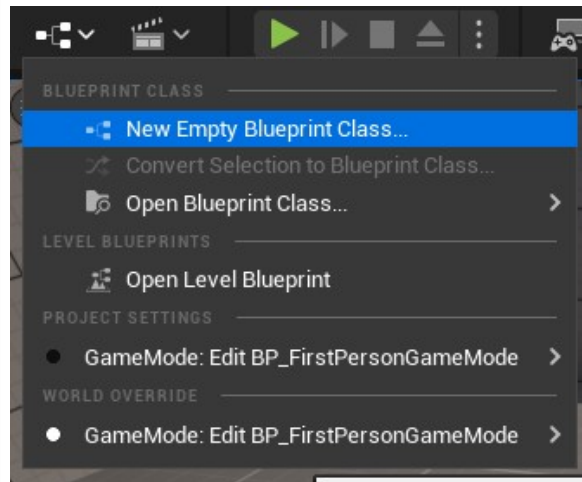
Pour que les objets subissent aussi les évènements faire :



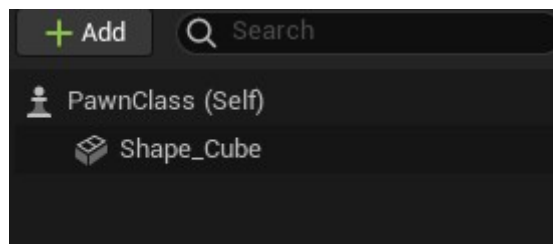
Définir les touches : Edit → project settings  
→ Engine → Input → Axis Mapping



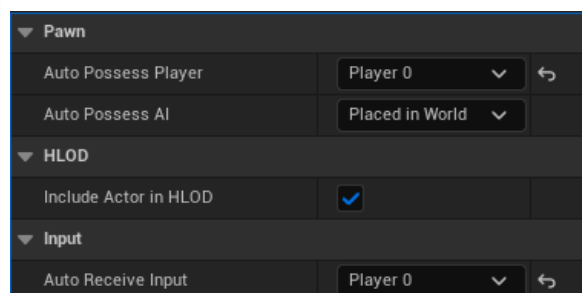
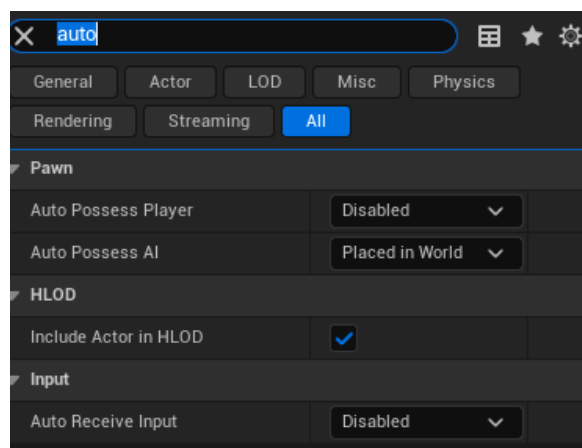
Création d'une classe : new blueprint class  
→ pawn



Attention à la hiérarchie est importante car Unreal Engine prend en compte uniquement l'élément racine de l'objet pour la gestion des collisions.

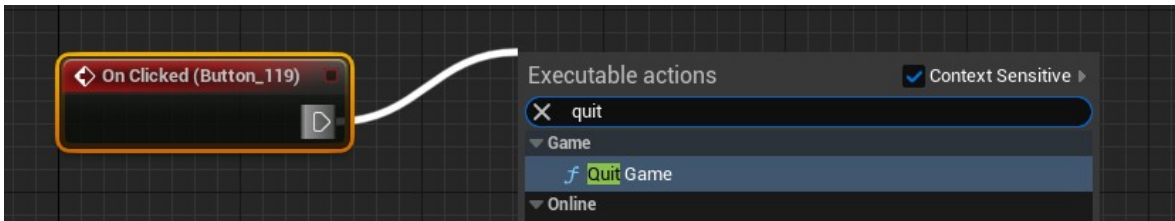
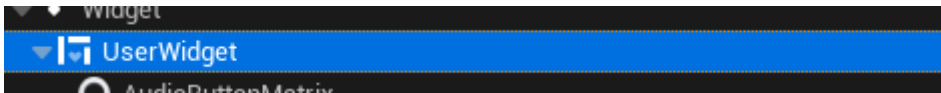
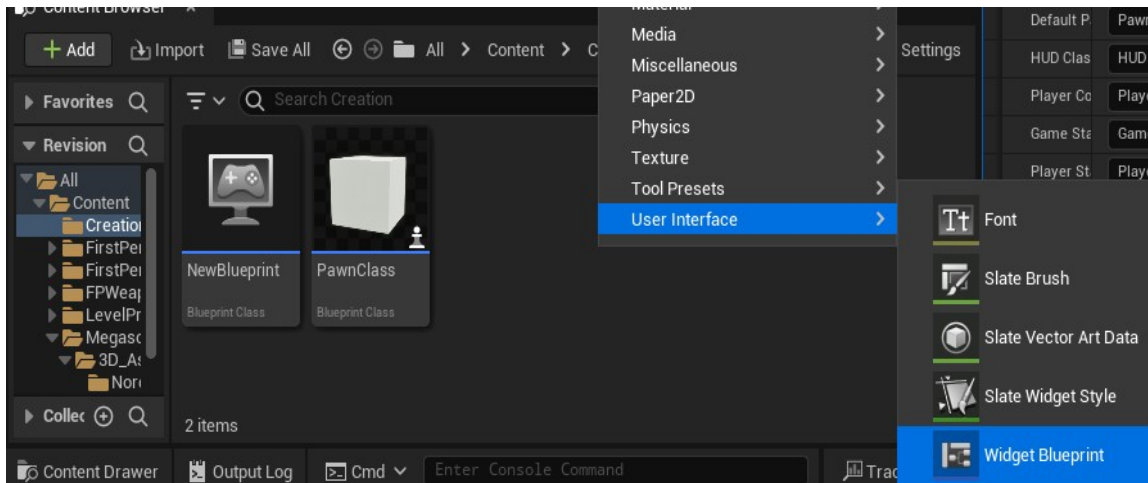


Quand on ajoute un nouvel objet qui représente le joueur il faut changer les deux auto :

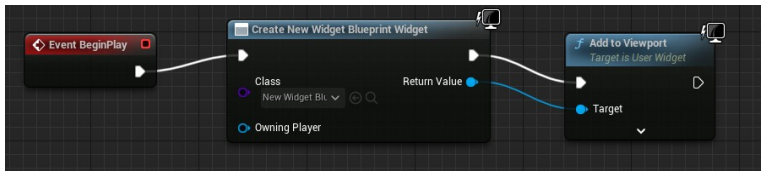




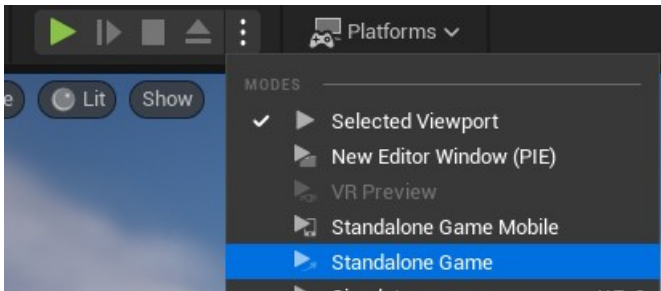
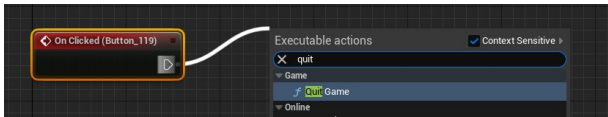
Créer une interface :



Dans le level blueprint faire : (avec dans « classe » le nom du widget)



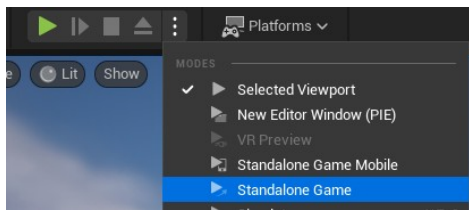
Quitter le jeu :



Changer la résolution de l'écran : vous utiliserez le bp *ExecuteConsoleCommand* avec comme paramètre par exemple la commande : *setres 640x480*. A noter : pour que le changement de taille d'écran soit visible, il est nécessaire que lancer le jeu en mode standalone game

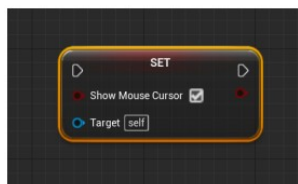
Changer la résolution de l'écran : vous utiliserez le bp *ExecuteConsoleCommand* avec comme paramètre par exemple la commande : *setres 640x480*. A noter : pour que le changement de taille d'écran soit visible, il est nécessaire que lancer le jeu en mode standalone game

Standalone permet de lancer le jeu en changeant les tailles d'écran :

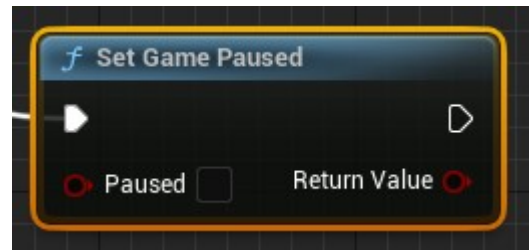


Bien cocher ça dans ce type d'exécution

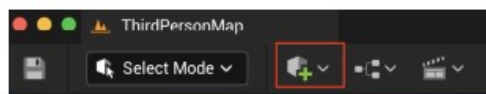
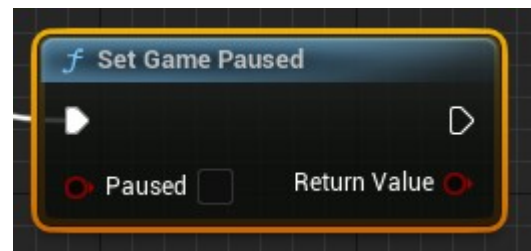
› **Show Mouse Cursor** du player cc



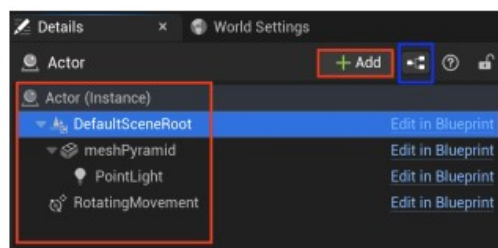
Pause a game :



Pause a game :



1.3 Une fois l'objet placé dans la scène, ajoutez les éléments suivants (bouton vert) : **StaticMesh**, **PointLight** et **RotatingMovement**. Vérifiez et modifiez éventuellement l'organisation de votre objet de façon à avoir la hiérarchie de l'image ci-dessous (cadres rouges).

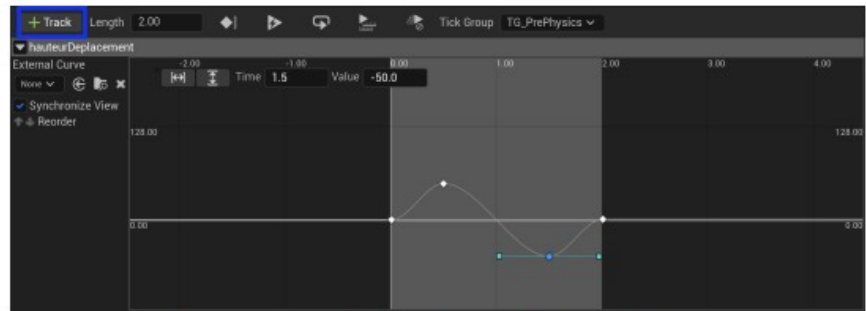


1.4 Convertissez l'objet en une classe **Blueprint** réutilisable (voir cadre bleu image ci-dessus) que vous placerez dans le répertoire **CreationsTP5**. Ouvrez l'objet et spécifiez le type de maillage de l'objet **StaticMeshComponent** comme étant **Shape\_QuadPyramid** et précisez aussi le matériel **M\_Metal\_Gold**.

1.6 Le rajout d'un objet *timeline* peut se faire dans l'onglet *Event Graph* de votre objet à animer. Rajoutez ce nouvel élément comme illustré sur l'image (a).



(a)



(b) ajout de point sur la courbe : clic droit

1.7 Double cliquez sur l'élément précédent de façon à faire apparaître l'éditeur d'animations. Rajoutez une nouvelle courbe (bouton *Track* encadré en bleu sur l'image ci-dessus) de types *Float*

UV : Modeling → UV → project uv

