

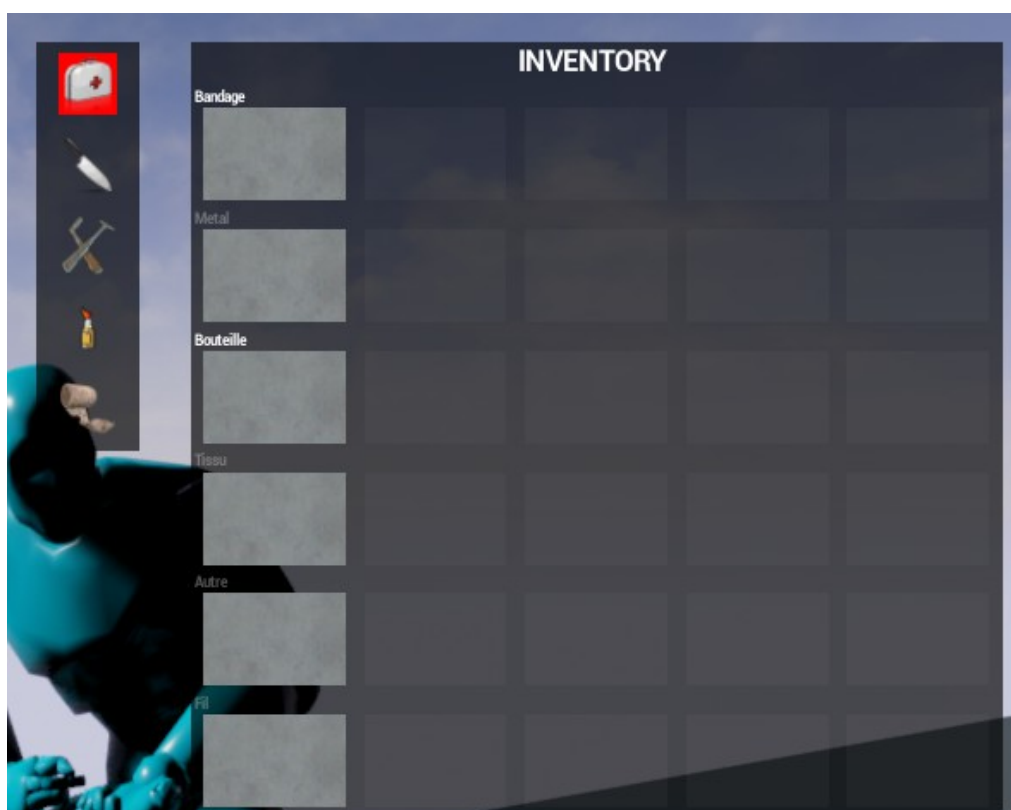
Introduction

(NB : J'espère que ce tuto vous sera utile et qu'il sera compréhensible par tous. Si vous avez des questions n'hésitez pas à m'envoyer un message sur Facebook « Antoine Gargasson » ou par mail antoine.gargasson@gmail.com)

(NB2 : Ce tuto est réalisé sur la version 4.8.3 de UE4)

Dans le premier tuto nous avons mis en place un système d'inventaire et une manière de le visualiser. Dans le tuto d'aujourd'hui nous allons continuer sur ce projet et rajouter le ramassage des objets ainsi qu'un déplacement basique.

Je rappelle l'objectif final du tuto : un système inventaire/craft à la The last of us.



La dernière fois nous avons mis en place le carré de droite avec les items et leur nombre.
Aujourd'hui nous allons pouvoir remplir l'inventaire et se déplacer dans le monde pour aller chercher du loot.

Ramassage des objets dans le monde

Nous allons donc commencer par rendre un objet récupérable dans le monde et qu'il s'ajoute à l'inventaire.

Rappelez vous que vous avez créé des fonctions qui seront fort utiles dans le tuto d'aujourd'hui.

Avant de créer les objets nous allons rajouter un peu de **mouvement** dans notre scène.

Pour cela il faut créer **4 axes de mapping**. Pour rappel vous trouverez l'éditeur d'Input dans **Edit, Project Settings** et **Input**. Le premier va s'appeler Avant, il va être bindé sur Z avec 1.0 en puissance (scale).

Je vais résumer ce bind comme suit : **Axe (Avant, Z, 1.0)** , c'est comme ça que je vous donnerai les informations sur un bind dorénavant : Type(Nom, Touche, (optionnel)Valeur).

Vous allez donc devoir créer 4 axes comme suit :

Axe (Avant, Z, 1.0)

Axe (Avant, S, -1.0)

Axe (Droite, D, 1.0)

Axe (Droite, Q, -1.0)

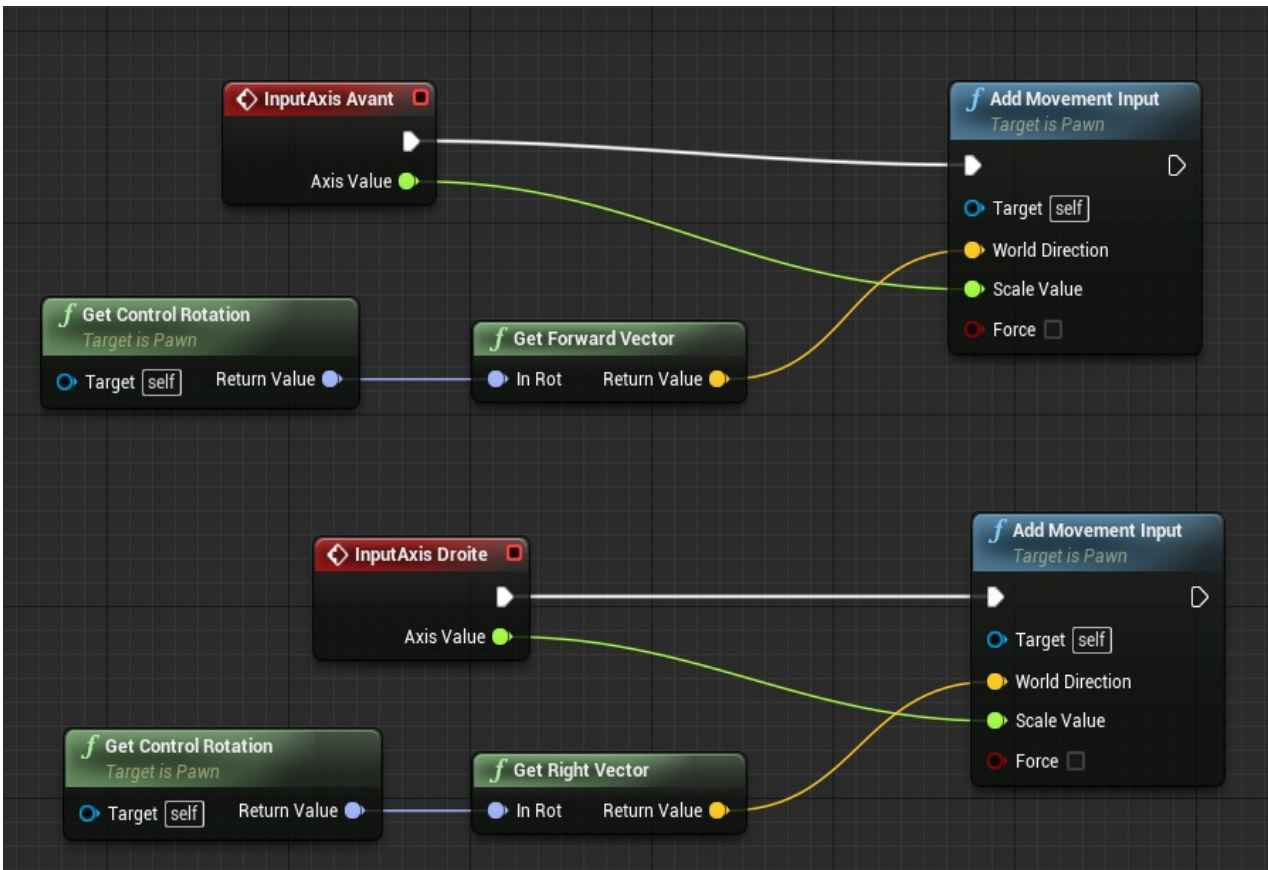
Axe (Tourner, MouseX, 1.0)

Axe (Lever, MouseY, -1.0)

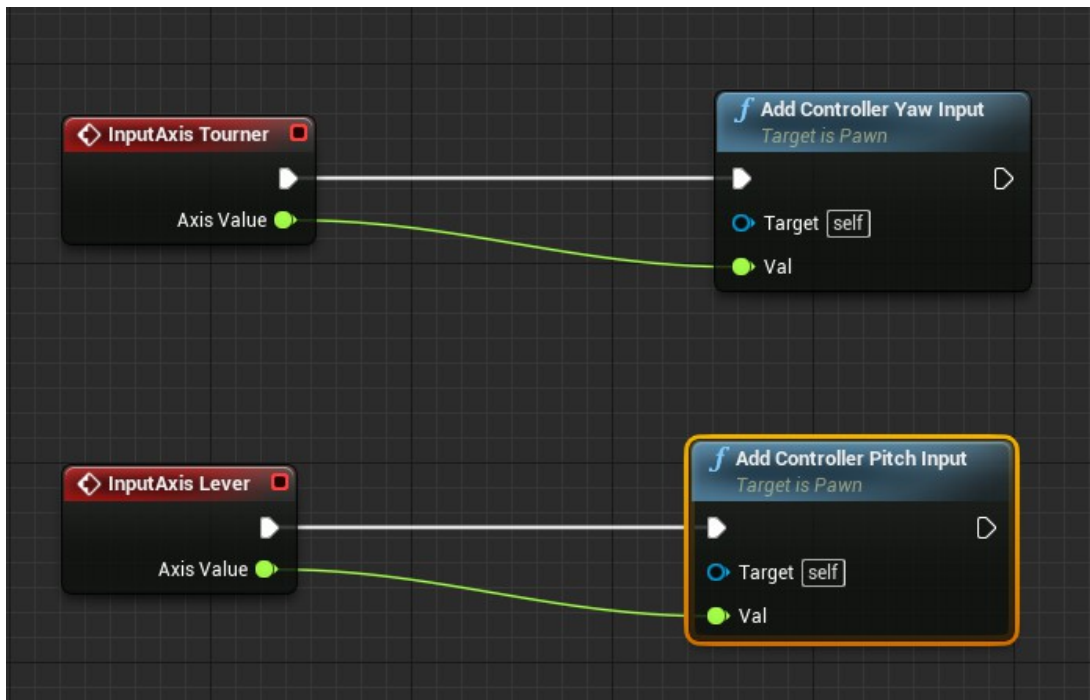
une fois ces axes créés, **rendez vous** dans le blueprint **BP_Player**. Ici nous allons rajouter du mouvement. Ce sont des déplacements basiques qui pourront vous servir dans vos projets futurs.

On va utiliser le **character movement** déjà présent pour gérer les déplacements.

Pour commencer avec les déplacements, procédez comme suit :



Si vous testez votre projet maintenant, vous pouvez voir que vous pouvez vous **déplacer** mais pas bouger la souris. Rajoutons un peu de mouvement à la **caméra** pour avoir un déplacement basique et pouvoir ramasser des objets plus facilement :



Vous avez maintenant un **déplacement** de personnage basique et classique. (C'est le déplacement utilisé dans les projets de base 3^e et 1^e personne)

Il est temps de passer aux choses sérieuses et de revenir sur nos objets **ramassables** (loot).

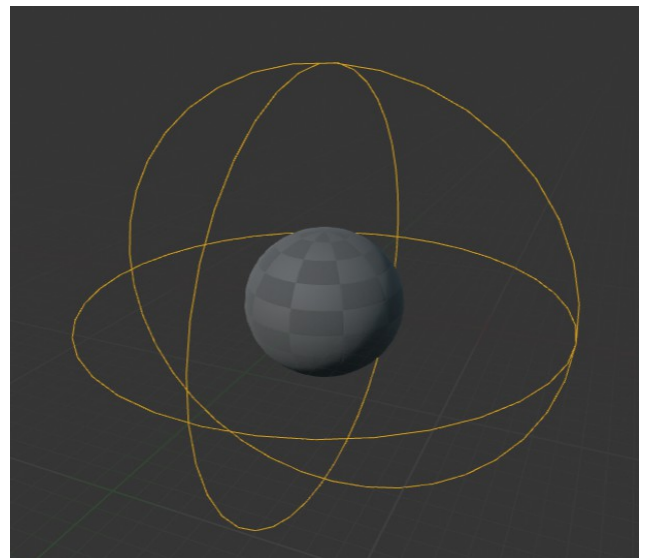
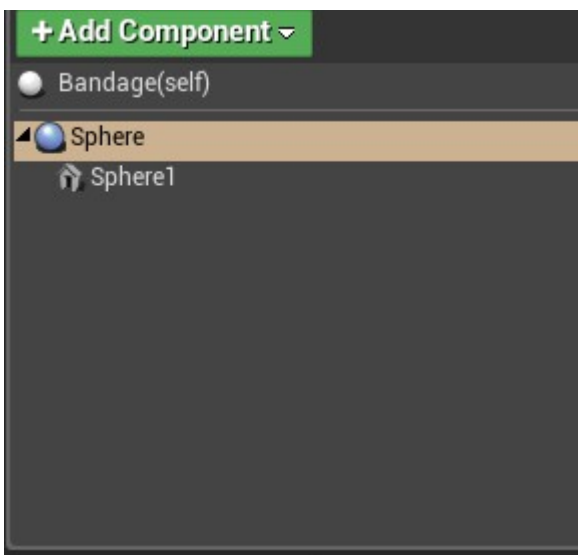
Un **loot** va être constitué d'un **static mesh** qui sera la forme que verra le joueur dans le jeu, d'une **sphère de collision** afin de détecter que le joueur est près de l'objet et enfin d'un **ID** qui sera utilisé pour ajouter le bon loot à l'inventaire.

On va commencer par créer l'Item. C'est une classe parent. Nous allons faire de l'héritage par la suite pour les autres loot. Je vous explique toutes les étapes pour créer un loot et je vous laisserai le soin de créer les autres objets par vous même ou de simplement créer des classes filles ;)

Pour **l'Item**, on va utiliser une représentation de **sphère** et rajouter une **sphère de collision** autour pour détecter le joueur. Sans oublier de mettre un **ID** pour ajouter le loot au moment du ramassage.

On commence par créer un nouveau **blueprint Actor** et on le nomme **Item**. Par défaut on a déjà une sphère. **Ajouter** une **sphère collision**. Vous remarquerez qu'elle s'est mis en enfant de la sphère déjà présente. Faites un **drag&drop** de votre **sphère collision** sur son parent afin que la sphère collision devienne le **composant racine**. Rajouter une sphère classique en enfant afin d'avoir un objet visible pour le joueur.

Oups la sphère est trop grande pour la collision ? Pas de soucis ! **Sélectionnez** la **sphère collision** et passer le **sphère radius** à **150** .

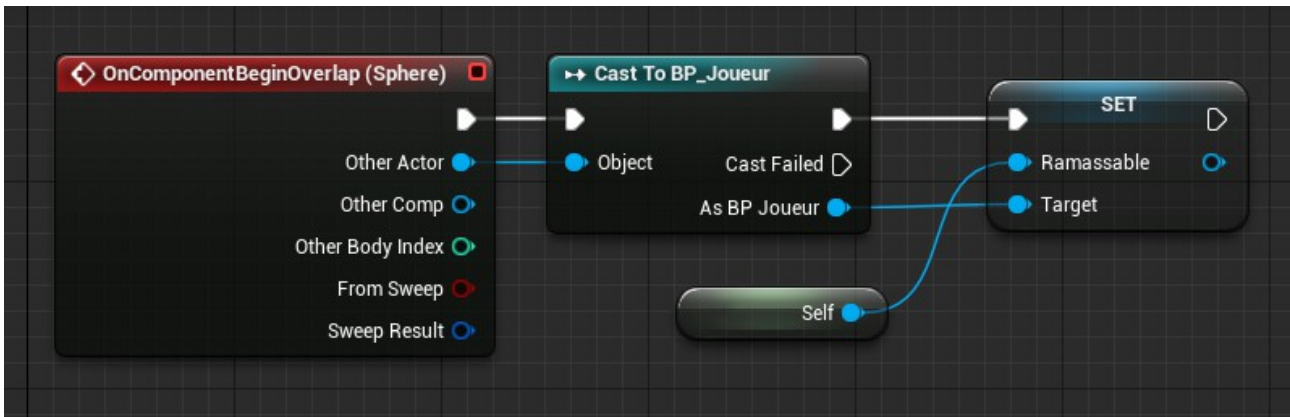


Nous sommes presque prêt à utiliser ce loot. N'oubliez pas de **rajouter** une variable de type **Integer**, nommé **ID**.

On compile et on sauvegarde !

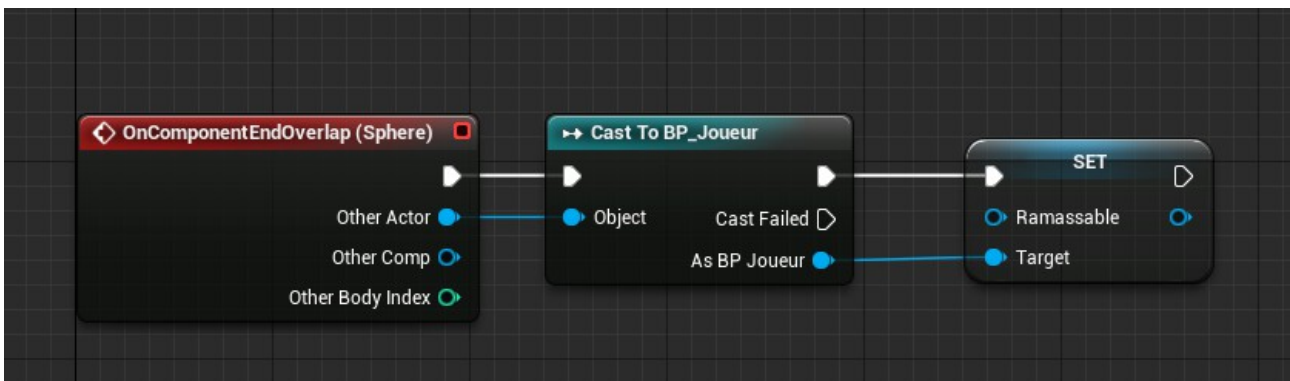
Aller dans BP_Player. Nous allons **rajouter** une variable de type **Actor** qui va être le loot dans lequel on est rentré. Nommez là **Ramassable**. Cliquez sur le petit œil pour la rendre publique.

De retour dans le blueprint du bandage sélectionnez votre sphère et regardez tout en **bas** dans les **details**. Il y a plein de **croix** ! Ce sont des fonctions que vous pouvez utiliser dans votre **event graph**. Trouver **On Component Begin Overlaps** et cliquez dessus. La fonction va s'ajouter dans le graph et nous allons pouvoir nous en servir.



On vérifie que le joueur entre dans la sphère et on attribut l'objet ramassable au bandage. (**self** est une référence à l'objet lui-même).

De même pour **On Component End Overlap** :



Ici en attribuant rien à l'objet ramassable, on efface l'ancien objet qui était contenu dans la variable.

Ces deux actions seront effectuées par tous les loots, on peut donc se permettre de faire une **classe mère** pour tous les loot ce qui nous évitera de tout le temps réécrire le code.

Il est l'heure de mettre à profit votre mémoire !

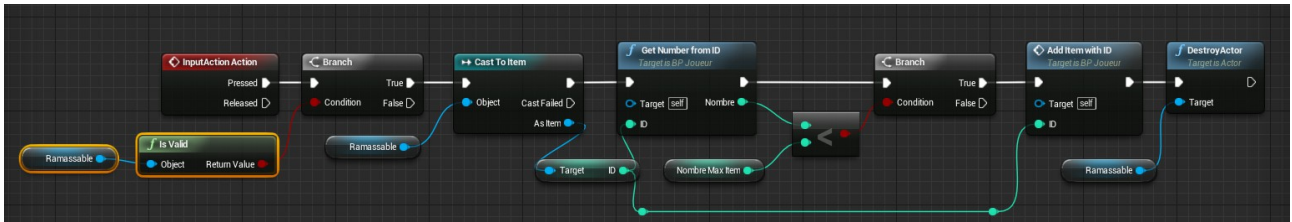
Il faut maintenant ajouter un mapping d'action pour ramasser l'objet. Rajoutez ce qui suit :

Action (Action, E)

Il fallait en effet rajouter un bind d'action avec la touche E, bravo à ceux qui l'avait déjà fait ! Ça commence à rentrer ;))

De **retour** dans le blueprint **BP_Player**, on va faire en sorte de **ramasser l'objet** et de **l'ajouter** dans l'inventaire **si c'est possible**.

On va pouvoir utiliser le bind d'action comme suit :



On vérifie si l'objet ramassable **existe**, on le **cast** pour pouvoir l'utiliser, on vérifie **combien** on en a et si on en a pas trop, on **l'ajoute** à l'inventaire et on **détruit** l'objet dans le monde.

Vous pouvez à présent faire un **clique droit** sur le blueprint **BP_Item** et faire un **Create Child Blueprint Class** . Une fois nommé **Bandage**, vous pouvez faire une **drag&drop** du **bandage** dans le monde et jouer. Approchez vous, faite **E** et magie ! L'objet **disparaît** et si vous ouvrez votre **inventaire** vous avez un bandage en plus ! :)

Essayez en ajoutant les autres loot et en lançant le jeu. N'oubliez pas de **changer** l'ID du nouveau loot.

Vous pouvez maintenant mettre des loot dans le monde et remplir votre inventaire. :)

Dans le prochain tuto vous allez réutiliser tout ce qu'on a fait jusqu'à présent et faire du craft !

Pour toute éventuelles questions, remarques, détails ou demandes, n'hésitez pas à prendre contact avec moi ! (Cf : début du tuto)