

Analyse théorique:

1. De quel type de jeu s'agit-il ?

Layers of Fear est un jeu vidéo d'horreur psychologique, racontant l'histoire d'un peintre troublé souffrant de troubles psychiatriques, errant dans son manoir afin d'y compléter son chef-d'œuvre.

2. Après une heure de jeu, nommez les différentes mécaniques de gameplay rencontrées.

(Pour des raisons de simplicité, nous considérerons que le jeu se joue sur ordinateur avec une souris.)

Le jeu permet au joueur de se déplacer dans un univers en 3 dimensions, de bouger la caméra selon les mouvements de souris, d'effectuer un zoom dans le champ de vision en appuyant sur une touche, allumer ou éteindre une lampe de poche sur une autre touche, se déplacer plus rapidement, considéré comme courir en maintenant une touche différente.

Le joueur peut également interagir avec des objets. Certains nécessitent simplement d'appuyer sur la touche correspondante lorsque l'icône d'interaction s'affiche, d'autres nécessitent également un mouvement de souris, c'est le cas des portes ou des tiroirs.

Enfin, d'autres objets peuvent être inspectés, ces derniers se divisent en deux catégories, les objets avec du texte qui peuvent être lus et des objets qui peuvent être manipulés et observés sous divers angles en manipulant la souris.

3. Ce jeu mise son attrait sur l'immersion du joueur. Expliquez en détail comment cela est mis en place (à l'aide de quelles mécaniques, feedbacks, ambiance,...).

L'immersion du joueur dans la peau du personnage se fait de différentes manières.

Tout d'abord, nous pouvons observer que la vision du joueur, la caméra, n'est pas complètement stable, et ce, même lorsque nous n'influons pas sur elle, celle-ci effectue quelques mouvements irréguliers verticaux et horizontaux, reflétant des mouvements de tête involontaires effectués naturellement, mais particulièrement, car le personnage a un défaut de posture, ayant du mal à se tenir droit.

Effectivement, nous remarquons également que le mouvement du personnage n'est pas complètement fluide, nous avons l'impression qu'il boîtie. Ceci est corroboré par les indices que nous trouvons au sein de la résidence du protagoniste, et ce, assez rapidement. En effet, nous

trouvons des notes sur lesquelles il est écrit que quelqu'un (le personnage lui-même), a du mal à se déplacer, nous trouvons également une canne au pied du lit de la chambre maîtresse.

Par ailleurs, à de nombreuses reprises, des sons divers peuvent être perçus, s'apparentant parfois à des chuchotements et ceux-ci se déplacent spatialement, c'est particulièrement perceptible lors du port d'un casque audio où les sons peuvent se déplacer d'une oreille à l'autre. Le fait d'entendre ces sons dérangeants peut nous mettre mal à l'aise, mais ceux-ci servent aussi à mieux ressentir le tourment psychologique du personnage incarné, autant victime que nous de ces hallucinations auditives.

Ensuite, les sons environnants permettent une fois de plus de bien se fondre dans l'univers du jeu, que ce soit le plancher qui grince lorsque nous marchons dessus, de même pour les morceaux de verre parfois éparpillés sur le sol, les bruits de portes qui s'ouvrent, le jeu n'a pas de musique ambiante en jeu qui serait extra-diégétique, ce qui paraît plus réel et immersif.

Et puis, l'ambiance générale du jeu complète les effets sonores, le manoir délabré et inhabité donne une sensation de malaise, sa grande superficie contribue à paraître faible et insignifiant dans ce dédale. La pluie et l'orage qui s'abattent dehors, perceptibles au travers des fenêtres ne sont pas non plus rassurants, mais inquiétants, limitant notre vision de l'extérieur, nous enfermant dans les confins du manoir où nous sommes contraints d'évoluer.

En outre, du fait que les tiroirs ou les portes s'ouvrent en fonction des mouvements de souris, cela donne l'impression au joueur qu'il les ouvre ou les ferme vraiment par un mouvement imitant ce que nous ferions hors du jeu pour interagir avec ces objets.

4. Le gamefeel du jeu passe par énormément de petits détails. Qu'est-ce que le gamefeel ? Donnez une liste des détails que vous avez perçus durant votre partie.

Le gamefeel correspond à une sensation intangible difficile à définir lorsque nous interagissons dans un jeu vidéo. Le gamefeel se manifeste de différentes manières et cherche à donner une sensation plus agréable, une expérience de jeu satisfaisante, cohérente et proportionnée aux actions effectuées.

Le gamefeel ne cherche pas toujours à être réaliste, mais à ce que l'expérience de jeu soit immersive, et ce, principalement par les actions du joueur, ses « inputs », leur réponse, le contexte et les règles établies, entre autres.

Par exemple, selon les caractéristiques du personnage incarné, si ce dernier est censé être rapide et agile, il n'a pas besoin de courir à une vitesse réaliste ou sauter une hauteur qui suivrait les lois de la physique, sauf s'il s'agit du but recherché, mais plutôt de se comporter de la manière que nous l'imaginons, donc dans ce cas, à une vitesse élevée et une hauteur de saut importante. Les attributs, capacités et effets relatifs aux personnages et leurs environnements sont parfois exagérés pour accentuer un effet désiré.

De plus, si le joueur appuie de manière plus importante ou plus longtemps sur une touche, il est attendu que la réponse soit proportionnée, et donc par exemple, asséner un coup plus puissant.

Le gamefeel est alors souvent attribué à des mouvements et à la manière dont les éléments interagissent entre eux.

Dans le cas de Layers of Fear, là où généralement, nous apprécions avoir des mouvements fluides, le mouvement titubant et incertain du protagoniste corrèle avec l'intention narrative d'avoir un personnage ayant du mal à marcher. Cela permet aussi d'avoir des mouvements plus lents et d'accentuer la panique du joueur s'il devait être exposé à un élément dangereux auquel il ne pourrait pas s'enfuir en raison de sa vitesse de déplacement.

Par ailleurs, le gamefeel de Layers of Fear passe aussi par le biais d'ouvrir les portes et les tiroirs par un mouvement de souris déjà évoqué, où en tant que joueur nous avons la sensation de réellement interagir avec ces objets dans une certaine mesure, ainsi que la possibilité d'inspecter et

d'examiner des éléments en les manipulant encore une fois avec la souris, donnant l'impression de palper et de déplacer ces objets.

De surcroît, le manoir semble labyrinthique, les salles semblent fondées sur un système de géométrie non-euclidienne ou alors se déplacer d'elles-mêmes. En effet, une porte menant à une pièce, en présentera parfois une autre lorsque le joueur essaiera d'emprunter le même passage, pouvant donner l'impression d'être perdu·e dans un dédale.

5. Expliquez comment est mis en place le game over dans ce jeu.

Il n'existe pas réellement de « game over » classique où le joueur a explicitement perdu et est renvoyé à l'écran titre, ou est contraint de refaire un certain passage en boucle jusqu'à ce qu'il réussisse malgré un contre-exemple.

Néanmoins, lorsque le personnage principal tombe d'une forte hauteur ou qu'il est attaqué par le fantôme de sa femme, il semble se réveiller d'un malaise dans une salle près d'où il a perdu connaissance, parfois avant, parfois après.

Dans le cas où une énigme ne serait pas résolue, notamment dans la bibliothèque où il est possible de chuter, le joueur recommencera un peu avant l'énigme, en passant par un passage légèrement différent qu'emprunté la première fois. Cet événement reste très spécifique et l'occasion d'y être confronté est moindre.

Tandis que si le joueur rentre en contact avec sa défunte compagne, il sera « puni » par un « jumpscare », il s'agit d'un événement brusque, généralement associée à une image ou vidéo accompagnée d'effets sonores dans le but d'effrayer brutalement le joueur. Le personnage réapparaîtra visiblement dans une salle suivante et n'aura plus l'opportunité de la croiser au même endroit et d'être victime du même jumpscare. Cet événement est plus commun que celui énoncé plus tôt.

Ainsi, le jeu et l'expérience de chaque joueur seront légèrement altérés selon les actions effectuées, et la possibilité d'avoir une fin alternative sera aussi impactée en fonction du nombre de fois qu'il est confronté au spectre de son épouse.

6. Layers of Fear essaye de faire perdre le joueur dans son labyrinthe dynamique. En effet, les salles "bougent" toutes seules. (un exemple à partir de ce moment de la vidéo : <https://youtu.be/fMgmb-J0Guc?feature=shared&t=1705>)

Expliquez en **détail** et en **termes de programmation** comment cela est mis en place.

Je suppose que la manière dont est structuré cet événement peut-être de deux façons, mais passe dans les deux cas par le biais d'un déclencheur, considéré « trigger » ou « overlap ».

Il est possible qu'au cours du jeu, les deux méthodes soient utilisées.

Premier cas:

Une fois que le joueur a passé la porte et que celle-ci se ferme, un événement va se déclencher, en l'occurrence, ce qui s'y trouve derrière va changer. La salle précédente va être déplacée ou supprimée de la scène de jeu et à la place, une nouvelle salle va être remplacée.

Deuxième cas:

Une fois que le joueur traverse une porte, il marche au travers d'une zone invisible qui va déclencher un événement et le personnage va être téléporté dans un autre endroit de la scène qui contiendra la même pièce qu'il visualisait ou allait traverser, mais celle-ci est reliée à une pièce différente de celle dont il est venu.

7. Personnel : le jeu vous a-t-il plu / déplu ? Pourquoi ?

Je pense pouvoir affirmer que le jeu m'a plu.

J'aime bien le genre de l'horreur et *Layers of Fear* a une approche différente de nombreux de ses contemporains, où l'horreur est souvent très brusque, par le biais de jumpscares assez souvent mal amenés ou par l'angoisse d'être pourchassé·e par une entité malfaisante. *Layers of Fear*, n'est pas exempt de ces deux points mais dans ce cas, je pense que les jumpscares sont moindres et ont un but narratif et punitif, ces derniers bénéficient également d'une tension constante agrémentée par l'ambiance du jeu et ne semblent jamais surexploités. Le fantôme de l'épouse du protagoniste hantera ce dernier tout au long du jeu mais ne le poursuivra jamais bien que nous ayons la possibilité de courir et que les déplacements hasardeux du personnage pourraient rendre ces courses-poursuites hypothétiques plus oppressantes.

La frayeur du jeu repose davantage sur l'anticipation de ce qu'il pourrait arriver, elle ne semble pas superficielle mais davantage psychologique.

Tout au long du jeu, nous apprenons progressivement que le véritable monstre qui tourmente le personnage que nous incarnons n'est nul autre que lui-même, un homme négligeant, obnubilé par son travail, sombrant dans des vices tels que l'alcoolisme et rendant la vie de ses proches un enfer. L'histoire du jeu est tragique, où nous observons le déclin d'une jeune musicienne jusqu'à son suicide. Le veuf dans un élan macabre vient à mutiler le corps de sa défunte compagne pour l'immortaliser sur sa toile. Récolter les différents morceaux de sa muse, d'abord sa peau, son sang, ses os, ses cheveux, son doigt et enfin son œil relève d'une obsession perverse mais toujours animée d'un amour passionnel et sordide.

Nous ne voyons pas directement ces scènes, nous n'avons que le résultat de celles-ci, mais nous arrivons tout de même à être mal à l'aise et à contextualiser les événements grâce à l'ambiance du jeu, les indices qu'il présente accompagnés de doublage.

De même, nous ne connaissons pas le nom du protagoniste et certains éléments ne sont pas explicités, nous en savons juste assez pour comprendre et imaginer ce qu'il s'est passé et de s'impliquer dans l'histoire funeste de cette famille sans être noyé·e dans une mer d'informations.

Le jeu est poétique, en utilisant les thèmes de l'amour, de la famille, de l'art pour raconter un drame, de manière assez maline dans son storytelling, en exploitant un fort sentiment de mélancolie et de solitude.

En outre, les références artistiques sont exploitées à des fins analogiques, créant des parallèles avec les affections s'abattant sur cette famille, je pense notamment à *L'enlèvement de Ganymède* par Rembrandt, pouvant catégoriser la perte de la garde de sa fille, *Saturne dévorant l'un de ses fils* de Francisco de Goya, montrant que le fléau qui les ronge provient du père, *Judith décapitant Holopherne* de Artemisia Gentileschi faisant écho à la dégradation du corps de la conjointe par son époux, ou encore *Le Cauchemar* de Johan Heinrich Füssli, dont le titre reflète le quotidien des personnages.

Je ne pense pas que *Layers of Fear* soit le jeu adapté si nous cherchons seulement à frissonner, il raconte une histoire et a besoin que le joueur s'investisse dans le jeu pour mieux comprendre ses subtilités et toucher le joueur plus profondément. Il s'agit plus d'une expérience immersive qu'un jeu avec une grande jouabilité.

Pratique :

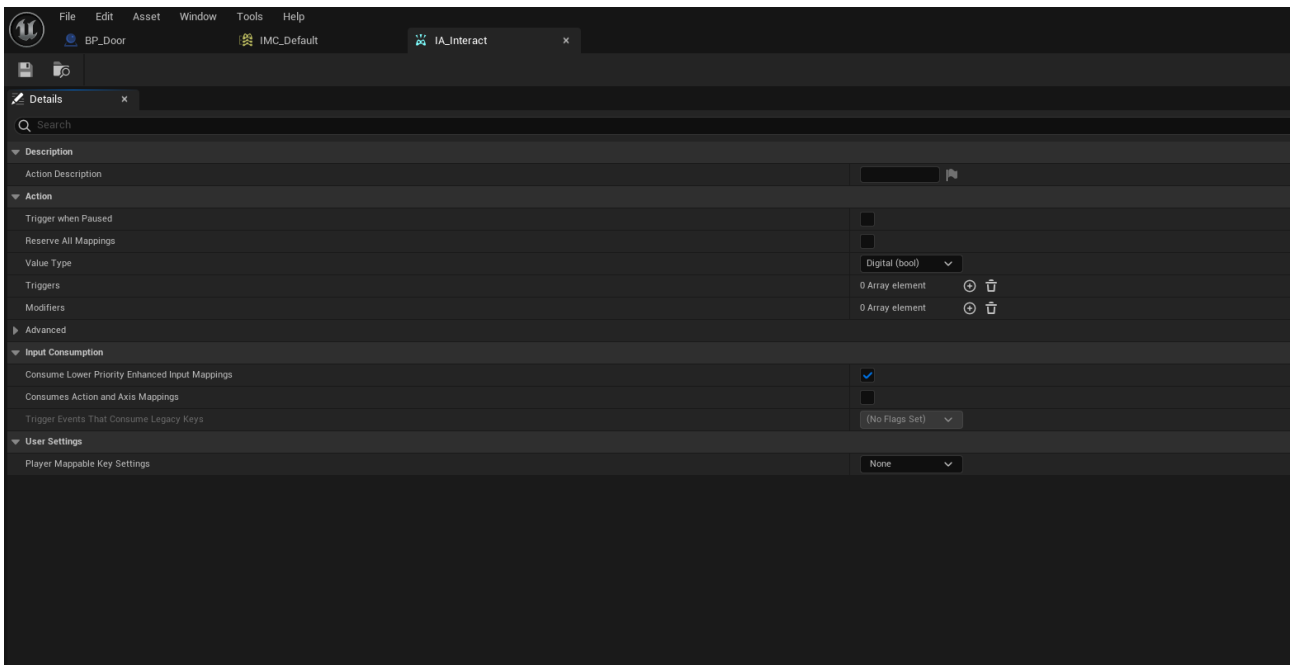
1. Définissez, selon vous, une des mécanique de gameplay principale du jeu, et reproduisez là sous le moteur de jeu Unreal Engine 5 (vous pouvez faire cette mécanique sur un projet de n'importe quel type, en 2D ou 3D, du moment que l'on "ressent" une similitude avec la mécanique originale).

J'ai décidé de reproduire la mécanique d'ouverture de porte en déplaçant la souris.

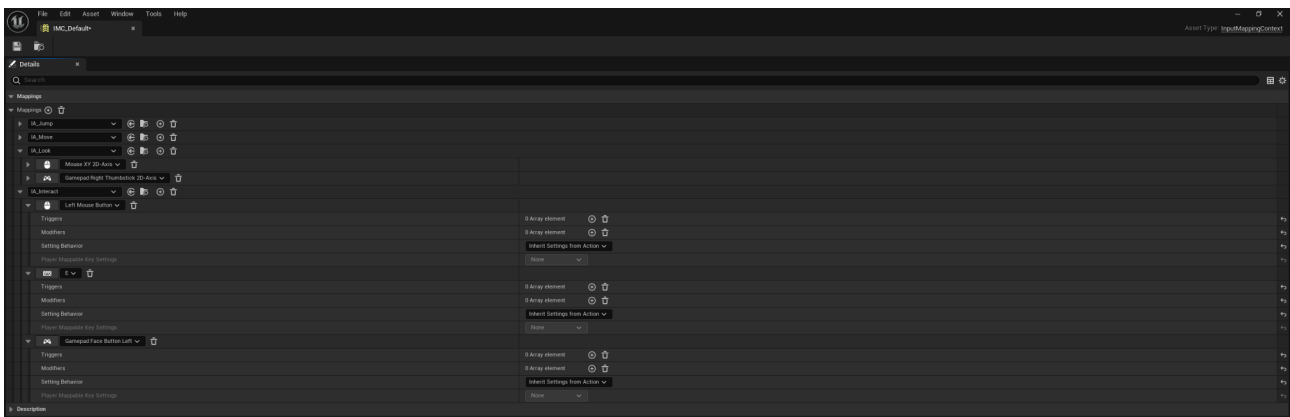
J'ai commencé sur la version 5.4.4 d'Unreal Engine avec le projet à la première personne comme base.

2. Décrivez, étape par étape, comment vous avez mis en place cette mécanique.

J'ai créé l'action d'interagir en plus de celles déjà fournies, à savoir se déplacer, sauter et bouger la caméra. Je l'ai configurée en tant que booléen car il n'existe que deux états à cette action, interagir en appuyant sur la touche correspondante ou ne pas interagir quand la touche en question n'a pas été touchée.

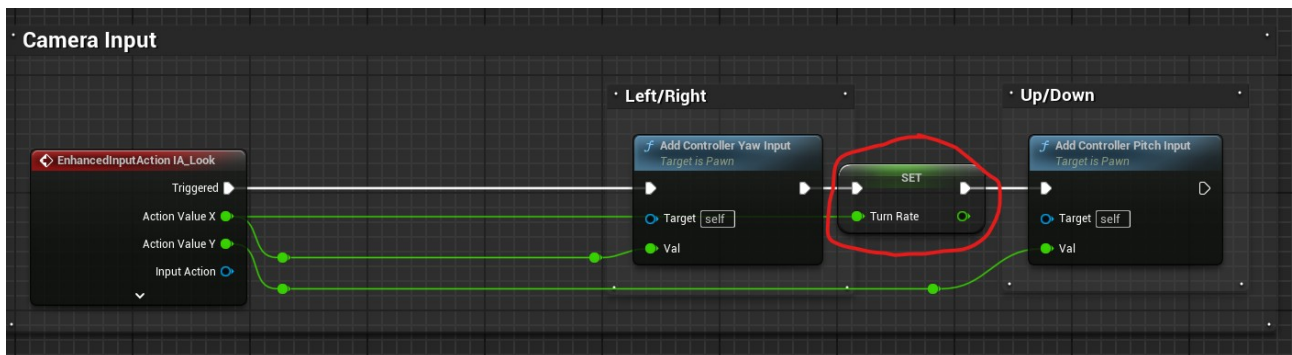


J'ai configuré cette action sur le clic gauche de la souris, la touche « E » ainsi que le bouton gauche d'une manette.



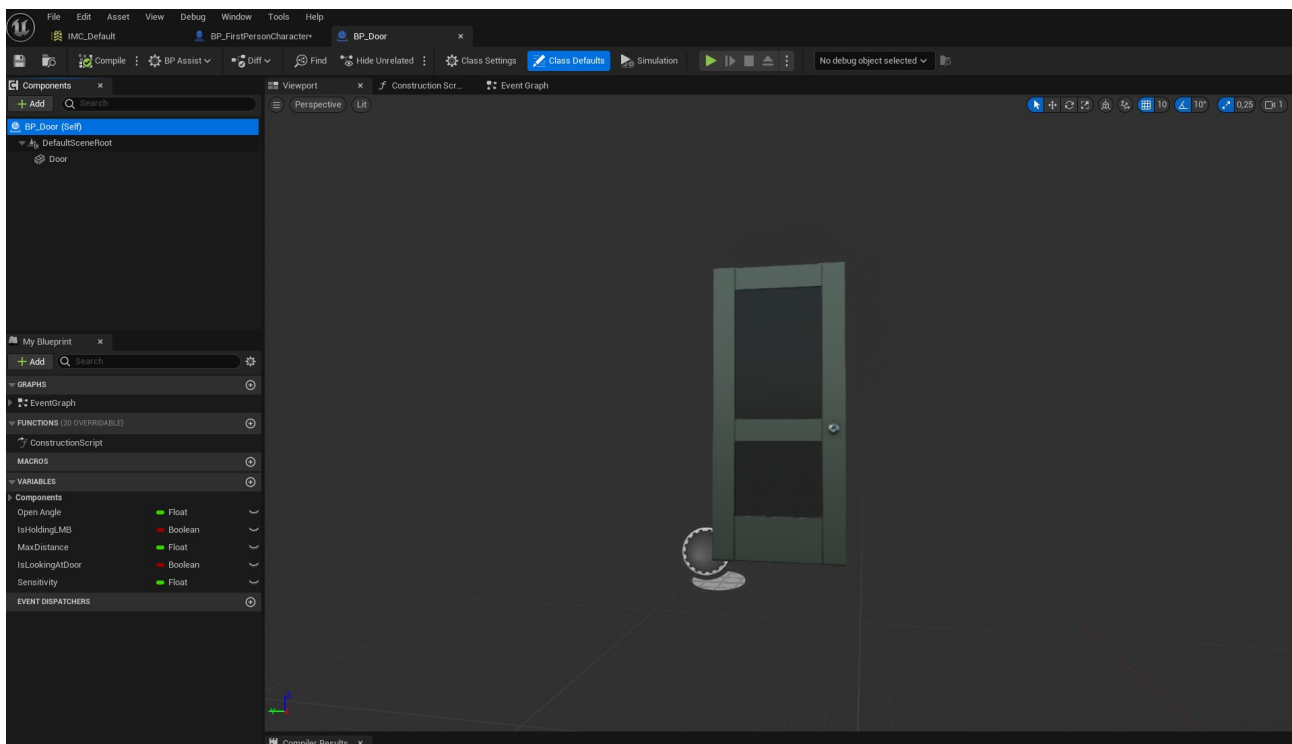
J'ai ensuite enregistré la valeur dont la caméra se déplace sur l'axe horizontal en tant que variable

(float, donc une valeur numérique pouvant être décimale) intitulée « Turn Rate » afin de m'en servir après pour déplacer la porte selon le déplacement de la caméra qui est contrôlé par la souris. Ainsi, lorsque le joueur déplacera la souris en interagissant, la porte tournera autour de son point de pivot en fonction du « Turn Rate » qui correspond donc au déplacement horizontal de la caméra.

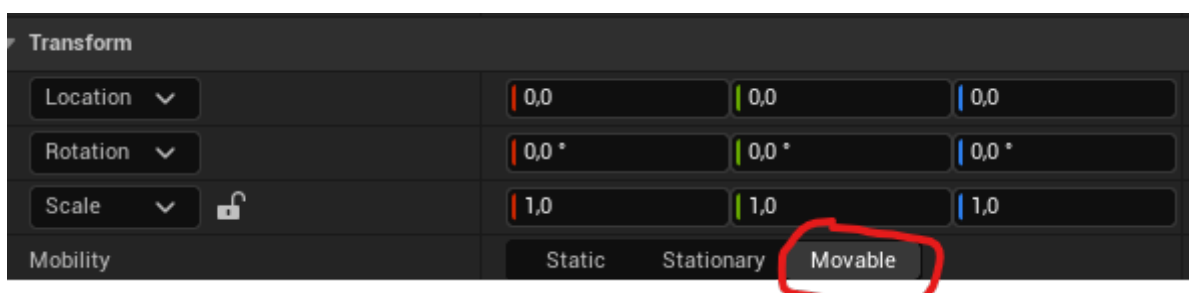


J'ai créé un nouveau Blueprint, nommé « BP_Door » auquel j'ai associé un mesh de porte dans le starter pack.

J'ai bien sûr au préalable donné une boîte de collision à la porte pour pouvoir interagir avec celle-ci.

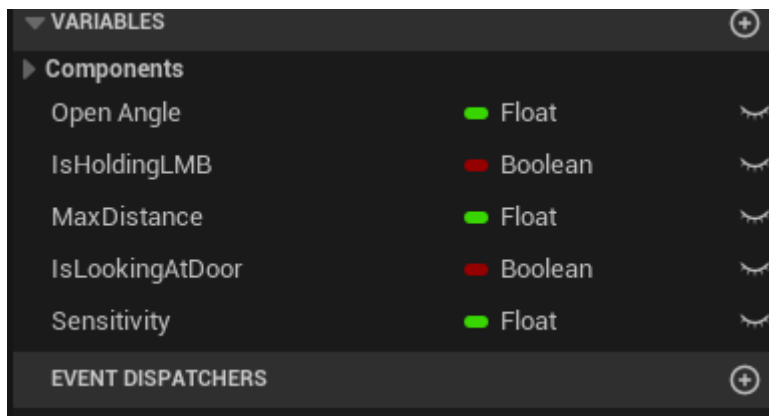


J'ai changé la mobilité du mesh en tant que « Movable » (amovable) afin qu'il puisse bouger et interagir avec le monde qui l'entoure.

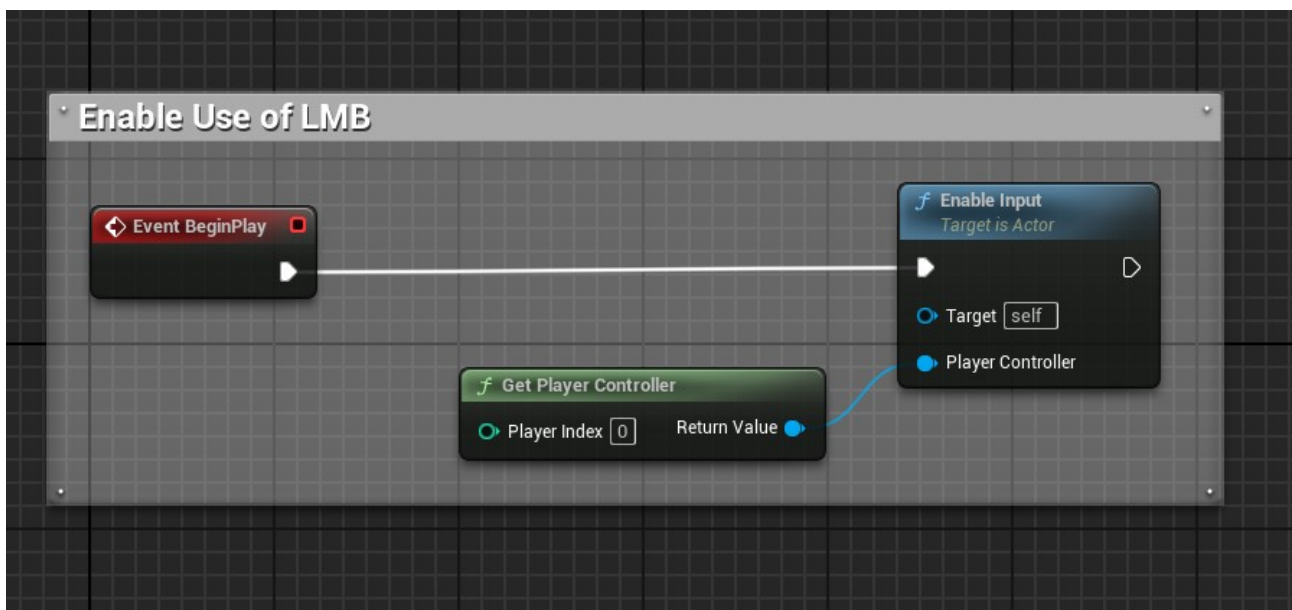


J'ai créé différentes variables pour ce Blueprint :

- « Open Angle », un float qui a pour but de définir l'angle maximal auquel la porte pourra s'ouvrir sur l'axe Z, donc vertical, en fonction du point de pivot de la porte, situé vers l'intérieur de son cadre, opposé à la poignée.
- « IsHoldingLMB » (maintien du clic gauche de la souris ?), un booléen qui détermine si le clic gauche est appuyé ou non, et sera donc vrai ou faux respectivement.
- « Max Distance », un float dont la fonction est de déterminer une distance maximale à laquelle le joueur pourra interagir avec la porte. Plus cette valeur sera élevée, plus le joueur pourra interagir avec la porte de loin.
- « IsLookingAtDoor » (regard vers la porte ?), un booléen qui détermine si le joueur regarde en direction de la porte ou non, et sera donc vrai ou faux respectivement.
- « Sensitivity » (sensibilité), un float qui servira à modifier l'influence du mouvement de la souris pour faire pivoter la porte. Pour un mouvement de souris de même intensité, la porte s'ouvrira grandement si cette valeur est très élevée, tandis qu'à l'inverse, si cette valeur est faible, alors la porte s'ouvrira faiblement.



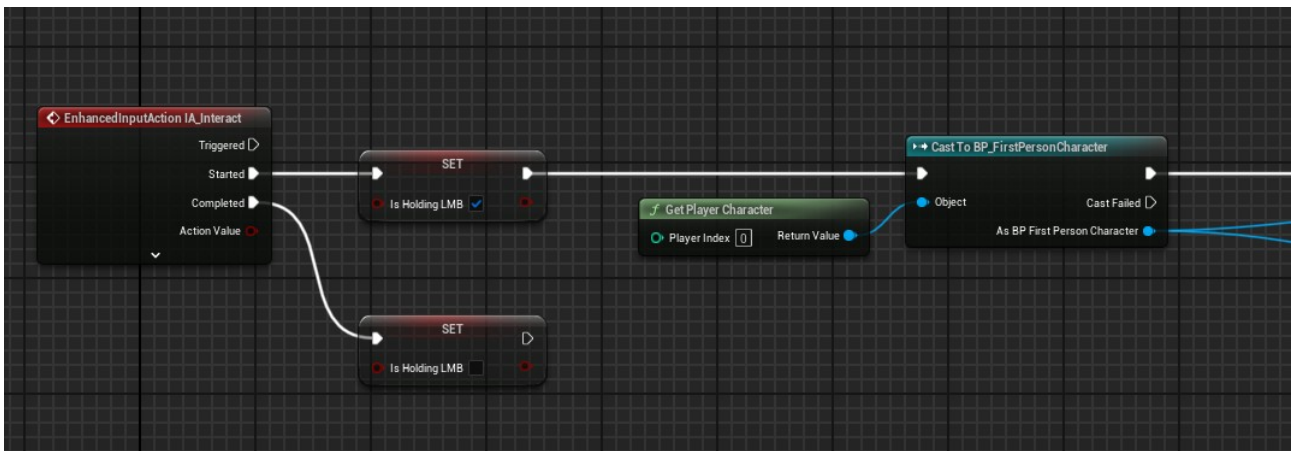
Ici, je permets au joueur d'utiliser le clic gauche dès que le jeu se lance.



Avec le premier node, ce qui s'en suit dépendra si j'appuie sur la touche correspondante et comment j'appuie sur la dite touche, à savoir « E », le bouton gauche d'une manette ou le clic gauche de la souris. Par raison de simplicité, nous n'évoquerons que cette dernière par la suite.

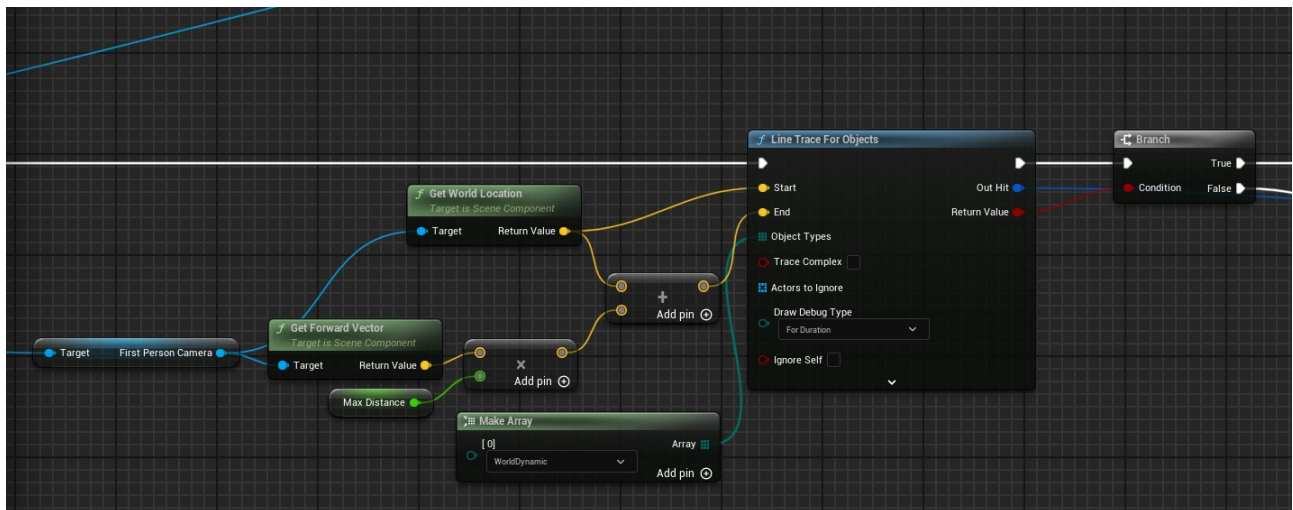
Lorsque nous appuyons sur le clic gauche, « IsHoldingLMB » sera considéré vrai, alors qu’une fois que le clic gauche est relâché, « IsHoldingLMB » deviendra faux. Il est à noter, que par défaut, lorsque le jeu se lance, le booléen est initialement considéré faux.

S’il est vrai, alors nous récupérons le personnage et son Blueprint, de ce dernier, nous récupérons le « Turn Rate » évoqué plus tôt ainsi que sa caméra.

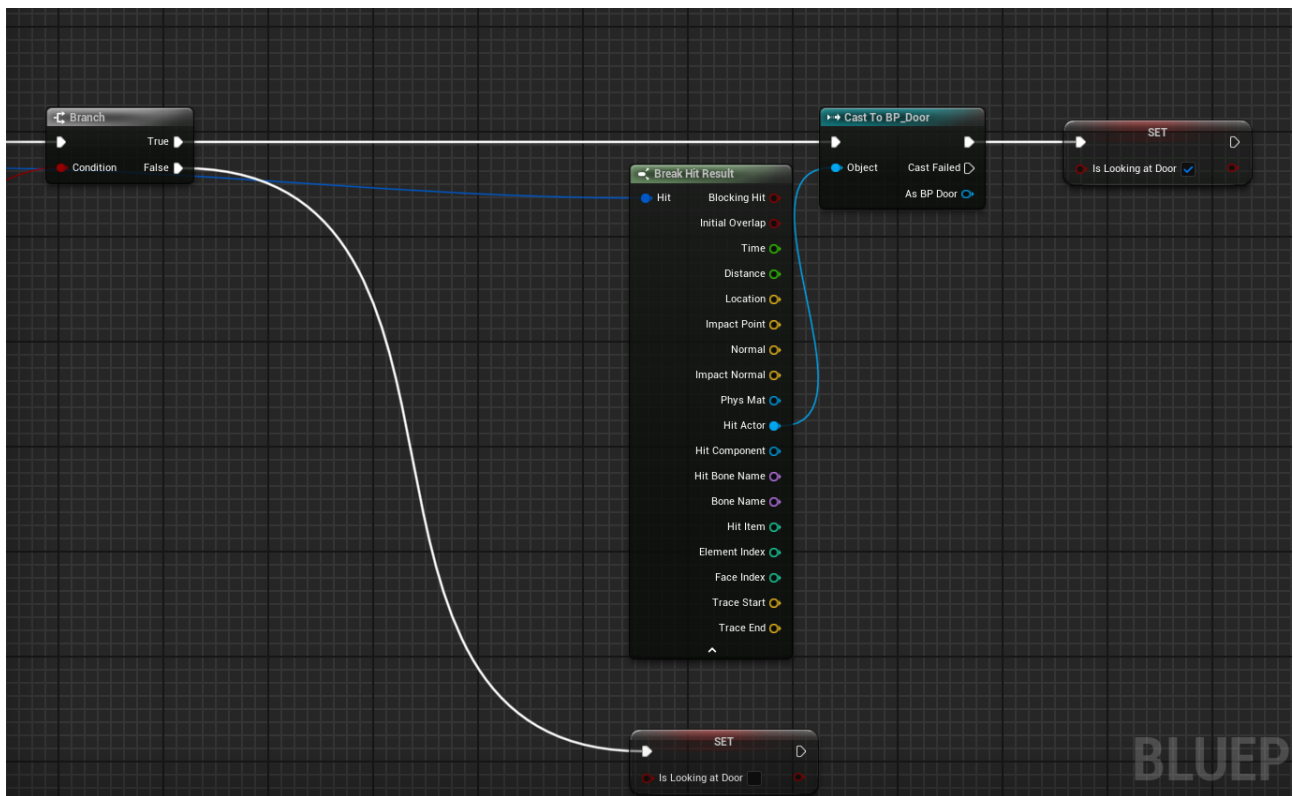


À partir de la caméra du personnage, nous récupérons sa position dans le monde et un vecteur qui part vers l’avant. Sa position est la même que celle du personnage, elle nous servira de point de départ afin de déterminer la distance qui sépare le personnage de l’objet avec lequel il peut interagir. Nous multiplions le vecteur qui part vers l’avant avec « Max Distance » pour établir que la distance définie dans cette variable sera celle d’un vecteur droit, partant à partir du personnage vers l’avant. Nous additionnons ce résultat avec la position de la caméra qui servira de point d’arrivée de la distance qui sépare le personnage de l’objet avec lequel il peut interagir.

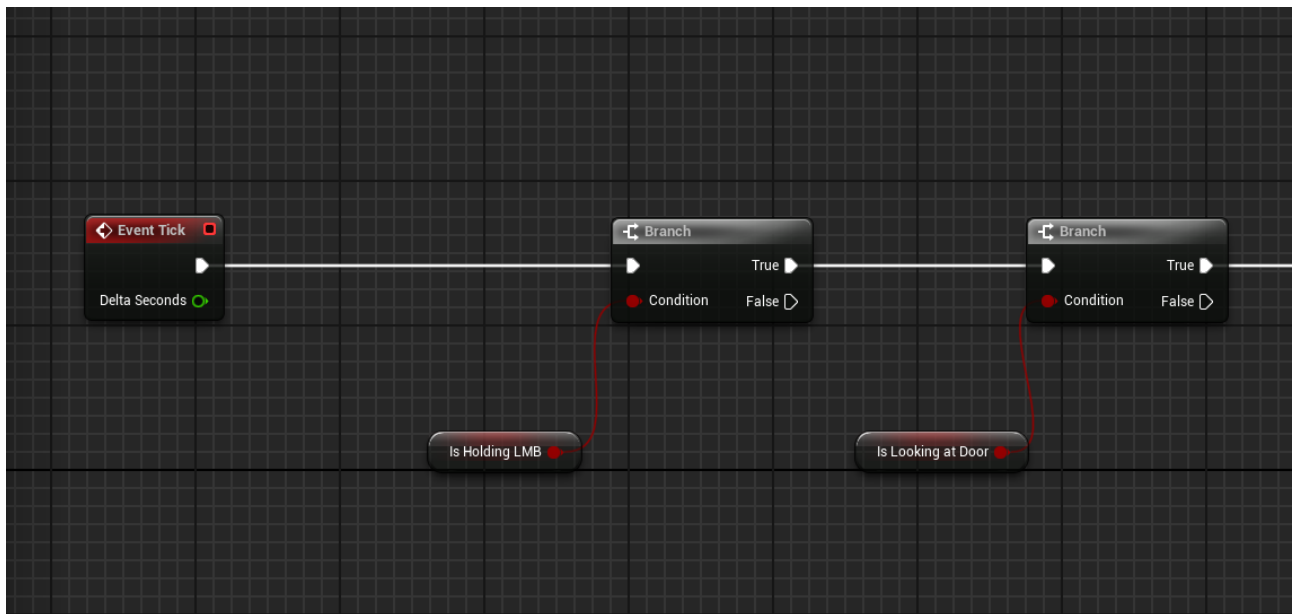
Si la différence du point d’arrivée et du point de départ, à savoir un vecteur vers l’avant de longueur « Max Distance » en partant du personnage, est positive, alors le joueur peut interagir avec l’objet, sinon, si la valeur est négative, alors il est trop loin et ne peut pas interagir avec l’objet.



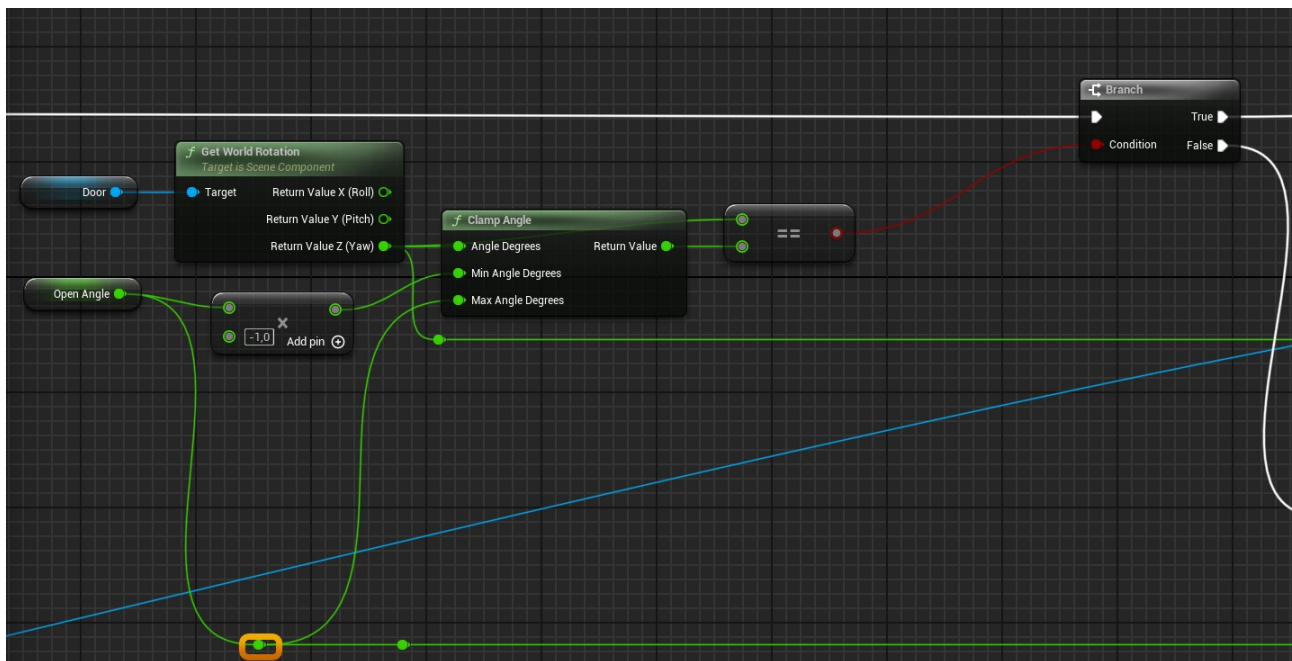
Si la distance est positive, et qu’un objet est touché par ce vecteur projeté, plus spécifiquement un acteur, et si l’acteur en question est BP_Door, alors nous considérons « IsLookingAtDoor » vrai. Autrement, « IsLookingAtDoor » sera considéré faux.



Ensuite, à tout moment, si « IsHoldingLMB » et « IsLookingAtDoor » sont vraies alors nous pouvons continuer à l'étape suivante.
Autrement, la suite du Blueprint ne s'effectuera pas.



Ici, nous prenons la rotation de la porte sur l'axe Z, nous allons lui donner des valeurs limites avec le node « Clamp Angle », en valeur minimale -100° et 100° en valeur maximale. Si ces valeurs sont dépassées, alors nous ne pourrons plus faire pivoter la porte. Nous ne pourrons que la faire pivoter si sa rotation est comprise entre -100° et 100° sur l'axe Z.



Ensuite, nous récupérons la valeur de « Turn Rate » précédemment évoquée à partir du Blueprint du personnage à laquelle nous multiplions « Sensitivity », modifiant la valeur de « Turn Rate » en fonction du ressenti que nous voulons donner. Je réitère que pour un mouvement de souris de même intensité, la porte s'ouvrira grandement si cette valeur est très élevée, tandis qu'à l'inverse, si cette valeur est faible, alors la porte s'ouvrira faiblement.

Nous donnons le résultat obtenu au node « Make Rotator » qui comme son nom l'indique va créer une rotation, en l'occurrence sur l'axe Z dans notre cas et nous rajoutons le résultat à la rotation locale de la porte.

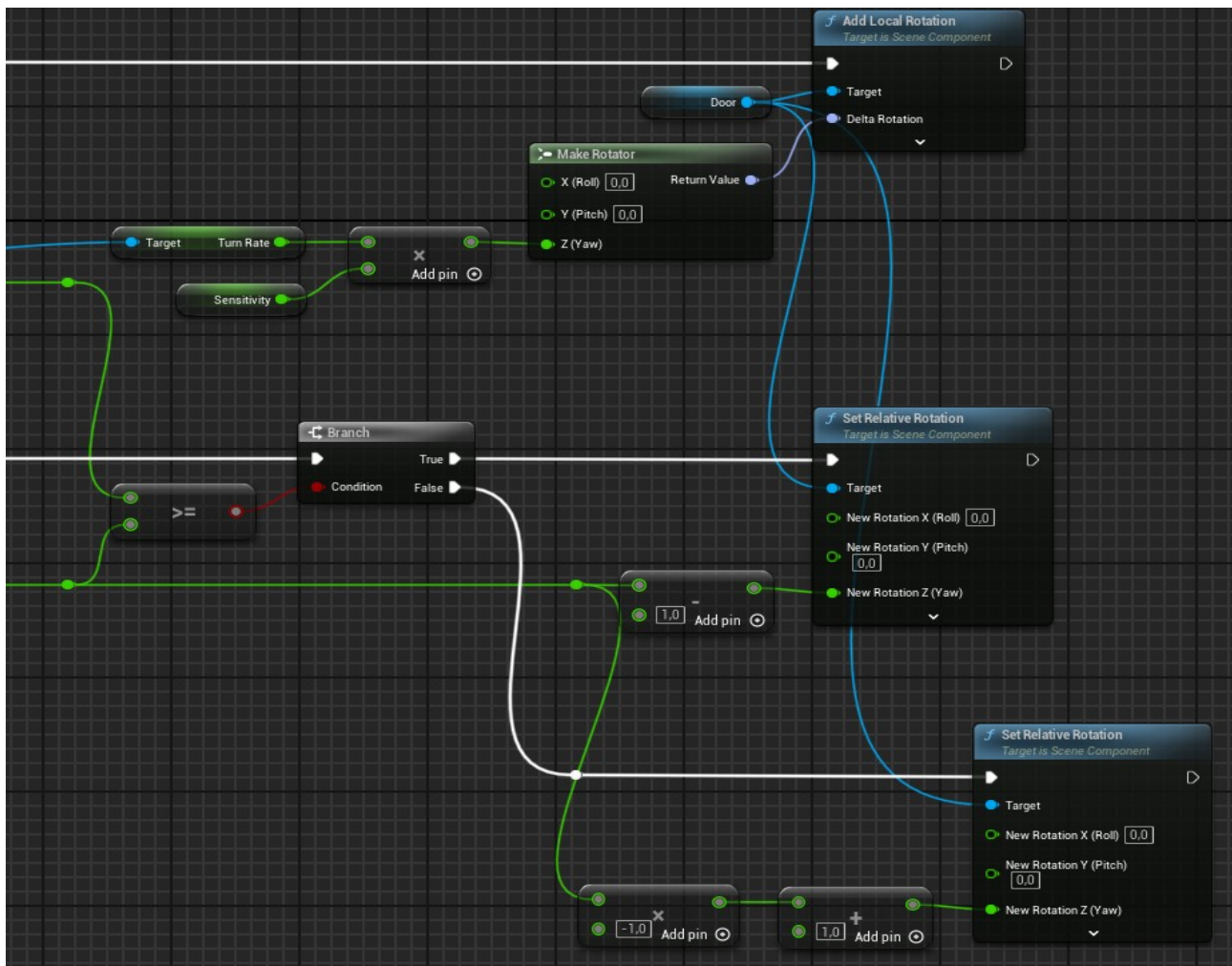
Ainsi, tant que la rotation de la porte sur l'axe Z est comprise entre -100° et 100° , nous pouvons augmenter ou diminuer sa rotation en fonction de « Turn Rate » et de « Sensitivity ». Il est important de préciser que nous rajoutons le résultat à la rotation de l'objet et ne lui changeons pas sa valeur avant de la faire tourner. Sinon, nous repartirions toujours de 0° , alors que nous pouvons faire tourner la porte peu importe sa rotation initiale tant que celle-ci est comprise entre -100° et 100° .

Ce n'est pas fini, puisque nous ne pouvons pas faire pivoter la porte si sa rotation est égale ou excède 100° ou -100° , une fois ces valeurs atteintes, nous ne pouvons plus la faire pivoter.

C'est pourquoi si la rotation de la porte est supérieure ou égale à 100° , nous lui donnons une nouvelle valeur de rotation à savoir « Open Angle », donc 100° à laquelle nous retirons 1, pour un résultat de 99° afin que nous puissions toujours la faire pivoter, car sa valeur de rotation sera à nouveau comprise entre -100° et 100° .

Il en va de même si la rotation de la porte est inférieure ou égale à -100° , nous lui donnons une nouvelle valeur de rotation à savoir « Open Angle », donc 100° que nous multiplions par -1, ce qui fait -100° à laquelle nous ajoutons 1, pour un résultat de -99° afin que nous puissions toujours la faire pivoter, car sa valeur de rotation sera à nouveau comprise entre -100° et 100° .

Ce n'est peut-être pas optimal et lorsque nous réinitialisons la rotation de la porte à 99° ou -99° , c'est visible et cela peut paraître bizarre, l'immersion se brise.



Toutefois, cette approche est perfectible, lorsque j'ajoute une nouvelle porte dans la scène, les deux portes s'ouvrent simultanément alors qu'une seule, celle au centre de notre curseur devrait s'ouvrir. Je présume que cela est dû au fait que le mesh de la porte s'ouvre, et ceci pour tous les acteurs qui contiennent ce mesh et non seulement l'instance de la porte elle-même.

J'ai essayé de remplacer la porte par l'acteur touché par le vecteur. Cependant, je n'ai pas réussi à faire en sorte que cela fonctionne.

Aussi, le mouvement de porte est synchrone avec celui de la caméra, ainsi, lorsque le joueur ouvre la porte et déplace la souris d'un côté ou d'un autre, la caméra se déplace aussi, et cela, peut-être déroutant.

3. Expliquez en quoi cette mécanique est primordiale tout au long du jeu.

Cette mécanique me semble primordiale, car nous sommes amenés à ouvrir des portes tout au long du jeu, nous déplaçant de salles en salles diverses et variées.

Non seulement comme explicitée déjà à maintes reprises, cette mécanique, reflétant une véritable ouverture de porte, accorde une meilleure immersion et implication du joueur dans ce monde, elle permet d'accentuer la peur du joueur.

En effet, il est impossible de savoir ce qui se cache derrière la porte, plus particulièrement quand les pièces semblent se déplacer, invalidant l'expérience et les connaissances acquises du joueur.

Cette incertitude constante, cette peur de l'inconnu, le suspens culminant de par l'ambiance pesante du jeu contribuent à maintenir en haleine le joueur à chaque porte. Il peut être terrifié à l'idée de ce qu'il peut y avoir derrière et par conséquent à redouter les portes elles-mêmes.

Ceci, couplé à la nécessité d'interagir avec elles, par un mouvement de main, empêche l'idée de se soustraire et de se détacher du jeu.

C'est pourquoi cette mécanique me semble essentielle à *Layers of Fear*, car elle contribue à la tension et l'angoisse constante du joueur, et aussi à cacher et masquer la manière dont les pièces se déplacent, qui est un autre élément crucial du jeu.

4. Pourquoi cette mécanique vous a-t-elle plu ?

Cette mécanique m'a plu pour les raisons énoncées précédemment, donc l'immersion accrue, le réalisme et l'effroi que cela peut générer d'une manière simple, indirectement et sans chercher à être volontairement effrayant. Je trouve aussi qu'elle complémente bien les autres mécaniques, en ayant un rôle central bien que présenté initialement de manière anodine.

De plus, c'était la première fois que je rencontrais cette mécanique, et j'ai été très agréablement surpris·e de son implémentation.