

Université Paris 8

Master Création Numérique

parcours : Arts et Technologies de l'Image Virtuelle

Génération d'un gameplay et d'un décor dynamique de jeu vidéo à partir du profil comportemental du joueur

Suzanne Tosun



Mémoire de Master 2, 2016 - 2017

Remerciements

Je tiens à remercier l'ensemble des personnes qui ont contribué à la réussite de mes années de formation en Arts et Technologies de l'image :

Tout d'abord, Mme. Chu-Yin Chen et Mr. Jean-François Jégo, qui m'ont encadrés tout au long de ce mémoire et qui m'ont aidé à organiser mes idées.

Tout particulièrement Mme. Chu-Yin Chen, dont les cours d'élicitation m'ont été très utiles ainsi que Mr. Cédric Plessiet, pour m'avoir orienté dans mes recherches techniques et pour le temps qu'il m'a accordé à m'enseigner les concepts de programmation et d'intelligence artificielle liés à ma problématique.

Merci également à tous les enseignants pour leur enseignement et leur avis professionnel ainsi que mes camarades de classe pour avoir partagé avec moi des moments de débat, d'histoire, de conseils et d'échanges d'idées.

Enfin, je remercie toutes les personnes qui ont contribué à la relecture de ce mémoire et qui m'ont soutenu durant toutes ces années de formation.

Ces trois années ont été particulièrement enrichissantes tant d'un point de vue personnel que professionnel. D'abord, pour les connaissances théoriques qui nous ont été enseignées ; puis pour avoir été en contact avec un grand nombre de personnes dont les observations et les jugements ont enrichi mes réflexions et mon développement personnel.

Résumé

Dans le monde réel, nos actes sont le résultat de ce que nous sommes, ce que nous ressentons et pensons. Mais en est-il de même dans les mondes virtuels ? Notre comportement est-il le reflet de notre personnalité ? Pour beaucoup de joueurs, vivre une expérience vidéoludique leur permet de s'émanciper de toute pression sociale, morale et physique afin de vivre d'incroyables aventures émotionnelles. Les jeux vidéo sont d'autant plus satisfaisants qu'ils permettent aux joueurs d'atteindre différents états émotionnels, souvent vécus de manière intense, plus fréquemment, facilement et rapidement que dans la réalité. Mais pour combler la soif d'aventure des joueurs, les concepteurs de jeux doivent construire un environnement régi de règles, peuplé d'ennemis, offrant divers objectifs à atteindre afin de permettre l'immersion du joueur au travers de ses interactions avec cet univers. Comprendre les joueurs, leurs besoins et leur personnalité, est nécessaire aux concepteurs afin d'offrir une expérience optimale. C'est pourquoi, depuis plusieurs années, des études psychologiques orientées vers les joueurs de jeux vidéo, tendent à comprendre la relation entre le comportement des joueurs dans ces univers virtuels et leur personnalité réelle. Si ces études aident les game et level designers à adapter l'environnement et le contenu général de leur projet, peu de jeux encore se concentrent sur des environnements et des capacités de personnage virtuel qui s'adaptent en fonction du comportement du joueur. C'est-à-dire de donner au joueur les actions qui conviendraient le plus à son comportement. Ainsi, chaque joueur aurait sa propre manière de jouer, ses propres actions correspondant à ce qu'il aime être. C'est pourquoi, j'ai choisi d'aborder dans ce mémoire, des méthodes pouvant permettre aux jeux vidéo de générer des actions et des décors s'adaptant à la manière de jouer des joueurs.

Abstract

In the real world, our actions are the result of who we are, our feelings and our thoughts. But is it the same in the virtual world ? Most of players dive in video games experiences to emancipate themselves from social, moral and physical pressures and live amazing emotional adventures. Video games are all the more pleasing since they allow players to reach intensive emotional states, more frequently, easily and quickly than reality. To satisfy the player's thirst of adventure, designers have to build environments governed by rules, populated by enemies and giving many goals to reach in order to immerse the player in this universe through his actions. Understand players, their needs and their personality is necessary for designers to offer an optimal experience. That's why, since many years, psychological video games studies try to understand the link between player's behavior in games and their real personality. Even if studies help designers to adapt their games environment and content to players, just few games change their environment and give skills to the player's character depending on player's behavior. Which mean, give to player the actions that suit the most with his behavior. Thus, each player could have his own way of playing, his own actions which suit with what he really is. That's why I chose to approach the different methods which can allow video games to generate actions and environment which adapt themselves to player's way of playing.

Sommaire

Introduction générale	6
I : Etat de l'art	8
<u>Introduction</u>	8
I.1) L'approche psychologique	10
I.1.1) Les besoins humains comme facteur motivationnel	10
I.1.2) Les jeux vidéo : un monde parallèle	19
I.1.3) Etude du comportement des joueurs dans les univers virtuels : quand un comportement virtuel dénote une personnalité réelle	26
I.2) Intégrer la personnalité des joueurs dans un jeu	32
I.3) Les jeux vidéo qui ont inspirés ce mémoire	37
<u>Conclusion</u>	44
II : Développement d'un système génératif	45
<u>Introduction</u>	45
II.1) Expérimentations personnelles	47
II.1.1) Appréhender le joueur	47
II.1.2) Quantifier les actions du joueur	51
II.1.3) Conception des règles d'évolution des actions du joueur	55
II.1.4) Génération du décor d'un niveau de jeu vidéo	60
II.2) Transcendance : expérimentation en groupe	65
II.3) Les évolutions et les risques	71
<u>Conclusion</u>	74
Conclusion Générale	75
Bibliographie	77
Webographie	77
Jeux vidéo cités	79
Table des illustrations	80
Index des termes	82
Annexes	84

Introduction générale

Comment pourrai-je débattre de l'univers du jeu vidéo sans évoquer le lien que j'entretiens avec celui-ci ? Côté quotidien ce milieu, je multiplie depuis de nombreuses années les expériences interactives. Du jeu de rôle (nommé RPG ou MMORPG pour les jeux de rôles multijoueurs), au jeu de tir à la première personne (FPS) en passant par le jeu de plateformes, de stratégie, de rythme (jeu musical), de course, etc. Je pensais au départ, comme cette vision courante, que le jeu vidéo n'était qu'une échappatoire permettant à l'être humain de s'extirper de son monde réel afin de cacher et d'oublier toute éventuelle pression sociale. Tout cela jusqu'à ce que je m'aperçoive que l'impact émotionnel d'un jeu sur un joueur peut être excessivement fort. Et donc que cette excuse "d'échappatoire virtuelle" n'expliquait plus suffisamment l'implication des joueurs dans les jeux vidéo pour me satisfaire.

Pourquoi jouons nous et qu'est ce qui nous motive à jouer ? Si les émotions ressenties grâce aux jeux vidéo sont réelles cela veut-il dire qu'elle résultent des actes virtuels du joueur qui eux-même sont influencés par son profil comportemental et sa personnalité réelle ? Si c'est le cas et que l'on place différents joueurs dans une même situation avec un même objectif, il devrait y avoir autant de réactions possibles face à cette situation qu'il y a de joueurs (en considérant qu'il y a un choix d'interactions illimité). Et donc autant de gameplay différents.

Dans les jeux vidéo actuellement développés les actions que le joueur peut exécuter au travers des compétences de son personnage comme sauter, tirer, esquiver, etc, sont définies dès le départ par les game designers. Ces compétences évoluent pas ou peu au cours du jeu. Le gameplay est donc orienté vers un type d'interaction avec le décor du jeu, les ennemis ou les autres joueurs. Parfois la fonctionnalité d'un objet peut changer comme l'arc de Tomb Raider qui au bout d'une certaine avancée dans le jeu peut non seulement tirer des flèches mais également tirer une corde accrochée à une flèche pour former une tyrolienne ou fournir un moyen d'accéder à des lieux plus élevés. Dans les jeux de rôle, les joueurs peuvent parfois choisir entre différentes classes de personnages telles que mage, guerrier, archer, etc. Ces classes orientent également le gameplay du joueur et le forcent à jouer avec des compétences définies comme appartenant à la classe choisie. Même si, au travers de ce choix de classe, le joueur peut choisir le type de gameplay correspondant le plus à sa "manière de jouer", son profil comportemental, le gameplay reste défini et limité. Que se passe-t-il si j'ai envie de jouer un guerrier qui peut jouer avec des compétences magiques et qui de temps à autre aime surprendre l'adversaire en s'infiltrant dans sa base grâce à des compétences d'invisibilité ? Peu de jeux permettent actuellement aux joueurs de mixer les différents types de compétences pour créer son propre gameplay.

Dans ce cas ne serait-il pas intéressant de développer un gameplay unique à chaque joueur mais découlant d'un même système ? C'est-à-dire qu'au lieu de programmer des actions fixées dès le départ par les game designers et qui n'évolueront pas ou peu durant l'intégralité du jeu, on programmerait un ensemble d'actions et d'interactions avec le décor, les ennemis ou les autres joueurs qui ne seraient qu'en partie disponibles au commencement du jeu. En fonction des actions que le joueur va utiliser dans le jeu, le gameplay du joueur pourrait se développer : les compétences de son personnage évolueraient jusqu'à débloquent de nouvelles compétences. Et pour aller plus loin, le décor du jeu devrait pouvoir s'adapter aux actions utilisées et débloquent par le joueur. Tout cela dans le but de voir émerger un

gameplay propre à chaque joueur.

C'est donc naturellement que j'oriente ma problématique vers la génération d'un gameplay et d'un environnement capable d'évoluer en fonction des actes du joueur dans le jeu.

Pour cela, nous devons savoir s'il est possible de "calculer" le degré d'utilisation des compétences du personnage dans un jeu. Et si oui, comment le calculer et répercuter les résultats sur l'évolution des compétences et la génération du décor. C'est ce que je cherche à expérimenter au travers de ma problématique de mémoire : Comment générer un gameplay et un décor dynamique de jeu vidéo à partir du profil comportemental du joueur ?

Dans un premier temps, je souhaite aborder cette problématique sous l'angle psychologique de la personnalité et du comportement humain. Car pour comprendre les agissements d'un joueur dans un environnement virtuel, nous devons d'abord comprendre les besoins psychologiques humains et ce qui le motive à agir. Grâce à cela, nous pourrons réaliser un rapprochement avec l'univers des jeux vidéo et comprendre comment ils arrivent à satisfaire ces besoins et par quels moyens.

Puis au travers d'une étude de l'art des travaux réalisés en intelligence artificielle sur le comportement humain, je parlerai de l'adaptation possible du profil comportemental du joueur en données numériques. Quelles techniques de programmations et d'intelligences artificielles peuvent permettre à un jeu de "calculer" le profil comportemental d'un joueur ? Je veux également aborder les jeux qui ont inspiré et motivé mon travail sur cette problématique.

Et enfin j'illustrerai toutes mes recherches au travers d'expérimentations que j'ai réalisées durant cette année de master 2. J'analyserai les points d'amélioration et évoquerai une possible évolution du projet.

I : Etat de l'art

Introduction

De l'amusement à la tristesse, de l'émerveillement à l'excitation, pour se divertir ou bien apprendre, les jeux vidéo plongent le joueur au coeur d'univers fantastiques, lui faisant vivre de nombreuses expériences uniques et éveillant en lui une multitude d'émotions. Mais pour imaginer ces environnements virtuels, les game et level designers doivent se représenter le joueur. Comprendre ses besoins, ses attentes, sa personnalité, les liens qu'il entretient avec l'univers du jeu et le(s) personnage(s), afin de concevoir une expérience lui apportant satisfaction, aussi bien en matière de challenge et de difficulté, mais également en ce qui concerne les interactions avec son propre personnage ou son environnement.

C'est pourquoi, depuis quelques années, les études psychologiques liées aux joueurs pour comprendre leur comportement et leur personnalité dans les jeux se multiplient. Elles soutiennent l'hypothèse, selon laquelle la personnalité et le comportement des joueurs dans leur vie réelle et leur vie vidéoludique comporterait des similitudes. Au travers de cela, comprendre ces deux aspects de la psychologie des joueurs dans ces deux mondes distincts permettraient de créer des environnements virtuels et un gameplay bien plus adaptés aux joueurs. Ceux-ci correspondant bien mieux à leurs attentes en matière d'actions à réaliser ou encore d'objectifs à accomplir et leur proposant des expériences plus immersives, satisfaisantes et même jubilatoires.

Chacun de ces aspects étant une pièce constituant le puzzle que représente le jeu vidéo et permettant de créer une expérience intéressante, je vais orienter ce mémoire sur le gameplay du joueur et plus particulièrement sur les actions que le joueur peut réaliser dans un jeu. Pour être plus précise, ce que je nomme "action" représente en réalité toutes les compétences du personnage que le joueur va pouvoir utiliser. Que ce soit avec l'environnement ou contre un ennemi. Pour illustrer, il peut s'agir de : grimper, sauter, esquiver, empoisonner, se mettre à couvert, se cacher, lancer des boules de feu, ralentir un ennemi, etc.

Pour ce faire je me dois de clarifier quelques termes :

Le gameplay :

Il n'existe pas de définition universelle du mot "gameplay". Dans sa publication *"Gameplay and game mechanics design : a key to quality in video games"*, Carlo Fabricatore aborde le gameplay sous le point de vue des joueurs et ce qu'il représente pour eux.

"Hence, a player-centered approach can lead to define gameplay as the set of activities that can be performed by the player during the ludic experience, and by other entities belonging to the virtual world, as a response to player's actions and/or as autonomous courses of action that contribute to the liveliness of the virtual world. "

"Par conséquent, une approche centrée sur le joueur peut conduire à définir le gameplay comme l'ensemble

des activités qu'un joueur peut réaliser durant son expérience ludique et par les autres entités appartenant au monde virtuel, comme une réponse aux actions du joueur et/ou comme un cycle d'action autonome qui contribue à rendre le monde virtuel vivant”.

Carlo Fabricatore
Gameplay and game mechanics design : a key to quality in video games

Ainsi ce que les joueurs appellent communément le gameplay, regroupe toutes les actions qu'ils peuvent réaliser dans un jeu ainsi que les réponses de l'environnement en réponse à leurs actions : nommé feedback. Ces réponses peuvent être liées au décor : je tire sur une caisse en bois, elle est détruite. Ou bien liées aux entités vivantes du jeu comme le PNJ ou personnage non joueur : je tire sur un ennemi, je vois l'impact des balles et lorsque sa barre de vie est vide, je le vois mourir via son animation, le son, etc. A cela s'ajoute les activités des entités du monde virtuelle qui ont un comportement sans que le joueur interagisse avec elles. Ce comportement participe à la vie du monde virtuel et le rend crédible au regard du joueur. En général, pour les joueurs, le gameplay est un élément clé significatif de la qualité du jeu.

Dans ce mémoire je me concentre uniquement sur les actions que le joueur peut réaliser dans un jeu en accord avec son profil comportemental et non sur le gameplay dans tous ses aspects. Par conséquent lorsque je parle “d'émergence” ou de “laisser émerger”, je souhaite signifier que l'apparition de nouvelles actions pour le joueur est liée à ses choix d'actions précédemment réalisés et non à un choix des concepteurs de donner une nouvelle action réalisable par le joueur lorsqu'il a atteint un certain stade du jeu.

Générer :

Lorsque j'emploie les mots générer ou génération, je fais référence à la génération procédurale. C'est-à-dire à la capacité du jeu de créer en grande quantité, des fonctionnalités ou des ressources de manière automatisée en répondant à des règles codées dans l'algorithme permettant la génération. Dans mon cas ce sera soit des actions que le joueur pourra utiliser au travers des compétences de son personnage, soit des éléments du décors comme des “cover” (des éléments permettant de se mettre à couvert), des plateformes, des éléments magiques, etc.

Décor dynamique :

Il s'agit de tout élément constituant le décor du jeu ou l'environnement 3D qui peut être généré.

I.1) L'approche psychologique

Lorsqu'un joueur joue à un jeu vidéo, un accord tacite se crée entre lui et le jeu. C'est-à-dire que le joueur accepte d'entrer dans l'univers du jeu, de répondre à ses règles et de se laisser guider narrativement par celui-ci. D'un autre côté le jeu doit répondre aux attentes du joueur : son univers doit être crédible, sa difficulté, progressive donnant suffisamment de challenge au joueur sans pour autant être insurmontable, les actions du joueur doivent lui permettre d'évoluer librement dans le jeu ou en tout cas, lui en donner l'impression. Pour répondre à ces attentes, les game et level designers simulent, se représentent une version mentale et cognitive du joueur : ses besoins et sa capacité physique de répondre correctement à un stimulus du jeu. En d'autres termes comprendre la relation que le joueur va entretenir avec le jeu. Mais cette relation avec le monde virtuelle est-elle si différente de la relation avec le monde réel ? Quels besoins psychologiques les jeux apportent-ils au joueur ? Quelles sont les raisons amenant une personne à jouer à un jeu vidéo ? Pour répondre à ces interrogations nous devons nous pencher sur l'humain, ses besoins et ses motivations, sa représentation du jeu vidéo et dans le jeu vidéo.

Dans cette partie de mon mémoire je vais aborder les travaux réalisés par des psychologues ou designer. Certains ayant lié leur travail sur la psychologie avec les jeux vidéo, d'autres se concentrant uniquement sur la psychologie de l'être humain, je ferai le rapprochement avec des situations et des expériences vécues dans des jeux vidéo.

I.1.1) Les besoins humains comme facteur motivationnel

A chaque instant d'un jeu le joueur est stimulé émotionnellement et physiquement pour agir face à une situation donnée, à un instant donné. Il décide inconsciemment et instinctivement de l'action qu'il va réaliser. Mais d'où vient cette prise de décision ? Selon Tony Robbins les décisions sont régies par les émotions que nous ressentons.

"[...] I believe that emotions are the force of life"

"[...] Je crois que les émotions sont la force de la vie"

*Tony Robbins
Pourquoi nous faisons ce que nous faisons
Ted.Com*

Dans sa conférence, Tony Robbins pose la question de ce qui détermine la capacité d'une personne à agir, à contribuer et à se dépasser. L'humain est déterminé par deux forces invisibles :

- La première le détermine sur le moment : il s'agit de son état physique et émotionnel.
- La seconde le détermine sur le long terme et correspond au modèle du monde qu'il se représente. Cette représentation de la réalité lui fait prendre des décisions. Graham Little dans son livre "The origin of consciousness" parle de notre représentation de la réalité comme étant différente de la réalité elle-même. D'ailleurs il appelle réalité avec un "r" minuscule, l'image que l'on se représente du monde et Réalité avec un "R" majuscule pour représenter la Réalité même du monde qui nous entoure. Ainsi notre image de la réalité influe ce que nous voyons ce qui détermine ce que l'on fait. Les répercussions

de ce que nous faisons modifie notre réalité et de ce fait, ce que l'on voit puis nos actes, etc.

Lorsqu'une personne réagit face à une situation, que ce soit en bien ou en mal, ce sont ses émotions qui régissent l'acte. Et c'est cette prise de décision qui m'intéresse lorsqu'un joueur choisit de répondre d'une certaine manière à un stimulus du jeu. Ainsi, selon Tony Robbins, l'acte décisionnel se divise en trois "étapes" :

- D'abord, la personne choisit consciemment ou non sur quoi elle se focalise. Est-ce-que cela concerne le passé ? le présent ? le futur ? Elle-même ou les autres ?
- Dès que l'objectif est déterminé, la personne donne un sens à cet objectif ce qui produit des émotions positives ou négatives. Est-ce-que c'est le début ou la fin de quelque chose ? Une punition ou une récompense ?
- Enfin, en fonction de ces émotions, l'action est déterminée. Ces émotions répondent aux besoins psychologiques de la personne. Qu'est-ce-que je fais : j'abandonne ou j'avance ?

Prendre une décision et agir répond à des besoins. Dans sa conférence il détermine les six besoins psychologiques de l'être humain. Quatre concernent les besoins de la personnalité et les deux restant, les besoins de l'esprit. Les besoins de la personnalité sont composés :

- du besoin de certitude afin de se sentir à l'aise. On l'obtient en contrôlant, développant des aptitudes, etc. Cependant, lorsqu'il y a trop de certitudes dans la vie d'une personne cela engendre l'ennui.
- du besoin d'incertitude : l'être humain a besoin de variété, de surprises, de problèmes.
- du besoin d'importance : de se sentir unique et spécial.
- du besoin de relation et d'amour : l'intimité, l'amitié, la religion, etc.

Les besoins de l'esprit comprennent :

- le besoin de grandir : évoluer pour transmettre quelque chose de valeur.
- le besoin de contribuer au-delà de soi-même : participer au développement de la culture collective, être intéressé par les autres.

À des degrés différents en fonction des gens, ces six besoins nous définissent.

On peut tout à fait illustrer ces notions au travers des jeux. Prenons une situation type de jeu dans laquelle le joueur voit un village attaqué par un ennemi très puissant. Un survivant du village l'implore de les aider. Le joueur est placé dans la position de sauveur et de protecteur (remplissant le besoin du joueur de se sentir important : c'est l'être ultime capable d'affronter un ennemi aussi puissant et de sauver le reste des villageois). L'objectif qui se dessine naturellement au joueur c'est d'aller affronter cet ennemi pour arrêter son attaque, le punir et l'empêcher de nuire de nouveau. Le sens donné à cet acte de violence du joueur envers l'ennemi, est que s'il n'intervient pas, c'est lui qui sera déterminé comme un monstre. Il aura échoué et ne recevra aucune récompense, voir même il pourra être puni. Les émotions suscitées dans cette situation peuvent être de l'empathie pour le village détruit et les gens tués, ce qui engendre de la colère vis-à-vis de l'ennemi, mais également peut être un sentiment de satisfaction qui va découler de la démonstration de puissance que le joueur va démontrer face à l'ennemi. Enfin l'action est déterminée : vaincre l'ennemi. La certitude qu'un joueur peut avoir dans ce genre de

situation, c'est qu'il arrivera à vaincre cet ennemi même si cela lui sera difficile et qu'il doit recommencer plusieurs fois. Cependant, l'incertitude qu'il peut y avoir concerne le comportement de l'ennemi. Quelles attaques va-t-il faire ? Dans quel ordre ? De quelle puissance ? De quel type : à distance ou au corps à corps, attaques physiques ou magiques ? Ce comportement de l'ennemi que le joueur essaie de définir et comprendre est appelé pattern. C'est le schéma d'actions et d'enchaînement d'actions auquel les ennemis répondent. Chaque type d'ennemi a son pattern. Il est plus ou moins facile à identifier. Par exemple, dans Darksiders 2 lorsque le joueur affronte le personnage nommé War, le schéma d'action de War est :

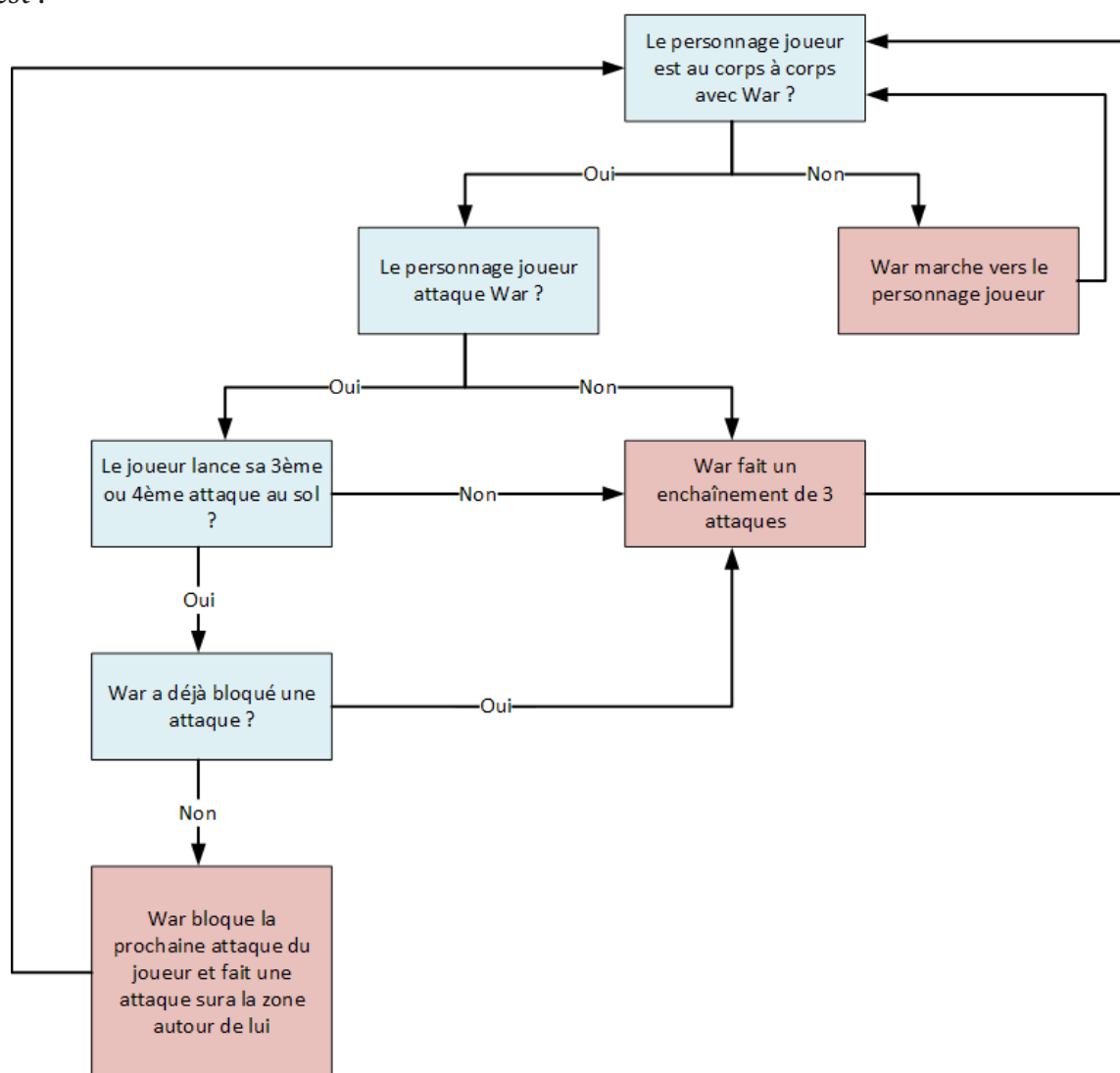


Figure 1 : Boucle de comportement du personnage War

Dans un jeu vidéo, lorsqu'un objectif à accomplir est donné au joueur, il faut que cet objectif ait un sens. La plupart du temps il est intégré narrativement dans l'histoire et en accomplissant cet objectif le joueur va ressentir des émotions répondant à ses besoins. Aussi, lorsque le game designer décide de l'objectif il doit déterminer quels besoins du joueur et quelles émotions vont être suscitées dans le jeu pour motiver l'action du joueur. Si la quête, l'objectif à accomplir ne répond à aucun besoin du joueur alors celui-ci ne s'impliquera pas à le résoudre. Ou s'il y est obligé, n'en tirera aucun plaisir.

Pour autant, on peut voir un paradoxe entre les émotions comme facteur motivationnel du joueur et la motivation engendrée par les récompenses. En effet lorsqu'un objectif est rempli par le joueur il obtient généralement une récompense : cela peut être de poursuivre son aventure dans le jeu, une récompense financière (le joueur gagne de l'argent virtuel lui permettant d'acheter des objets consommables du jeu), ou encore de l'expérience qui une fois accumulée peut permettre au personnage d'évoluer, etc. Dans certains cas la récompense peut être un facteur motivationnel qui va prendre l'avantage sur le besoin. Prenons pour exemple les jeux de rôle, qui sont des univers virtuels dans lesquels le joueur fait évoluer son ou ses personnages selon un certain nombre de niveau, gagnant des compétences et de la puissance au fur et à mesure de la progression des niveaux du personnage.

Le premier moyen donné au joueur pour faire évoluer son personnage dans ce genre de jeu est de réaliser des quêtes qui, en général, se caractérise par des objectifs comme :

- Aller chasser x monstres
- Ramasser x châtaignes
- Parler à des personnages non joueurs (PNJ)
- etc

Ces objectifs sont souvent récompensés par de l'expérience que gagne le personnage pour monter en niveau, de l'argent virtuel et parfois une nouvelle arme ou une nouvelle armure donnée au personnage du joueur. Le besoin premier mis en avant est le besoin de faire progresser son personnage, engendrant de la satisfaction lorsque celui-ci réussit sa quête et gagne de l'expérience. Cependant, la répétition d'accomplir des quêtes et la redondance de leurs objectifs (certitude) peut entraîner de l'ennui de la part du joueur qui peut se désintéresser des quêtes et chercher un autre moyen de faire évoluer son personnage si le jeu en présente un. Dans le cas où le jeu ne présente pas d'autre moyen d'évoluer, le joueur ne sera non plus motivé par l'émotion de satisfaction en résolvant sa quête, mais simplement par le gain de récompense, ce qui n'inhibe en aucun cas l'émotion négative qu'est l'ennui. De mon expérience de joueuse sur ce genre de jeu : un contenu répétitif n'engendrant plus de motivation par le simple fait d'être satisfait de l'action que l'on réalise, forcera tôt ou tard le joueur à délaisser complètement le jeu. Le temps passé à jouer dépendra simplement de la patience du joueur à endurer.

C'est ce que Edward L. Deci et Richard Flaste évoquent dans leur livre *"Why we do what we do : understanding self-motivation"* lorsqu'ils parlent de motivation intrinsèque. Qu'est ce que la motivation intrinsèque ?

"[...] the concept of intrinsic motivation, which refers to the process of doing an activity for its own sake, of doing an activity for the reward that is inherent in the activity itself."

"[...] le concept de motivation intrinsèque, se réfère au processus de faire une activité de son propre chef, de faire une activité pour la récompense qui est inhérente à l'activité elle même."

*Edward L. Deci et Richard Flaste
Why we do what we do : understanding self-motivation*

L'être humain est intrinsèquement motivé lorsqu'il est à l'origine de sa propre action. C'est-à-dire de faire une activité pour le simple plaisir de la réaliser. Comme les enfants sont motivés à découvrir

un nouvel objet ou une nouvelle notion : pour réussir le défi de la découverte et apprendre. Ce qui est leurs principales motivations. Cette notion de motivation intrinsèque disparaît ou s'amenuise en grandissant, car les activités exercées par l'être humain ont pour la plupart une récompense en facteur motivationnel : l'école, le travail, etc.

Aux travers de leurs expérimentations, les auteurs posent la question de l'impact des récompenses sur la motivation intrinsèque des gens. Est-ce que les récompenses inhibent la motivation intrinsèque à réaliser une activité ? Si oui, est ce toujours le cas ? Quelles sont les conséquences ? Comment délivrer des récompenses sans inhiber la motivation intrinsèque ?

“Reward the desired behavior, and there is increased likelihood that the behavior will be repeated”
“Récompensez le comportement désiré, et il y a une probabilité que le comportement se répète.”

Edward L. Deci et Richard Flaste
Why we do what we do : understanding self-motivation

Le parfait exemple du maître qui récompense son chien après s'être assis sur ordre, ou à l'enfant qui a le droit de regarder les dessins animés après avoir fait ses devoirs. Ce sont les principes du behaviorisme et du concept de renforcement en psychologie. A savoir que le comportement désiré est récompensé, on parle de renforcement positif, alors qu'un mauvais comportement engendre un renforcement négatif via une punition. Le problème de ce genre de procédé c'est qu'il fonctionne uniquement s'il y a récompense. Que se passe-t-il si l'activité n'est plus récompensée ? Comme est cité Barry Schwartz dans l'ouvrage :

“People are fundamentally passive and will respond only when the environment tempts them with the opportunity to get rewards or avoid punishment”
“Les gens sont fondamentalement passifs et répondront seulement lorsque l'environnement leur permettra d'obtenir des récompenses ou d'éviter une punition”

Citation de Barry Schwartz
Why we do what we do : understanding self-motivation

Donc ce qu'il faut trouver, c'est comment encourager l'autonomie des gens à poursuivre une activité même si elle n'est pas récompensée.

“being intrinsically motivated has to do with being wholly involved in the activity itself and not with reaching a goal”.
“être intrinsèquement motivé à avoir avec le fait d'être entièrement impliqué dans l'activité elle-même et non avec l'accomplissement d'un objectif”

Edward L. Deci et Richard Flaste
Why we do what we do : understanding self-motivation

Si je reprends les quêtes que le joueur réalise dans les jeux de rôles et que je les transpose sur cette problématique. Les quêtes sont ambiguës car elles possèdent deux façons de motiver le joueur :

- la première par la satisfaction des besoins du joueur : il vit une aventure le mettant dans la peau, d'un guerrier, d'un mage ou de tout autre personnage envoyé remplir l'objectif d'une quête que seul lui peut accomplir. En plus de cela son personnage va acquérir de l'expérience en tuant les ennemis qu'il va rencontrer au cours de la quête. Sentiment d'importance ainsi que besoin de grandir satisfait. Certitude de réussir la quête (car les quêtes sont disponibles pour le joueur en fonction du niveau de son personnage, donc le joueur réussira obligatoirement sa quête moyennant quelques difficultés), mais incertitude quant au niveau de difficulté et ennemis rencontrés (avec tout le questionnement vis-à-vis du comportement ennemi).

- la seconde par les récompenses qu'offre le jeu : en plus de l'expérience obtenue durant sa mission, de l'expérience supplémentaire est donnée à la réussite de son objectif. Et d'autres récompenses sont données en supplément. Le besoin de grandir est donc en partie satisfait grâce au gain d'expérience.

Tant que les besoins sont satisfaits, il y a motivation intrinsèque à réaliser les quêtes. Le problème que l'on a vu précédemment, est dans la répétition des objectifs. Certes le joueur part à l'aventure, mais on a vu qu'une trop longue répétition des actions accomplies pour réaliser la quête peut devenir ennuyeux s'il n'y a pas d'autres moyens de satisfaire les besoins d'évolution du joueur. Dans ce cas la motivation intrinsèque à faire l'activité s'amenuise et l'intérêt du joueur à poursuivre l'accomplissement des quêtes ne reste présent que pour l'obtention des récompenses. C'est ce que les auteurs appellent l'aliénation : la perte de motivation intrinsèque à faire une activité.

Je tiens à préciser qu'à aucun moment du livre les auteurs défendent l'idée de supprimer les récompenses, mais soutiennent la possibilité de les délivrer autrement pour conserver la motivation intrinsèque d'une personne. Dans ce cas nous devons comprendre comment l'être humain perçoit les récompenses. Inconsciemment les récompenses sont perçues comme un objet de pression de manipulation du comportement : on reçoit une récompense pour avoir fait un travail ou pour avoir réalisé une activité en général. Plus globalement tout ce qui exerce une pression sur la volonté de l'être humain à réaliser une activité diminue la motivation intrinsèque de l'individu. Autre que les récompenses il peut également s'agir de deadline ou encore de compétition, etc. Cette pression donne l'impression à l'être humain de ne pas être complètement libre de faire l'activité qu'on lui demande et donc diminue la motivation intrinsèque à réaliser cette activité.

"[...] maybe people have needs of minds as well as need of body"

"[...] peut être que les gens ont autant de besoin de l'esprit que de besoin du corps"

Citation d'Henry Murray

Why we do what we do : understanding self-motivation

C'est-à-dire que nous avons un besoin inné ou intrinsèque de ressentir une autonomie ou une détermination personnelle dans la prise de décision à faire une action. Dans un jeu de plateforme / action où le comportement du joueur est déterminé à l'avance par les game designers, celui-ci agit comme ce

qui est prévu qu'il fasse. Mais que se passe-t-il si le jeu ne correspond pas à ce que le joueur aimerait faire ? Si la situation dans le jeu demande au joueur de tuer tous les ennemis qui lui bloquent le passage alors que celui-ci préférerait s'infiltrer et contourner les ennemis pour retirer toute la satisfaction de l'expérience. Peut être que le joueur va contraindre son comportement à ce qu'il lui est demandé dans le jeu. Peut être qu'il le fera plusieurs fois et peut être même s'il en a le courage jusqu'à la fin du jeu (si la récompense à la fin du jeu lui semble suffisante pour justifier sa poursuite de l'aventure). Mais que va-t-il ressortir de l'expérience qu'il a vécue ? De la déception, de la frustration... Le jeu n'aura pas répondu à ses besoins et ne l'aura donc pas satisfait. Je tiens à préciser cependant qu'un jeu ne répondant pas au besoin d'un joueur ne signifie pas que ce jeu est mauvais, mais simplement que l'expérience que le joueur retient du jeu n'est pas optimale. Ce jeu pourrait très bien convenir à un autre joueur. Cependant, dans une industrie où l'objectif est tout de même de fournir des expériences vidéoludiques à un maximum de personnes, il serait dommage de restreindre l'impact du jeu simplement parce qu'il ne convient pas à tous les joueurs adeptes du type de jeu en question. En soit les actions du joueur peuvent très bien être un facteur de motivation intrinsèque. Et c'est la raison pour laquelle j'aborde dans ma problématique la notion de génération d'actions en fonction du comportement du joueur.

La récompense d'un joueur à générer un gameplay au travers de ses actions, est d'avoir un jeu qui s'adapte entièrement à sa manière de jouer et à ses besoins. Dans ce cas comment générer de la motivation intrinsèque au joueur au travers des actions qu'ils exécutent et sans que ces actions ne soient perçues comme dirigées par les concepteurs du jeu ?

"People who were ask to do a particular task but allowed the freedom of having some say in how to do it were more fully engaged by the activity, they enjoyed it more than people who were not treated as unique individual"

"Les gens à qui on a demandé de faire une tâche mais à qui on a laissé la liberté de choisir comment la réaliser étaient complètement engagés dans l'activité, ils l'ont plus appréciées que les personnes qui n'étaient pas traités comme un individu unique".

*Edward L. Deci et Richard Flaste
Why we do what we do : understanding self-motivation*

Une manière de demander une activité à un individu sans trop impacter sa motivation intrinsèque est de lui laisser le choix quant la manière de résoudre cette activité.

Dans la plupart des jeux, le but des concepteurs est de faire vivre une expérience, une aventure aux joueurs. Ils doivent donc forcément orienter les actions de celui-ci. Pour autant donner le choix au joueur de choisir quelle action il va réaliser est un point essentiel à sa motivation, et de plus en plus de jeux intègrent des choix à faire par les joueurs comme élément de gameplay.

Par exemple le jeu "Life is strange" du studio Dontnod, est un jeu épisodique narratif dont l'intégralité du jeu repose sur les choix que le joueur fait. Dans chaque situation le joueur a le choix d'agir de différentes manières. Chaque décision d'action aura un impact sur la vie du joueur, la vie des personnages de l'histoire et même sur l'environnement du jeu. De plus, le personnage que le joueur contrôle a la capacité de remonter dans le temps, de "rembobiner" l'action qu'il vient juste de réaliser. Donc si les

répercussions du premier choix d'action ne lui conviennent pas il peut rejouer la situation autant de fois qu'il le désire pour trouver la décision qui lui convient le plus. Le joueur "choisit le choix".

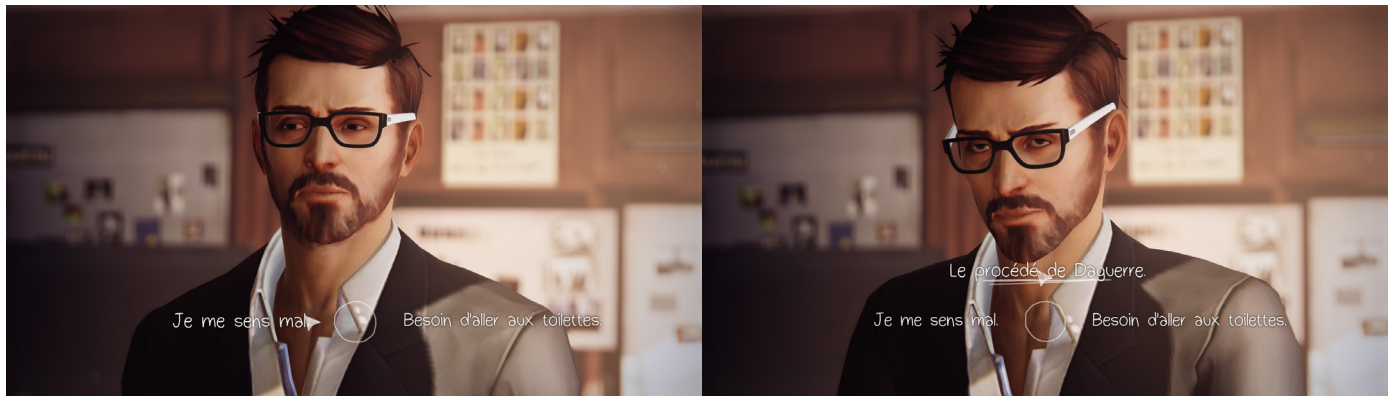


Figure 2 : Après avoir vécu la première situation le joueur remonte dans le temps pour revivre la même scène, mais cette fois un autre choix s'offre à lui

Autre exemple de jeu : Gears of war, un jeu de tir à la troisième personne (ou TPS) dans lequel le joueur affronte des monstres (les locustes). C'est un jeu d'action axé sur le combat via armes à feu. A certain instant du jeu, le joueur peut choisir la tactique d'approche des ennemis. Pour illustrer : le joueur doit s'infiltrer dans les cavernes ennemies. Il y a deux passages qui peuvent être empruntés. Le joueur choisit comment départager son commando. Il choisit par quel chemin lui et son coéquipier vont passer, les deux autres personnages non joueur prendront l'autre chemin. Cela permet de varier les phases de jeu. Lorsque le joueur fait un choix et que les deux équipes prennent des chemins différents, elles doivent évoluer en même temps, une équipe débloquent le chemin de l'autre et inversement.

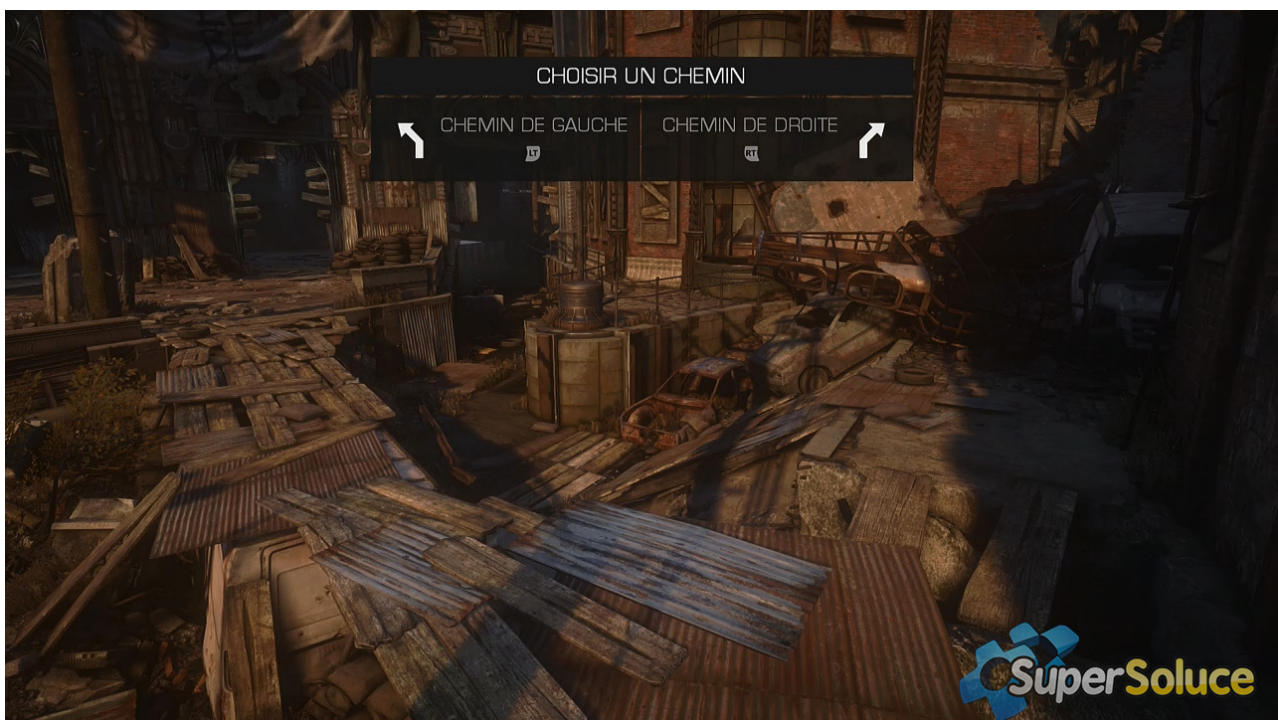


Figure 3 : Le joueur doit choisir quel chemin il va emprunter

Lorsque je souhaite que le gameplay se génère en fonction du joueur, cela implique qu'il se génère en fonction des choix d'action du joueur, afin de promouvoir son autonomie dans les jeux vidéo. Ces choix étant implicitement décidés par ses besoins et ses émotions dans une situation donnée. Le gameplay doit pouvoir se générer et s'adapter en fonction du profil comportemental du joueur. Voir le monde sous le même regard que les joueurs pour leur prodiguer une expérience qui convient le mieux à leurs besoins est appelé par les auteurs de *why we do what we do* : "autonomy supportive", être un support à l'autonomie.

"Quand on veut influencer quelqu'un, on doit savoir ce qui l'influence déjà"

Tony Robbins
Pourquoi nous faisons ce que nous faisons
Ted.Com

C'est à dire déterminer l'objectif engendré par les besoins d'une cible identifiée, déterminer comment satisfaire ses besoins, et enfin déterminer par quel moyen, quel médium ils peuvent être satisfaits.

I.1.2) Les jeux vidéo : un monde parallèle

Les jeux vidéos offrent la possibilité aux joueurs d'assouvir inconsciemment leurs besoins psychologiques aux travers de ces univers virtuels. En comprenant ces besoins, les concepteurs de jeu mettent en place des mécanismes et des environnements virtuels attisant la motivation du joueur pour lui faire poursuivre le jeu. Mais que représentent ces mondes vidéoludiques pour les joueurs ? Quelles sont les différences et les similitudes avec le monde réel ? Comment le joueur se représente-il dans ces mondes virtuels ? Pour comprendre la motivation générale des joueurs à jouer à des jeux vidéo nous devons comprendre comment ils sont constitués.

Jouer n'est pas une invention de l'être humain. Revenant au principe primaire de l'activité "jouer", Chris Crawford défend l'idée selon laquelle la motivation première des jeux est d'apprendre.

"Games are thus the most ancient and time-honored vehicle for education. They are the original educational technology, the natural one, having received the seal of approval of natural selection."

"Les jeux sont donc le plus ancien et authentique moyen d'éducation. Ils sont la technologie d'éducation originelle, la plus naturelle, ayant reçu le sceau d'approbation de la sélection naturelle."

*Chris Crawford
Art computer game design*

Présent dans notre environnement naturel, le jeu peut être observé chez les animaux comme un acte d'apprentissage de l'art de survie. Chris Crawford fait un parallèle avec les lionceaux qui chassent un papillon non pas pour s'amuser, mais parce que jouer leur apprend l'art de la chasse. Chez l'être humain le jeu comme moyen d'éducation a été remplacé par l'école et la notion de jeu et de jouer est attribuée aux enfants comme un médium secondaire reconnu à leur développement personnel et intellectuel.

Au-delà de l'apprentissage comme facteur premier motivant le jeu, l'auteur dénombre six autres facteurs secondaires associés aux jeux vidéo. Je vais regrouper ces six facteurs en cinq (car deux d'entre eux abordent la même notion).

Le premier facteur concerne l'environnement physique de l'univers virtuel, ce qui en est visible. Ce facteur est appelé Fantaisie et exploration. C'est le principe de transporter le joueur dans un univers complètement différent du sien dans lequel il va partir à l'aventure, explorer une multitude de lieux étranges, effrayants, ou fantastiques.

"Many art forms attempt to transport the audience into a different world, to present experiences or feelings not often known in the everyday world."

"De nombreuses formes d'art tentent de transporter l'audience dans un monde différent, pour présenter des expériences ou des émotions peu connues dans le monde de tous les jours"

*Chris Crawford
Art computer game design*

Partir à l'aventure et vivre une expérience dans laquelle le joueur peut interagir avec son environnement (dans la limite de ce qui a été programmé), ressentir des émotions qu'il n'a pas ou peu l'habitude de ressentir dans sa vie réelle. Les jeux vidéo ont cet avantage que le joueur n'est pas passif, il peut agir et "vivre" dans cet univers contrairement aux autres médiums de divertissement tel que le cinéma ou les livres, qui eux aussi transportent leur audience dans un autre univers, mais dans lequel cette audience est passive et n'est présente qu'en terme de spectateur. Si le cinéma 3D ou les simulateurs ont été inventés c'est probablement pour réduire cette frontière entre le passif et l'actif : plonger le spectateur au coeur de l'action, même si celui-ci est assis sur un siège.

Le second facteur de motivation concerne les restrictions sociales présentes dans la vie de chaque individu. L'auteur l'appelle : "nose-thumbing" ou pied-de-nez en traduction française littérale. C'est l'idée que le jeu vidéo place le joueur dans un rôle qu'il ne pourrait endosser dans la vie réelle car ce rôle ne serait pas accepté : jouer un personnage psychopathe, drogué et mégalomane qui s'associe à deux autres criminels pour réaliser des braquages (Grand theft auto V) n'est pas vraiment le modèle de vertu répandu dans notre monde social actuel. Même si cet exemple est un cas extrême ne pouvant être représentatif de tous les jeux vidéo. Si je reprends pour exemple le village attaqué par un ennemi puissant que le joueur doit défendre. L'acte de violence du joueur envers l'ennemi, aussi violent soit-il, est justifié par les actes d'atrocité commis par l'ennemi afin de décharger le joueur de toute culpabilité.

Le troisième facteur correspond au besoin de reconnaissance du joueur ce qui correspond au besoin d'importance de Tony Robbins. Le joueur aime se sentir important et au-delà de cela, le joueur aime montrer ses prouesses et son importance aux autres joueurs. Cela peut être par un système de score et de classement qui classe les joueurs : de celui qui a obtenu le plus de points, ou le plus haut rang à celui en ayant le moins.

"If the game in any way allows individual prowess to affect the outcome, then the outcome is perceivable as a reflection of individual prowess"

"Si le jeu, de quelques manières, permet à la prouesse individuelle d'affecter le résultat, alors le résultat est percevable comme le reflet de la prouesse individuelle".

*Chris Crawford
Art computer game design*

Il va s'en dire que plus le rang du joueur est élevé et plus celui-ci va éprouver de satisfaction et de fierté. Parfois ce sentiment peut engendrer un comportement prétentieux du joueur. Bien sûr cela dépend de la personnalité du joueur en question. Être reconnu par les autres joueurs peut se matérialiser par les prouesses du joueur et son rang mais également par tout élément montrant la richesse du personnage joué. Comme par exemple les "achievements" qui sont des objectifs que les joueurs peuvent remplir pour obtenir des points, des titres, des apparences d'armures, etc. Les joueurs sont également très fiers de montrer leur armure ou les "skin" qu'ils ont placés dessus (le skin correspond à une apparence de vêtement). Les montures et les mascottes (petit animaux qui courent à côté du joueur lorsqu'il se déplace) sont également des éléments que les joueurs sont fiers d'afficher. Ce désir d'afficher ce que le joueur possède correspond exactement à la fierté qu'une personne a de montrer sa rolex : parce que

cela montre le niveau du joueur et le taux d'éléments débloqués qui correspondent à des dizaines, parfois des centaines d'heures de jeu.

Le quatrième facteur correspond aux liens sociaux. Dans les jeux vidéo et plus particulièrement ceux qui permettent d'être en contact avec d'autres joueurs ; des liens se forment entre les individus : parfois positifs conduisant à l'amitié ou l'amour (certaines relations amoureuses se construisent via les jeux), parfois négatifs. Les barrières sociales entre individus deviennent très fines ce qui permet de former très facilement de nouvelles relations ou d'engendrer des comportements sociaux qui ne pourraient exister dans le monde réel. Ainsi des joueurs ayant un même but commun, peuvent se regrouper sous la même bannière en formant une guildes pour jouer et évoluer ensemble. En général les guildes demandent beaucoup d'investissement personnel, que ce soit dans le jeu pour aider ses co-équipier ou dans la vie réelle pour créer un forum à la guildes, une page facebook, un logo, etc. Cela donne une image à la guildes et permet aux joueurs d'afficher avec fierté ce qu'ils ont accomplis socialement dans le jeu (retour sur le facteur du besoin de reconnaissance, de relation et d'évolution). Sans oublier que les personnes faisant partie d'une même guildes sont très souvent en contact audio via des logiciels tels que mumble, teamspeak ou discord. En dehors de guildes, les jeux en ligne permettent aux joueurs de jouer en communauté ne serait-ce que par les donjons qui requièrent d'être appréhendés à plusieurs. Les donjons sont des environnements "extérieurs", "parallèles", à la zone de jeu que les joueurs pratiquent habituellement et requièrent des conditions pour y accéder : être en groupe ou non, avoir débloqué son contenu via une quête, niveau du personnage ou d'équipement, etc. C'est un lieu fermé, défini qui ne change pas et que les joueurs peuvent explorer en affrontant des monstres et des boss. En général vaincre les boss d'un donjon rapporte de belles récompenses.



Figure 4 : Dojon du jeu Final fantasy XIV online dans lequel les joueurs affrontent des monstres. Les donjons sont des environnements isolés du monde global du jeu comme le montre cette carte.

Enfin les joueurs de mondes virtuels aiment participer à des événements inattendus simplement pour le plaisir de participer à un acte social. Je vais prendre pour exemple une situation que j'ai vécue récemment sur le jeu Final Fantasy XIV online. Avec trois autres membres de ma guilde nous avons commencé à jouer au petit train, chacun étant à dos de cheval nous paradions dans une ville très fréquentée. Nous n'avions aucunement annoncé notre jeu aux joueurs qui nous entouraient. Quelques minutes après le début de notre parade plus d'une trentaine de joueurs nous suivaient.

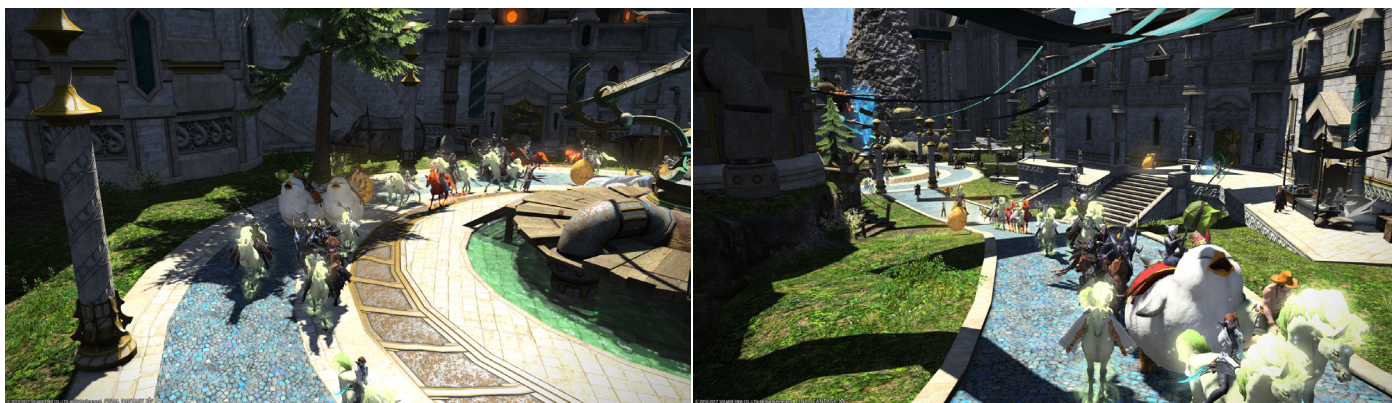


Figure 5 : Parade formée par les joueurs du jeu Final Fantasy XIV online

Cela dénote typiquement du comportement des joueurs à rejoindre un événement, quel qu'il soit, même si cela ne le concerne pas au premier abords. Pourquoi ? Parce que c'est un geste simple qui ne demande aucune implication personnelle avec les initiateurs de cet événement, mais simplement de réaliser un acte collectif à s'amuser en essayant de faire la plus longue chaîne possible (motivation intrinsèque de l'activité). Et si nous transposions ceci dans le monde réel : essayez avec trois de vos amis de démarrer une chenille en pleine place publique simplement pour vous amuser et sans inviter les gens qui passent à vous rejoindre. Comment pensez-vous que les gens vont réagir autour de vous ? Vont-ils vous rejoindre de leur propre chef, simplement pour la simple motivation de s'amuser ? Ou plutôt de s'afficher en public ? Car c'est plutôt cette opinion négative qui primera par le regard de ceux qui vous jugeront. Sur un jeu, il n'y a pas cette peur du regard des autres, personne ne vous "dévisagerait" et de ce fait, vous vous sentez bien plus libre de participer à un acte social (pas de timidité).

Enfin le dernier facteur n'est ni plus ni moins que l'exercice physique ou mental. Que ce soit dans des jeux d'énigmes, de tir, de rôle, etc, le joueur est amené à anticiper, à mémoriser, à réagir cognitivement à un stimulus du jeu. Et plus un stimulus est utilisé pour engendrer une réponse et plus la réponse à ce stimulus sera rapide au fur et à mesure de son utilisation, ce qui fait progresser et évoluer le joueur. Pour stimuler ces réflexes physiques et mentaux les jeux sont conçus selon une courbe de difficulté croissante afin d'apporter du challenge au joueur sans pour autant le placer dans une situation insurmontable.

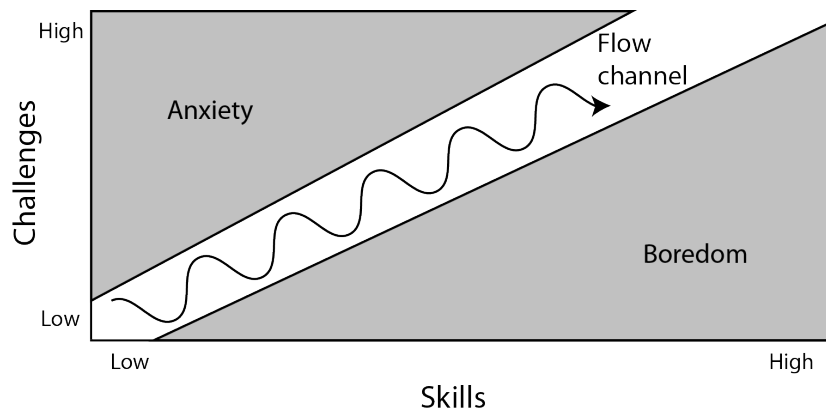


Figure 6 : Pour que l'expérience d'un jeu soit optimale la progression de son niveau de difficulté doit se trouver dans le «Flow channel»

Au travers de ces cinq facteurs motivationnels nous pouvons aisément faire des liens avec les besoins humains cités par Tony Robbins :

- le besoin d'importance est clairement comblé par le facteur du besoin de reconnaissance. Les joueurs montrent leurs prouesses et le développement de leur personnage. Ils en sont fiers et se sentent admiré par d'autres joueurs. Il peut également se retrouver dans le facteur social, notamment dans les guildes où même si les joueurs s'adressent entre eux sans distinction de rang, les guildes possèdent une hiérarchie : le chef de guildes puis son ou ses seconds, etc. Également avec le facteur de restriction sociale car le joueur est placé dans la peau du héros, souvent très puissant ou en tout cas bien plus fort qu'un individu normal.

- Le besoin de relation est comblé de différentes manières. Tout d'abord, par le lien que le joueur entretient avec son environnement (facteur fantaisie et exploration). Cet univers a tout de même été construit pour lui et ses besoins. Les personnages non joueurs l'accompagnent dans son aventure et vont également créer des liens avec lui et être ses partenaires, ils vont le soutenir, l'aider, etc. Enfin, par les relations sociales humaines qu'il va pouvoir construire, notamment au travers des jeux en ligne.

- Les besoins de certitude et d'incertitude se retrouvent dans presque chacun des facteurs exprimés auparavant. La certitude concernera plutôt l'état du joueur au travers de son personnage et l'incertitude tous les paramètres du jeu que le joueur ne peut contrôler, comme les ennemis, les probabilités de gagner ou non un objet, etc.

- Les besoins de l'esprit à savoir, l'évolution et la contribution du joueur, concernent plutôt ce qu'il va réaliser au travers de son personnage. Tout au long d'un jeu le joueur évolue, que ce soit de manière virtuelle en faisant gagner de l'expérience à son personnage ou de par ses aptitudes cognitives et mentales qui vont se développer. De plus, ses actions vont laisser une trace dans le monde virtuelle. Cela est montré au travers de la narration du jeu en lui montrant les conséquences de ses actes. Les répercussions peuvent être plus ou moins grandes en fonction de l'action effectuée. Donc le joueur contribue à l'évolution du jeu. Même si cela est programmé, l'histoire du jeu ne peut poursuivre sans le joueur.

Les jeux vidéo reprennent les codes de notre monde et les transforment, les adaptent pour créer des univers excluant les contraintes liées au monde réel. Chaque facteur motivationnel que nous avons vu jusqu'ici existe également dans notre monde. Ils s'expriment différemment et sont plus restrictifs,

car dans notre réalité nous sommes soumis aux contraintes physiques qui nous sont propres ou sociales et morales qui régissent notre société. Cette minimisation des contraintes donnent aux joueurs la possibilité de pouvoir réaliser des aventures, des expériences et de ressentir des émotions qu'ils ne pourraient pas ou peu ressentir dans la réalité. Ce dédoublement de leur personnalité se fait au travers de leur personnage virtuel qui va traduire virtuellement les actes du joueur découlant de ses propres émotions et besoins psychologiques.

Cette implication psychologique des joueurs dans des mondes virtuels est telle, que des designers cherchent à dupliquer cette implication pour la retranscrire dans le monde réel et motiver les actions des gens à réaliser une certaine activité dans un domaine précis. C'est ce que l'on appelle la gamification. C'est-à-dire l'utilisation des mécaniques et des codes propres aux jeux vidéo dans des applications n'étant pas des jeux afin de contrôler le comportement de la cible. Cela s'utilise notamment sur les sites internet pour garder la fidélité du client et l'inciter à revenir régulièrement.

Dans sa conférence *"The game layer on top of the world : how game dynamics are reshaping the world"*, Seth Priebatsch expose quatre mécaniques de jeu transposées dans la réalité en les illustrant avec des exemples :

- La première qu'il nomme "appointment dynamique" qui correspond à cette faculté des jeux à faire revenir les joueurs toutes les x heures tels que les jeux sociaux comme Farmville. Lorsque les plantations d'un joueur on finit de pousser, il reçoit une notification pour revenir jouer. Seth Priebatsch fait le lien avec les happy hour qui motivent les gens à aller boir un verre dans les bars chaque jour. C'est un rendez-vous social.

- La seconde qu'il nomme "influence and status" qui poussent les gens à modifier leur comportement pour atteindre une récompense. Par exemple, lorsqu'un joueur voit un super badge qu'il ne peut avoir qu'en ayant atteint le niveau dix. Il va modifier son comportement pour arriver le plus rapidement à son objectif.

- La troisième appelée "progression dynamique" utilisée par les réseaux sociaux ou les sites web pour faire passer le visiteur au travers de différentes étapes. Comme les niveaux que le joueur va gravir dans un jeu de rôle, une personne va compléter son statut de réseau social. Ce statut souvent signalé par une barre de progression montre à quel pourcentage la personne a rempli ses informations.

- La dernière, "communal discovery" correspond à la capacité des gens à s'associer pour atteindre un objectif.

Pour aller plus loin dans l'implication des joueurs à réaliser des actions incroyables dans les jeux vidéo, Jane McGonigal parle de retranscrire ce comportement dans la réalité pour sauver le monde. Dans sa conférence *"Gaming can make a better world"*, elle aborde les différences entre les mondes virtuels et la réalité, l'implication des joueurs et comment motiver cette implication dans le monde réel. Lorsqu'elle parle des joueurs, et plus particulièrement des émotions qui les impliquent dans un jeu, elle aborde le terme *"d'epic win"*.

"An epic win is an outcome that is so extraordinary positive that you had no idea that it was even possible until you achieved it. It was almost beyond the threshold of imagination, and when you get there, you're shocked to discover what you're truly capable of"

“Une victoire épique est un aboutissement qui est si extraordinairement positif que vous n’aviez pas idée que c’était même possible jusqu’à ce que vous l’ayez atteint. C’était presque au delà du seuil de l’imagination, et lorsque vous y accédez, vous êtes choqués de découvrir de quoi vous êtes réellement capable.”

Jane McGonigal
“Gaming can make a better world”

Cette notion correspond aux challenges dans lesquels peuvent se lancer les joueurs sans même savoir s’ils vont le réussir jusqu’à ce qu’ils y arrivent vraiment. Et lorsque l’objectif est atteint, avec tout l’investissement virtuel et temporel auquel les joueurs font appel, ils atteignent cette émotion de pleine satisfaction, de pleine jouissance. Pour Jane McGonigal c’est ce genre d’émotion qu’il faut pouvoir susciter dans la vie réelle des gens pour leur permettre de sauver le monde. Mais parce qu’il est plus facile d’atteindre un objectif et de sentir qu’on contribue à quelque chose qui nous dépasse dans un jeu vidéo, les joueurs se sentent meilleurs dans les jeux que dans la vie réelle. Ils sont bien plus prêts à coopérer dans les mondes virtuels que dans la réalité.

“[...] I believe that many of us become the best version of ourselves, the most likely to help at a moment’s notice, the most likely to stick with a problem as long as it takes, to get up after a failure and try again”
“Je crois que beaucoup d’entre nous deviennent la meilleure version d’eux-même, la plus susceptible d’aider à un moment donnée, la plus susceptible de résoudre un problème aussi longtemps que cela prendra, de se relever après un échec et de recommencer”

Jane McGonigal
“Gaming can make a better world”

Ce qui freine la motivation des gens à résoudre une solution dans la réalité c’est la peur de l’échec et l’investissement personnel qui est bien plus grand dans la réalité. Si le joueur échoue dans un jeu, ce n’est pas grave, il peut recommencer. Peut-être qu’il ressentira de la déception ou de la frustration, mais elle sera contrebalancée par l’envie de réussir et de se dépasser. Alors qu’une personne qui échoue dans la réalité va se sentir submergée, triste, anxieuse... Ces émotions ne peuvent être ressenties dans un jeu. En jouant, les joueurs pensent qu’ils peuvent changer le monde dans lequel ils évoluent. Cette émotion de sauveur de monde virtuel, Jane McGonigal essaie de la transposer dans la réalité. Elle propose donc aux gens de participer à des expériences épiques au travers de son travail à “The Institute for the future”. Cet institut imagine différentes issues de scénario d’évolution du monde futur. Puis il propose ces issues aux gens qui vont imaginer comment y arriver.

“We want to imagine epic wins, and then give people the means to achieve the epic win”
“Nous voulons imaginer des victoires épiques, puis donner aux gens les moyens d’atteindre ces victoires”

Jane McGonigal
“Gaming can make a better world”

Parmi les jeux qu’elle a pu inventer, l’un d’eux, un jeu en ligne, place 1700 joueurs dans un monde où il

y a une pénurie d'essence. Cette pénurie est fictive, mais les concepteurs du jeu ont placé suffisamment de contenu sur le site internet pour faire croire au joueur que c'est réel. Ainsi lorsqu'il s'inscrit, le jeu lui envoie de fausses vidéos d'information, des données quant au coût de l'essence, ce qui est disponible, les ressources alimentaires et les transports impactés, si les écoles sont fermées, s'il y a des émeutes, etc. Grâce à ces informations, les joueurs doivent imaginer des moyens pour vivre et poster sur le blog du jeu les solutions qu'ils ont trouvées.

En transportant les joueurs dans une aventure épique, l'institut génère bien plus de motivation des joueurs à résoudre un problème mondial, qu'en leur demandant de changer leur mode de vie simplement pour améliorer le monde. Le scénario de départ devient une aventure incroyable à laquelle il faut participer. Par ailleurs une grande partie des joueurs ont conservé certaines habitudes trouvées durant le jeu et continuent de vivre avec.

Nous avons vu jusqu'ici que les actions des joueurs étaient motivées par différents aspects des jeux qui répondent à leurs besoins psychologiques. Comprendre les besoins des joueurs pour créer des univers virtuels ; puis utiliser les mécaniques de ces mondes pour les transposer dans la réalité afin de modifier le comportement humain montre bien que les jeux et la réalité entretiennent des relations très étroites. Peut être même plus que ce qui est communément admis. Dans ce mémoire je m'intéresse aux actions immédiates du joueur dans le jeu comme étant le reflet de sa personnalité. Le joueur est dans une situation particulière (par exemple face à un boss), en ressentant des émotions à un instant précis il va déterminer inconsciemment l'action qu'il va réaliser. Chaque joueur ayant une réaction particulière, il faut pouvoir lui laisser le choix de l'action qui va lui paraître la plus instinctive à réaliser dans cette situation. Au travers de cela, les actions du gameplay seront différentes pour chaque joueur qui aura sa propre manière de jouer. Tout cela dans un seul but, générer de la motivation ne serait ce que par le simple fait d'agir, de réaliser des actions, de jouer.

I.1.3) Etude du comportement des joueurs dans les univers virtuels : quand un comportement virtuel dénote une personnalité réelle

De plus en plus, ces dernières années, les concepteurs de jeu cherchent à mieux comprendre les subtilités de la personnalité des joueurs afin d'offrir un contenu de jeu plus à même de les captiver. Mais la question que les concepteurs et les psychologues se posent c'est si la personnalité des joueurs dans un jeu est vraiment significative de sa personnalité réelle. Est-ce que l'univers du jeu n'entache pas la personnalité du joueur ? Est-ce que le comportement des joueurs dans les jeux est le reflet de ce qu'ils sont vraiment ?

Jusqu'à maintenant il y a eu différentes manières de comprendre la personnalité des joueurs. Dans son livre "*Designing virtual worlds*" Richard Bartle réalise une classification des joueurs. Dans un premier temps, ils sont classés selon quatre catégories :

Les "*Achiever*" : représentent les joueurs qui aiment accomplir des défis, les différents objectifs du jeu et avoir un certain challenge.

Les “*Explorer*” : qui aime découvrir et connaître l’intégralité du contenu du jeu.

Les “*Socializer*” : qui vont orienter leurs actions vers le jeu collectif et la construction de relations sociales avec d’autres joueurs.

Les “*Killer*” : qui aiment la compétition et la confrontation avec les autres joueurs. Ils s’orientent vers le contenu PvP (player versus player) des jeux.

Ce qui ressort communément des travaux de Richard Bartle est ce graphique qui montre l’organisation des quatre catégories de joueurs, selon deux axes :

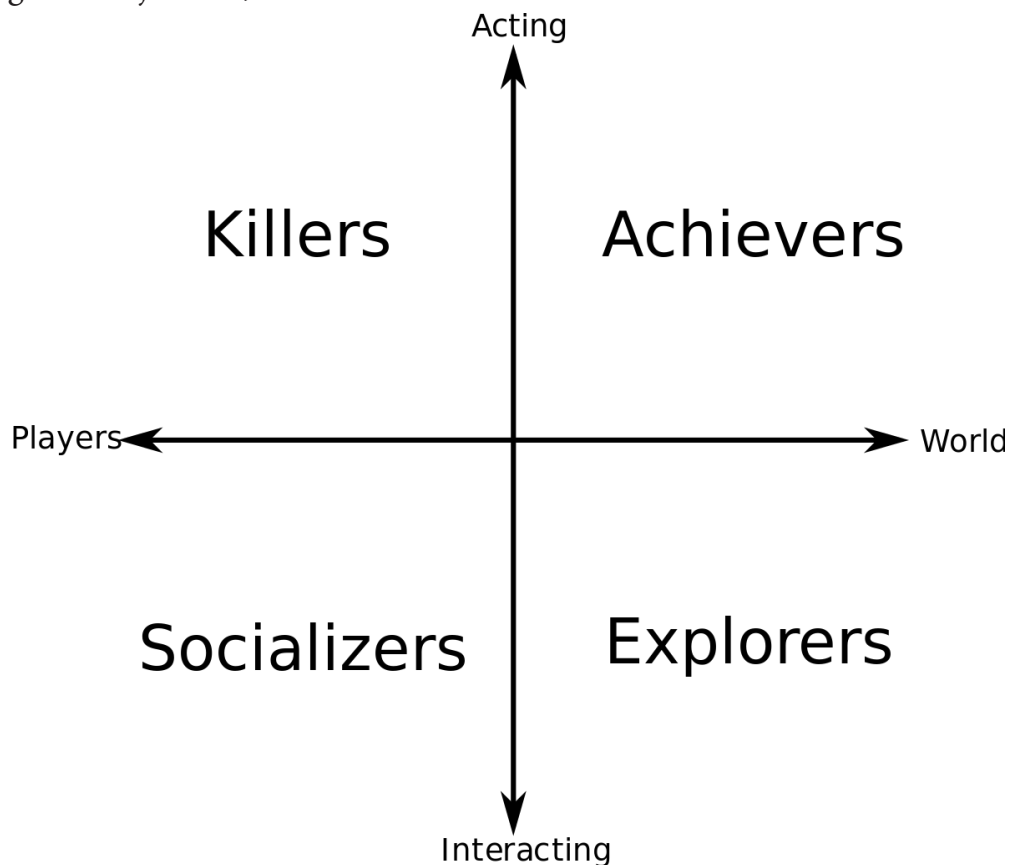


Figure 7 : Graphique représentant les types de joueurs mis au point par Richard Bartle

Au travers de ce graphique il montre les actions et interactions que les différentes catégories entretiennent entre elles et avec l’univers du jeu.

“I immediately saw that achievers and explorers were more interested in the game than in the players of the game, whereas socializers and killers were more focused on the players than any game aspects.”

“J’ai immédiatement vu que les “achiever” et “explorer” étaient plus intéressés par le jeu que par les joueurs du jeu, tandis que les “socializer” et “killer” étaient plus centrés sur les joueurs que sur n’importe quels autres aspects”

“I felt confident that achievers and killers were active, in that they like doing things to other things (the game or its players, respectively). Explorers and socializers were passive, in that they liked things being done to them.”

“J’étais sûr que les “achiever” et “killer” étaient actifs, ils aiment agir sur quelque chose (le jeu ou les joueurs respectivement). Les “explorer” et “socializer” étaient passifs, ils aiment que les choses soient faites pour eux.

*Richard Bartle
“Designing virtual worlds”*

Par ailleurs, il explique les possibles relations entre les différentes catégories et les conséquences de l’augmentation d’un type de joueur par rapport aux autres. Par exemple si dans un jeu le nombre de “Killer” augmente alors il devrait y avoir moins “d’Achiever”, un peu moins “d’Explorer” et beaucoup moins de “Socializer”. Cela peut-être parce que les Killer se concentrent uniquement sur la confrontation envers les autres joueurs entraînant moins de relations sociales positive et donc peut-être plus facilement l’isolement d’un joueur vis-à-vis des autres. Bien entendu, il n’exclue pas la possibilité qu’un joueur puisse faire partie de différentes catégories voir même de toutes les catégories.

Afin de tester la classification des joueurs de Richard Bartle, Erwin S. Andreassen et Brandon A. Downey mettent au point le Bartle Test : un questionnaire à choix binaires auquel les joueurs pouvaient répondre pour découvrir de quels types de joueur ils étaient. A la fin du test, le résultat que le joueur obtient est composé des quatre lettres désignant les types de joueur, rangé dans l’ordre de la catégorie la plus représentative de la personnalité du joueur à la catégorie le représentant le moins. Puis le pourcentage de chaque catégorie est détaillé. Par exemple le résultat de mon propre test est : ESAK, 80% Explorer, 53% Socializer, 40% Achiever et 23% Killer. Cela veut dire que je suis en priorité un Explorer, puis un Socializer, puis un Achiever, et enfin un Killer. De mon propre avis je n’ai pas l’impression que ces pourcentage me représente réellement. Tout d’abord, ce test est principalement dédié au jeu de rôle multijoueurs. Malgré cela on peut noter quelques limites au test :

- Le test se réalise en dehors des sessions de jeu du joueur. Il est lié à la vision que le joueur a de sa propre manière de jouer et n’est peut être pas à 100% objectif.
- Toutes les questions n’offrent au joueur qu’une seule réponse possible alors qu’à certaines d’entre elles je n’avais aucune préférence pour une réponse précise et aurait voulu choisir les deux.
- De même si je ne me sens pas concernée par aucune réponse de la question, il n’y a pas de case “aucun de ces choix” que je peux cocher
- Enfin, je dirai que chacune de mes réponses seraient différentes en fonction du jeu de rôle multijoueurs auquel je joue. Ce test est souvent mis en relation avec le jeu world of warcraft, j’ai donc répondu en ayant en tête ma manière de jouer à ce jeu. Si je m’étais remémoré ma façon de jouer dans un autre jeu de rôle multijoueurs j’aurais eu des pourcentages complètement différents. Je ne peux donc pas dire que ce test est représentatif de ma personnalité générale dans les jeux de rôle multijoueur.

Au final, même si la taxonomie des types de joueur comporte quelques limites, elle reste utile pour concevoir le contenu qui convient dans les grandes lignes aux objectifs que se donnent les différents types de joueur. Néanmoins, elle repose sur la catégorisation des joueurs et non sur la possibilité d’offrir à chacun d’entre eux une expérience unique grâce au contenu du jeu qui s’adapterait individuellement à leur personnalité. Car si l’on prend tous les joueurs faisant partie de la catégorie “Explorer”, chaque joueur aimera explorer les zones du jeu, pour autant chacun aura sa manière d’explorer son environnement préféré. Est-ce que le joueur aime trouver des environnements cachés sous-terrains ?

aériens ? aquatiques ? Aime t-il uniquement explorer des paysages ? ou également des villages et avoir des relations avec les différents peuples du monde virtuel ? Si oui, quelles histoires chaque peuple peut amener au joueur pour lui faire découvrir des lieux que d'autres ne pourraient trouver ?

En 2014, Nicolas Ducheneaut et Nicholas Yee publient un article sur l'étude de la personnalité d'un joueur à partir de son comportement. L'article se nomme "*Les jeux vidéo en ligne, un miroir de la personnalité des internautes ?*" et montre les résultats de l'étude de la personnalité d'un joueur via la méthode des Big 5 (détection de la personnalité selon les cinq grands facteurs : l'ouverture, la conscience professionnelle, l'extraversion, l'agréabilité et le névrosisme). Au travers de cette étude, les auteurs cherchent à savoir si la personnalité réelle d'un joueur peut être détectée à partir de son comportement dans les jeux vidéo et si oui quels sont les indicateurs révélateurs de la personnalité des joueurs (d'où les big 5). Enfin, si ces indicateurs sont les mêmes dans les jeux que dans la réalité.

Pour réaliser cette étude, les auteurs ont demandé la participation de 1040 joueurs du jeu World of Warcraft. Un jeu de rôle en ligne multijoueurs, possédant une très grosse communauté. Chacun de ces joueurs a rempli le questionnaire des Big 5 (questionnaire de 120 affirmations dans lequel le joueur note son degré d'accord avec l'affirmation) afin que les auteurs puissent connaître leur profil sociodémographique et leur personnalité. Les résultats de leur étude sont issus de la comparaison entre les résultats du questionnaire et le comportement des joueurs dans le jeu. Pour suivre les joueurs en jeux, un logiciel collecte les données des joueurs pendant quatre mois en mesurant quotidiennement 3500 variables comportementales pour chaque personnage que les joueurs possèdent. Les jeux de rôles multijoueurs sont des environnements virtuels qui facilitent la collecte de données. Notamment, car le système informatique traque le comportement de chaque avatar pour permettre les interactions entre les joueurs et l'environnement. Par ailleurs les serveurs gardent en mémoire toutes les ressources accumulées par les joueurs : items, statistiques de leur avatar, localisation, objectifs et quêtes accomplis... Enfin, la collecte de données se fait de manière transparente pour les joueurs qui ne modifient pas leur comportement en se sentant observés. Par ailleurs le site Battle.net qui est le site d'information des différents produits du studio Blizzard, dont le site d'information de World of Warcraft, garde également en ligne les informations de l'avatar du joueur. Ces informations peuvent être consultées par les autres joueurs et a assisté les auteurs de cette étude.

A la fin des quatre mois de collecte d'informations, Nicolas Ducheneaut et Nicholas Yee ont comparé le comportement des joueurs dans le jeu aux informations recueillies grâce au questionnaire des Big 5. Voici ce que les auteurs ont remarqué pour chaque trait des Big 5 :

- L'extraversion : les joueurs ayant un résultat élevé d'extraversion semble faire plus d'activités coopératives en groupe comme des donjons demandant un grand nombre de joueurs. A l'inverse, les joueurs préfèrent les activités solitaires, la collection de mascottes, la compétition dont ils sortent le plus souvent gagnant et créent plus facilement des personnages du sexe opposé.

- L'agréabilité : les joueurs ayant un résultat élevé d'agréabilité réalisent plus d'activités coopératives ou non violentes comme l'exploration, l'évolution de métier et participent plus aux événements mondiaux (Noël, Pâques, Halloween, etc). A contrario, les joueurs au résultat plutôt bas sont plus axés vers la compétition (plus de victoire dans les arènes et les champs de bataille) et sont plus égoïstes dans l'obtention d'objet.

- La conscience professionnelle : les joueurs possédant un résultat élevé collectent plus d'objets dans un but non combatif (mascotte, monture, apparence d'armure), évolue des métiers comme la cuisine et la pêche et participent plus aux événements mondiaux. A l'inverse, les autres joueurs sont plus spontanés et ont tendance à mourir plus souvent dans le jeu.

- Le névrosisme : les joueurs possédant un résultat élevé créent plus de personnages du sexe opposé alors que les joueurs possédant un résultat plus faible réalisent plus d'activités de joueur contre joueur et ont plus de victoire dans ce domaine.

- L'ouverture : les joueurs ayant un résultat élevé d'ouverture créent plus de personnages sur plus de serveurs différents et passent plus de temps en jeu à explorer et à réaliser des activités pacifiques. A l'inverse les autres joueurs recherchent les activités plus traditionnelles du jeu comme les combats ou les donjons.

Au travers de leur étude, les auteurs semblent avoir trouvés des similitudes entre la personnalité réelle des joueurs et leur comportement dans un univers virtuel. Une personne avec un haut pourcentage d'extraversion aura tendance à s'orienter vers le jeu collectif et le partage avec d'autres joueurs. Cela correspond à la catégorie *"Socializer"* de Bartle.

"Les résultats montrent que la personnalité s'exprime en ligne à travers des indicateurs stables qui reflètent la plupart du temps les définitions classiques des cinq facteurs dominants en psychologie."

Nicolas Ducheneaut et Nicholas Yee

"Les jeux vidéo en ligne, un miroir de la personnalité des internautes ?"

Ce genre d'étude peut tout à fait permettre au concepteur de jeu de mieux appréhender les joueurs et de proposer des expériences plus proches de leurs besoins. Les cinq facteurs peuvent tout à fait se mettre en relation avec l'assouvissement des besoins psychologiques dont parlait Tony Robbins. Par exemple, si le degré d'extraversion est élevé, le besoin de relation sera plus facilement comblé ce qui veut également dire que cette personne a un besoin plus évident de relation qu'une personne ayant un taux d'extraversion plus bas. Le besoin de relation se comblera différemment, peut être plus par la sensation de faire partie d'une équipe, même éphémère, d'arène ou de champ de bataille. De plus, ce sera peut être plus le besoin d'importance qui sera ici comblé, car en champ de bataille les joueurs affirment leur domination sur les autres.

Par ailleurs, les auteurs définissent les limites de cette étude. Les tests ont été réalisés sur un jeu de type donjon et dragon (comme pour le test de Bartle) et donc les résultats pourraient être différents sur d'autres type de jeu. De plus, ils évoquent certaines études qui montrent que les joueurs peuvent idéaliser leur comportement dans les jeux comme sur les sites de rencontre, ce qui pourrait affecter la validité de leur étude.

Bien que très intéressante, cette étude analyse la personnalité du joueur sur l'ensemble d'un jeu en étudiant ses interactions avec chaque élément du jeu. L'étude traite du comportement globale du joueur alors que dans ce mémoire je souhaite aborder les actions directes du joueur, les compétences du personnage qu'il utilise (aussi dit les *"skills"* du personnage). On pourrait tout à fait réaliser la même

étude, mais en se concentrant uniquement sur le gameplay de l'avatar du joueur. Par exemple, dans leur étude, Nicolas Ducheneaut et Nicholas Yee prennent en compte les différents personnages que les joueurs jouent (classes et sexe) mais à aucun moment ne pose la question de savoir si la première classe choisie par le joueur correspond à sa personnalité via comparaison avec le questionnaire des Big 5. Certains joueurs ne jouent qu'une seule classe de personnage, d'autres en jouent plusieurs. Pourquoi ? Quelle différence entre ces joueurs ? Est-ce qu'un joueur ne joue qu'une seule classe parce que celle-ci correspond complètement à sa personnalité ? A contrario un joueur jouerait différentes classes parce que la première ne conviendrait pas complètement à son profil et le joueur trouverait à satisfaire ses besoins via d'autres classes en complément. Ou alors, un joueur en jouerait-il plusieurs parce que son pourcentage "d'Achiever" serait élevé ? Plutôt que de catégoriser les joueurs afin de créer du contenu pour les différentes catégories je souhaite travailler sur un système de jeu, qui, à partir du même contenu programmé (compétence du personnage, modèles 3D, etc), adapte ce contenu individuellement à chaque joueur. Bien entendu, cela veut dire qu'il faut programmer un plus grand nombre de compétences et de modèle 3D que ce qui ne sera utilisé par un même joueur. Par exemple programmer une cinquantaine d'actions différentes même si le joueur ne va en utiliser que vingtaine. Chaque joueur utiliserait une vingtaine de compétences, mais en faisant différentes combinaisons de ces compétences. Autrement dit le joueur ne débloquera pas les cinquante compétences, mais uniquement celles qui conviendront à son profil. Pour autant, chaque joueur étant différent, il débloquera les compétences qui lui conviennent dans les cinquante programmées de base. Tout cela pour donner l'impression au joueur qu'il y a une infinité de possibilités et que son gameplay est unique.

I.2) Intégrer la personnalité des joueurs dans un jeu

Comprendre la personnalité et le comportement des joueurs permet aux concepteurs de jeu de proposer des univers plus adaptés à leurs besoins. Pour autant cela passe la plupart du temps par des questionnaires que les joueurs remplissent ou sur l'étude de l'utilisation du contenu déjà existant. Mais comment faire en sorte que le jeu lui-même s'adapte en temps réel au comportement des joueurs ? Pour cela il faudrait concevoir un jeu dont le ou les premiers niveaux seraient fixes (conçu par les game et level designers), cela permettrait au jeu de "calculer" le comportement du joueur. Puis, pour les niveaux suivants, d'organiser l'espace du niveau en fonction du "calcul" précédemment réalisé. C'est-à-dire que la forme de la zone serait définie par les concepteurs du jeu, mais le placement des objets sur le terrain serait organisé par le système du jeu en fonction du comportement du joueur précédemment "calculé". Autrement dit, créer une intelligence artificielle pour gérer ce système.

Que ce soit pour donner du challenge ou assister le joueur, l'intelligence artificielle prend aujourd'hui une énorme place dans le monde des jeux vidéo. Lorsque celle-ci régit les comportements des personnages non joueurs que le joueur va rencontrer (alliés comme ennemis) il faut que ces comportements lui paraissent naturels, logiques et équilibrés par rapport aux capacités d'un être humain. Par exemple un ennemi qui tire avec une arme à feu et qui touche constamment sa cible ne paraît pas naturel, voir même peut donner au joueur l'impression de tricher. C'est pourquoi une marge d'erreur est programmée aux intelligences artificielles pour imiter le comportement humain. Dans tous les cas c'est le modèle du vivant qui inspire la programmation des phénomènes comportementaux des intelligences artificielles.

"Ces algorithmes s'inspirent de modèles issus des sciences cognitives et des sciences du vivant — notamment du connexionnisme et de la génétique."

Marie-Hélène Tramus, Michel Bret, Edmond Couchot
"La seconde interactivité : vers de nouvelles pratiques artistiques"

Dans les jeux, lorsqu'un algorithme est programmé pour comprendre le comportement du joueur et parfois calculer un comportement optimisé selon différents paramètres, les programmeurs utilisent des systèmes apprenant appelés réseaux de neurone. Dans l'article "Jeux vidéo et intelligence artificielle située", Benoît Girard, Gabriel Robert et Agnès Guillot, parle d'un jeu de simulation de régates nommé Virtual skipper 2, qui analyse le comportement du joueur pendant une course et lui signale après celle-ci les moments où son comportement a été inadapté à la situation en fonction des variables telles que l'orientation et la vitesse du vent, la direction et la vitesse du bateau, le sens du courant, etc. En identifiant les données que le joueur génère durant la course, l'intelligence artificielle va catégoriser son comportement. Les différentes catégories ont été, au préalable, fixées grâce aux connaissances d'experts en navigation et aux recherches des concepteurs du jeu. L'observation du joueur dans le jeu se fait au moyen d'un animat qui simule les sens du joueur afin de pouvoir collecter les mêmes données que celui-ci et catégoriser sa perception du jeu à un instant T. Chaque réseau de neurone permettant de calculer le comportement idéal pour la course se spécialise dans une des catégories présentes en

donnée d'entrées. Ces entrées correspondent aux données sensori-motrices retenues par l'animat à un instant T. Chaque réseau va alors faire une prédiction du comportement en sortie à T+1. Puis les données réelles sont comparées aux prédictions de chaque réseau neuronal pour déterminer leur crédibilité vis-à-vis de chaque catégorie. A long terme chaque réseau se spécialise dans la prédiction de la catégorie où il a eu le moins d'erreurs : là où la crédibilité est la plus forte.

Dans ce jeu l'intelligence artificielle tente d'orienter le comportement du joueur vers un comportement idéal et optimisé pour réussir parfaitement une course. Le comportement du joueur est modifié par le jeu. Bien qu'intéressant, ce que je souhaite obtenir dans ce mémoire est l'effet inverse : le joueur conserve son comportement et agit selon sa personnalité. C'est donc l'intelligence artificielle qui doit comprendre ce comportement et proposer un développement du jeu qui va intéresser le joueur. Autrement dit ce n'est pas le joueur qui doit changer, mais le jeu.

Dans le temps imparti de ce mémoire et avec le peu d'expérience en réseau neuronal, programmer un réseau de neurone me semblait trop complexe. De plus, un problème majeur semblait se poser : le réseau de neurone a un temps d'apprentissage. Si je souhaite qu'un jeu s'adapte à plusieurs centaines de comportements différents, je dois pouvoir simuler tous ces comportements pour que le réseau apprenne de chacun d'eux. Par ailleurs il me semblait difficile de simuler autant de comportements de mon seul point de vue. En effet, il n'y a que mon propre comportement que je saurai retranscrire fidèlement. Mais comment pourrai-je imiter aussi bien le comportement imaginé d'une seconde personne pour le simuler ? Je risquais d'avoir un trop grand nombre d'erreurs. Il me fallait donc le moyen de comptabiliser le nombre de fois qu'une action est utilisée. Au début de leur livre *"Why we do what we do : understanding self motivation"*, Edward L. Deci & Richard Flaste parlent des récompenses qui sont données à chaque fois qu'un comportement désiré est obtenu.

"Reward the desired behavior, and there is increased likelihood that the behavior will be repeated"
"Récompenser le comportement désiré, et la probabilité que le comportement se répète augmentera."

Edward L. Deci & Richard Flaste
Why we do what we do : understanding self motivation

Ce principe vient du behaviorisme et plus précisément de la psychologie de renforcement de Burrhus Frederic Skinner. Selon ce concept, l'organisme doit avoir un besoin et y répondre par une action. Le résultat de cette action doit être renforcé. Il existe deux types de renforcement :

- Le renforcement positif qui correspond à la récompense pour avoir correctement agité. On encourage le comportement à se reproduire.
- Le renforcement négatif qui est égal à la punition et décourage le comportement à se reproduire.

Le renforcement accroît la probabilité d'émission ou de diminution d'une réponse. Plus on renforce positivement une action et plus on a de chance qu'elle se reproduise et inversement pour le renforcement négatif. De plus, il existe deux catégories de renforcement chacun possédant ses renforcements positifs et négatifs :

-
- Le renforcement primaire qui correspond aux besoins biologiques ou aux réflexes d'une personne.
 - Le renforcement secondaire qui correspond à l'apprentissage. Comme le chien qui s'assoit lorsqu'on lui donne une friandise.

Ce concept psychologique est utilisé dans les jeux vidéo de différentes manières. La plus imagée et correspondant le plus fidèlement à la définition psychologique du renforcement de Skinner est celle que l'on peut trouver dans le jeu "Black and White 2", un jeu de stratégie où le joueur peut éduquer une créature qui pourra ensuite participer à la conquête de territoire. Cette créature agit selon un arbre de comportement. Elle peut aider le village à se développer en récoltant avec les villageois récolteurs, participer aux batailles et défendre la cité, ramasser de la pierre, du bois ou encore, aider à la construction de bâtiments. Cependant, ce comportement peut être renforcé par le joueur. En effet il peut éduquer sa créature en la caressant ou en lui donnant des gifles pour chaque action que la créature peut faire. Dans le cas ci-dessous, la créature dit qu'elle fera ses besoins sur les bâtiments du joueur.



Figure 8 : Le joueur gifle sa créature pour en changer le comportement

Si le joueur ne souhaite pas que sa créature le fasse, il lui donnera des gifles jusqu'à ce que la créature dise "Je ne ferai jamais mes besoins sur les bâtiments du joueur".



Figure 9 : La créature a modifié son comportement

De même s'il souhaite encourager sa créature à construire des bâtiments, il la caressera lorsqu'il aura sélectionné la catégorie correspondante. Ainsi à chaque fois que la créature verra un bâtiment à construire elle aidera les villageois à la construction.

Dans d'autres jeux, le renforcement est utilisé pour augmenter la capacité des armes que le joueur utilise. C'est le cas dans le jeu Borderlands, créé par le studio américain 2K, dans lequel chaque arme que le joueur peut utiliser correspond à une catégorie. Par exemple : sniper, fusil à pompe, arme élémentaire, etc.



Figure 10 : Jauge de chaque arme avec leur niveau d'utilisation

Plus une arme est utilisée et plus la catégorie correspondante gagne de points d'expérience. Lorsque la jauge est complète, la catégorie gagne un niveau et améliore une caractéristique de l'arme comme la vitesse de rechargement de l'arme, la capacité du chargeur, la cadence de tir, etc. Plus le joueur renforce une catégorie et plus les armes appartenant à cette catégorie seront puissantes. En général, le renforcement des catégories correspond à la manière de jouer du joueur : aux armes avec lesquelles il préfère jouer.

Cette méthode me semble adaptée pour comptabiliser le nombre de fois qu'une action est utilisée par le joueur. Elle est également suffisamment simple, rapide et adaptable pour pouvoir l'intégrer à ma problématique. Je pense donc en faire la base de mes tests.

I.3) Les jeux vidéo qui ont inspirés ce mémoire

Je vais aborder ici les jeux qui utilisent des mécaniques ou des concepts que j'aborde dans mon mémoire et qui m'ont inspiré dans mes recherches et mes tests.

The Elder Scroll Online

The Elder Scroll Online est un jeu de rôle multijoueurs (aussi nommé MMORPG) conçu par le studio américain Bethesda à l'origine de la franchise Skyrim dont le jeu tire son univers. Dans ce jeu le joueur crée physiquement son personnage et lui choisit une "classe" qui correspond à un type de gameplay particulier : chevalier dragon, templier, lame noire et sorcier. Le joueur fait évoluer son personnage de niveau en niveau (jusqu'au niveau 50) en gagnant des points d'expérience. Ce qui m'intéresse dans ce jeu c'est l'évolution du personnage et plus particulièrement celle de ses compétences. En effet il y a deux types d'évolution :

- L'évolution du personnage : lorsque le personnage gagne un niveau il gagne trois points de caractéristiques et un point de compétences. Les points de caractéristiques se dépensent dans les statistiques physiques du personnage à savoir la vigueur, la santé et la magie, qui correspondent à l'endurance du personnage (lui permettant d'esquiver plus souvent ou de sprinter plus longtemps), la vitalité du personnage (son nombre de point de vie) et le mana du personnage (son essence magique lui permettant de lancer des sortilèges). Les points de compétences, eux, permettent au joueur de débloquent une compétence active (action d'attaque) ou passive (effet appliqué constamment au personnage sans avoir besoin de l'activer) ou bien de faire évoluer une compétence déjà débloquée. Par exemples si je mets un point dans la compétence "envoyer une boule de feu", la compétence ne fera non plus 50 points de dégâts mais 70.



Points de caractéristiques : 0			
Magie	Santé	Vigueur	
11	2	0	
Magie maximale	21446	Récupération de Magie	1290
Santé maximale	16149	Récupération de Santé	508
Vigueur maximale	14563	Récupération de Vigueur	774
Dégâts des sorts	1701	Dégâts des armes	1555
Critique magique	10,0 %	Critique physique	10,0 %
Résistance aux sorts	9140	Résistance physique	8711
Résistance aux critiques	0		

Figure 11 : Statistiques que le joueur peut améliorer grâce à ses point de caractéristiques

- L'évolution de la compétence d'arme du personnage. Le personnage du joueur est équipée d'une arme. Chaque classe pouvant posséder différents types d'arme. Plus le joueur va utiliser un type d'arme et plus la barre d'expérience de ce type d'arme va se remplir jusqu'au maximum. Lorsqu'elle l'atteint, la compétence d'arme du personnage gagne un niveau et la jauge de ce nouveau niveau retombe à zéro jusqu'à être de nouveau remplie. Lorsque la compétence d'arme atteint un certain niveau, une compétence active liée à l'utilisation de ce type d'arme devient disponible au joueur. C'est-à-dire que lorsqu'une compétence active est disponible, le joueur peut dépenser ses points de compétences dans cette action pour pouvoir la débloquent et l'utiliser (c'est le même principe qu'un achat).



Figure 12: Compétences du personnage qui se débloquent en fonction du niveau de la compétence (ici la magie noire est au niveau 17) et que le joueur peut acheter moyennant des points de compétences.

Autrement dit la disponibilité des compétences actives du personnage est liée au renforcement de cette arme. C'est la technique de renforcement utilisée en intelligence artificielle pour augmenter l'effet de

l'action renforcée. Ce principe de renforcement dans le jeu m'intéresse, car il laisse la possibilité au joueur de renforcer le type d'arme qui convient le plus à sa manière de joueur, puis en fonction de cela, de choisir lui même les compétences actives et passives qu'il va débloquent avec ses points de compétence.

Black Desert

Black Desert est un jeu de rôle multijoueurs créé par le studio coréen Pearl Abyss. Tout comme dans The Elder Scroll online, le joueur crée physiquement son personnage et lui choisit une classe. Cependant les compétences que le personnage va débloquent ne sont pas liées aux choix du joueur mais sont uniquement liées au niveau du personnage. C'est-à-dire qu'à certains niveaux le personnage va obligatoirement débloquent les compétences que les concepteurs du jeu ont voulu rendre accessible à ce stade. Le joueur n'a pas de choix quand au gameplay de son personnage. Néanmoins, le jeu exploite lui aussi une technique de renforcement. Même si celle-ci n'est pas liée au personnage, elle reste dépendante des actions du joueur et a un impact direct sur les informations que lui fournisse l'environnement. Plus le joueur va effectuer une action et plus l'environnement va répondre. Par exemple lorsque le joueur affronte pour la première fois un ennemi, il ne possède aucune information sur cet ennemi : il ne voit pas sa quantité de point de vie, ne connaît pas ses faiblesses, son comportement, etc. Plus il va affronter ce type d'ennemi et plus les informations concernant l'ennemi vont se débloquent, la jauge de point de vie va apparaître, les résistances de l'ennemi seront identifiées, etc.

La barre de vie de l'ennemi ne s'affiche pas

La barre de vie de l'ennemi s'affiche



Figure 13 : Plus le joueur tue de belettes et plus il va acquérir d'informations sur l'animal. Comme par exemple, la barre de vie de l'ennemi qui s'affiche.

Bioshock Infinite

Bioshock Infinite est un jeu de tir à la première personne (ou FPS : first person shooter), troisième opus de la série Bioshock et créé par le studio américain 2K. Le joueur incarne Booker DeWitt, un détective endetté dont la mission consiste à libérer une jeune femme nommée Elizabeth de la ville flottante de Columbia. Ce qui m'intéresse dans ce jeu c'est la possibilité offerte au joueur de pouvoir faire apparaître un élément du décor à un endroit précis. En effet, le personnage Elizabeth, qui suit et aide le joueur dans ses aventures, a la capacité d'ouvrir des failles spatio-temporelles dans certaines zones de l'environnement. Ces failles permettent d'importer un élément de décor dans le même espace temps que le joueur pour qu'il puisse l'utiliser. Il peut ainsi faire apparaître des troussees de soin, différentes tourelles, des "cover" (élément permettant de se mettre à couvrir), des munitions des attaches pour accrocher votre grappin, etc. Ces éléments que le joueur peut faire apparaître lui permettent de modéliser sa zone de combat et de l'optimiser en fonction des situations et de ses besoins durant un combat. Par exemple si le joueur a besoin de régénérer sa vie, il peut faire apparaître des troussees de soin à un emplacement précis. Attention cependant chaque faille à un nombre limité de décor qu'elle peut faire apparaître. Aucune d'entre elle ne peut faire apparaître tous les éléments possibles de décor.



Figure 14 : Le joueur voit les différents décors qu'il peut faire apparaître.

No man's sky

No man's sky est un jeu d'exploration spatiale en vue à la première personne, conçu par Hello Games. Le joueur explore des planètes qui ont été créées par génération procédurale. Chaque planète possède ses propres caractéristiques. Elles sont toutes différentes que ce soit en terme de radioactivité, température, environnement terrestre ou aérien (planète plus ou moins aqueuse, montagneuse, etc), diversité de la faune et de la flore. C'est cette génération procédurale engendrant chaque planète avec ses caractéristiques et ses créatures qui m'intéresse. Il y a au total 18 446 744 073 709 551 616 planètes générées

et qui peuvent être visitées par les joueurs. Lorsqu'un joueur découvre pour la première fois une planète, il peut lui donner un nom (ou garder le nom généré à l'origine) et le prénom de son personnage apparaît dans la page d'information de la planète que d'autres joueurs pourront voir en la visitant. Ce qui paraît surprenant c'est que le jeu est très léger, à peine une dizaine de gigas installée (9,67 Go) et tout cela, car le jeu comprend très peu de modèle 3D mais génère procéduralement des centaines de variantes de ces modèles en fonction des paramètres définis : multiplication de mesh, joints, faces, paramètres des collisions, etc.

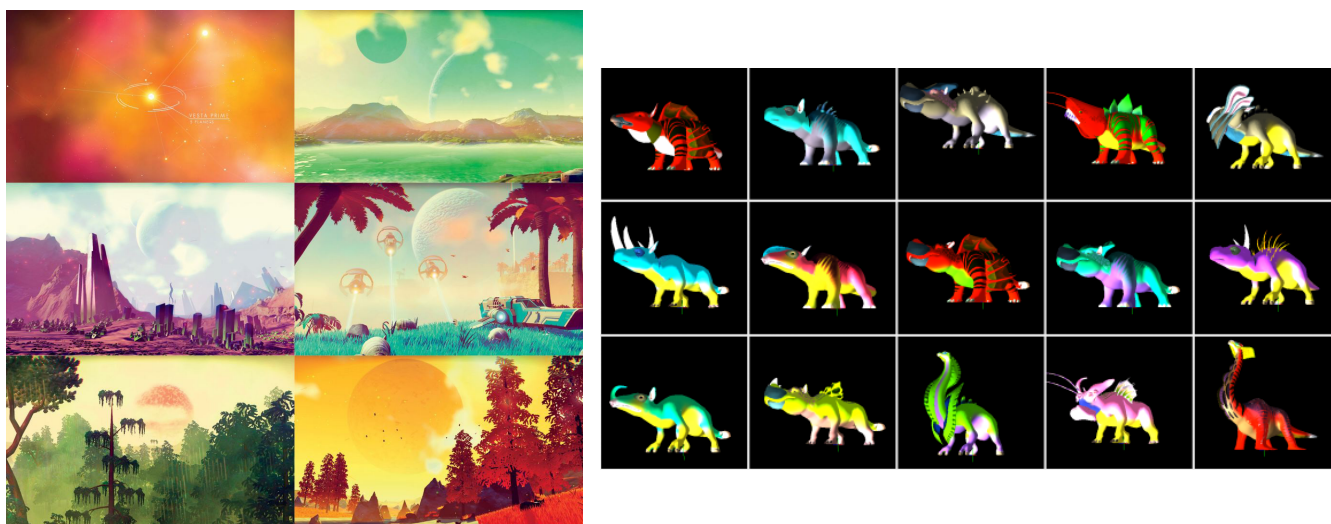


Figure 15 : Différents univers et modélisations créés par génération procédurale dans No man's sky.

Bien que cette génération procédurale générant une multitude de modèles 3D différents soit très intéressante. Je ne me suis pas donné pour objectif de générer différents modèles 3D de mon décor, mais de générer l'apparition de ces modèles 3D à différents emplacements, moyennant les données obtenues par le "calcul" du profil comportemental du joueur.

Life is strange

Life is strange est un jeu narratif épisodique à choix multiples créé par le studio français Dontnod. Dans ce jeu, le joueur est amené à faire des choix qui vont impacter la suite du déroulement narratif, mais également les relations entre les personnages. Il y a différentes conséquences possibles :

- les conséquences sur le court terme : le joueur fait un choix et voit les conséquences quelques secondes après
- les conséquences sur le long terme : le joueur fait un choix crucial qu'il sait va engendrer quelque chose plus tard, mais ne se doute pas de l'amplitude de ces conséquences.

Dans le premier cas si les conséquences ne plaisent pas au joueur il peut remonter dans le temps grâce aux pouvoirs de son personnage et faire de nouveau face à la question pour répondre différemment. Autrement dit, lorsque le joueur a trouvé "sa solution idéale", il accepte les éventuelles conséquences qui en découleront. Le choix du joueur correspond à ce qui convient le plus à ce qu'il est dans la situation donnée (en général il est difficile de faire un choix contraire à sa propre morale même dans un jeu). Ce qui peut donc dire que le choix reflète la personnalité du joueur à cet instant. Autre notion

intéressante du jeu ; à chaque fin d'épisode le jeu affiche les statistiques des réponses des joueurs pour chaque choix qu'ils ont dû faire dans l'histoire.

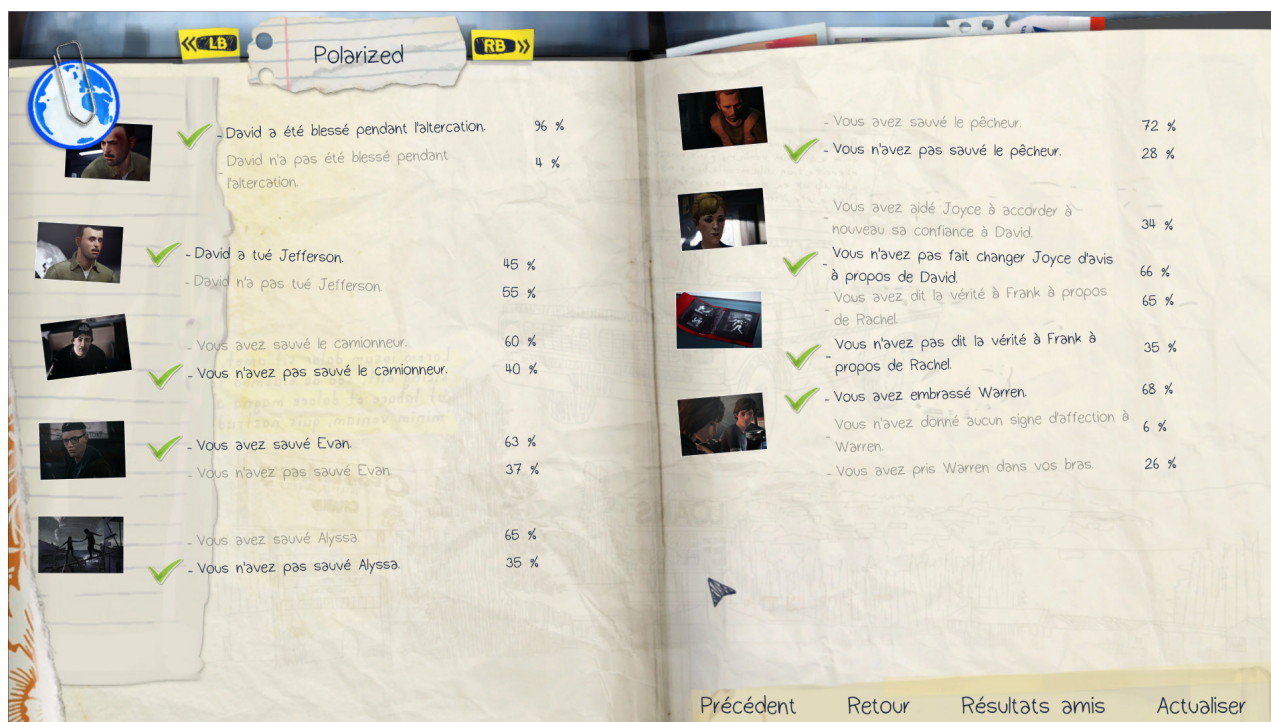


Figure 16 : Statistiques mondiales des choix des joueurs. L'encoche verte correspond au choix que le joueur a fait lui-même.

Même si le jeu comporte des situations par lesquels le joueur doit passer (peu importe les choix pris), la manière d'aborder ces situations seront différentes. Au final, même si les joueurs doivent répondre aux mêmes choix, la combinaison de toutes leurs réponses à ces choix sera différente et ils auront moins de chance de vivre les différentes situations de la même manière, en ressentant les mêmes émotions. Par exemple, en fonction des choix précédemment fait, un personnage du jeu vient en aide au joueur dans son enquête ou non. Par contre, la situation où le personnage du joueur enquête est obligatoirement jouée. On a donc deux situations différentes : situation A, le personnage X vient en aide au joueur ou situation B, le joueur doit se débrouiller par lui-même. Il n'y a que deux situations pour autant, ce sont les choix du joueur qui l'amène à ces situations et de ce fait il a l'impression que le jeu s'adapte à sa personnalité.

Kingdom of Amalur : Reckoning

Kingdom of Amalur : Reckoning est un jeu de rôle qui a été développé par 38 Studio et Big Huge Game dans lequel le joueur est plongé dans un monde fantastique et où il choisit de développer son personnage selon trois arbres de talent qui correspondent respectivement aux classes de guerrier, sorcier et rôdeur. Les arbres de talent sont des arbres de compétences dans lesquels les joueurs peuvent dépenser des points qu'ils accumulent en faisant progresser leur personnage de niveau en niveau, pour débloquer des compétences qu'ils pourront utiliser par la suite.



Figure 17 : Arbre de talents de sorcellerie de Kingdom of Amalur : Reckoning dans lequel le joueur dépense des points de compétence.

Ce qui m'intéresse particulièrement dans Kingdom of Amalur c'est qu'un joueur peut dépenser des points dans autant d'arbres de talent qu'il souhaite pour ne pas rester bloqué à jouer qu'une seule classe. Par exemple il peut très bien jouer un sorcier avec des compétences de rôdeur pour se faufiler en étant invisible, ou encore jouer un guerrier avec des compétences magiques, voir même utiliser des compétences de chaque arbre. Le joueur crée sa propre manière de jouer en mélangeant les différentes aptitudes que son personnage peut obtenir.

Conclusion

Bien que virtuels, les univers que présentent les jeux vidéo sont régis par des règles que les game et level designers ont conçues pour immerger le joueur. La barrière entre monde réel et virtuels s'amincit progressivement. De par leur virtualité, les jeux ont cet avantage de permettre aux joueurs de vivre des aventures épiques et de ressentir des émotions qu'ils ont peu l'habitude de ressentir dans le monde réel. Les limites sociales et physiques sont repoussées et libère le joueur de toute pression en lui apportant des sensations et émotions qui répondent à ses besoins psychologiques. Le joueur peut alors les assouvir dans les jeux bien plus facilement, rapidement et intensément que dans la réalité. Les jeux vidéo exacerbent les émotions et c'est la spontanéité qui prime : être comme le joueur est réellement au travers de ses actes plutôt que de paraître en devant réfléchir aux conséquences dans la réalité. Il n'y a pas de peur de l'échec, mais uniquement des instants où le joueur peut se surpasser en découvrant de quoi il est capable. Quand bien même c'est un avatar qui le représente sa personnalité s'exprime de par sa façon de jouer, ses choix d'actions et de dialogue. C'est pourquoi je n'arrive pas à adhérer à cette idée populaire qui tend à affirmer que les jeux vidéo ne sont qu'un refuge, ou encore, que parce qu'ils sont virtuels, ce qui en découle n'est qu'illusion. Selon moi, les jeux vidéo sont, pour les joueurs, un moyen d'optimiser leur vie dans le sens où ils vivront bien plus d'aventures et ressentiront bien plus d'émotions intenses au travers d'eux. Par ailleurs, certains jeux permettent aux joueurs d'en apprendre plus sur eux-mêmes et abordent des sujets parfois controversés qu'il serait bien plus difficile d'aborder dans la réalité. Je ne crois pas que les joueurs cherchent refuge, je pense simplement qu'ils ne se contentent pas uniquement des expériences que leur apporte leur vie réelle. Le monde réel et les mondes virtuels sont des réalités dans lesquels il est bon de vivre des expériences que seul chacun d'eux peut nous apporter.

Toutefois, aussi immersifs soient-ils, les jeux restent limités. Notamment en ce qui concerne les actions que les joueurs peuvent réaliser et dont l'évolution, ou l'utilisation est souvent déterminée à l'avance par les concepteurs du jeu. Les jeux comportent donc toujours le risque de ne pas laisser au joueur la possibilité de faire ce qu'il désire et donc de limiter ou de diminuer sa motivation. Quand bien même certains jeux essaient de comprendre le comportement du joueur, cela est pour calculer un comportement optimal que le joueur va chercher à atteindre en modifiant son comportement. En parallèle depuis quelques années, les études tentent à montrer que la personnalité des joueurs dans le monde réel motive leurs actions et leur comportement dans le monde virtuel. Au travers de leur avatar, les joueurs agissent selon leur personnalité, leur image du monde et leurs émotions, même si cela n'est pas toujours utilisé pour concevoir un gameplay, et plus précisément les actions du gameplay du joueur. Il serait intéressant de permettre au joueur d'utiliser les actions, les compétences de son personnage en fonction de sa personnalité. Motiver un joueur à poursuivre un jeu ne serait-ce qu'en jouant de sa propre façon serait un idéal à atteindre. C'est pourquoi j'ai souhaité axer ma problématique de mémoire sur l'élaboration d'un système de jeu qui adapte l'évolution des compétences du personnage du joueur et l'agencement du décor du jeu en fonction du comportement du joueur.

II : Développement d'un système génératif

Introduction

Cette année de master a été ponctuée par différentes expérimentations cherchant à répondre à la problématique de ce mémoire, à savoir : Comment générer un gameplay et un décor dynamique de jeu vidéo à partir du profil comportemental du joueur ? Aussi j'ai divisé cette problématique en trois parties : La première partie concerne la compréhension du comportement du joueur et plus précisément la quantification de ses actions dans l'environnement du jeu. Je dois connaître les actions que le joueur va aimer utiliser pour pouvoir les faire évoluer et créer de nouvelles actions qu'il va pouvoir débloquent. Cette partie comporte deux problématiques : comment appréhender le joueur et concevoir des actions pouvant répondre à n'importe quelle manière de jouer ? et, comment quantifier les actions ?

La seconde partie est une partie de transition entre la compréhension du comportement du joueur et l'agencement du décor en fonction de ce comportement. Il s'agit ici de récupérer les données comportementales de la première partie et de les soumettre aux règles qui vont décider de l'évolution des actions du joueur au travers des compétences de son personnage et des nouvelles actions qu'il va pouvoir acquérir.

Enfin, la dernière concerne la génération de l'agencement du décor du jeu en fonction des données obtenues en sortie de la première et de la seconde partie.

Autrement dit je devais concevoir entièrement un niveau de jeu avec son décor, et donner différents types d'actions au joueur qu'il peut utiliser librement. Pendant que le joueur résout la situation du niveau (tuer des ennemis), ses actions sont quantifiées. L'évolution des actions et l'acquisition de nouvelles se font en temps réel. Une fois que tous ses adversaires sont morts, le jeu modifie l'agencement du décor en fonction des données comportementales obtenues. L'avantage qu'ont les jeux vidéo, c'est que lorsque le joueur est en action sous pression (pendant les combats par exemple), il réagit souvent instinctivement. Dans ce cas il doit pouvoir utiliser l'action qui lui paraît être le plus en adéquation avec sa personnalité. C'est à différencier des actions que le joueur va utiliser parce qu'elles lui paraissent plus rentables. Par exemple, quelle arme fait le plus de dégâts en fonction du type d'ennemi. Ce genre d'action peut également satisfaire les joueurs mais laisse un risque que certains d'entre eux n'apprécient pas d'agir de cette manière, car elle ne leur correspond pas (donc diminution de leur motivation intrinsèque à jouer). L'objectif de la conception d'un jeu dont le gameplay s'adapte au profil comportemental du joueur est de garder cette motivation intrinsèque du joueur à jouer et ainsi de pouvoir lui fournir une expérience unique et optimale (ou en tout cas de s'en rapprocher). Tout comme le joueur évolue dans un univers, les actions de gameplay sont également un moyen de répondre aux besoins que Tony Robbins définit dans sa conférence. Par ailleurs je trouve dommage que les études comportementales sur les joueurs de jeux vidéo ne se concentrent uniquement que sur les différents types d'interaction du joueur dans son environnement virtuel, car ces études gardent un regard sur l'environnement dans sa globalité plutôt que de se concentrer sur une partie de cet univers. Ce qui amène les concepteurs de jeu à placer les joueurs dans des catégories et concevoir un contenu pour ces catégories plutôt que de concevoir un contenu qui s'adapte individuellement à chaque joueur. Bien entendu, cela représenterait un coût de production supérieur pour créer un jeu de la même envergure et c'est donc peut-être pour

cette raison que les jeux n'exploitent pas encore complètement cette voie. Cela dit, je pense que l'on pourrait tout à fait adapter une étude des Big 5 aux actions des joueurs au travers de l'utilisation des compétences de leur personnage. Comme par exemple, chercher à savoir si dans les jeux de rôle les joueurs choisissent leur première classe de personnage en fonction de la symbiose avec leur personnalité. Ou encore si le choix d'une arme dans un jeu de tir à la première personne peut être révélateur des caractéristiques du comportement du joueur. Si par exemple, on prend un joueur qui aime jouer avec un fusil à pompe et un second aimant jouer avec un sniper, les deux auront ce besoin d'importance qui va se satisfaire au travers du sentiment de puissance lorsqu'ils vaincront un adversaire. Pour autant, ce sentiment de puissance ne découlera pas de la même représentation psychologique que le joueur a de lui-même dans cette situation. Si l'on pouvait interpréter la représentation mentale que le joueur a de lui-même dans un jeu alors on pourrait pousser la compréhension de sa personnalité. Dans ce cas peut-être qu'un joueur jouant principalement avec un fusil à pompe serait une personne qui aimerait prendre des risques, être spontanée jusqu'à parfois foncer tête baissée, avoir un caractère franc. Mais peut-être aussi attirer l'attention sur lui pour protéger ses alliés, être "l'appât", ou encore qu'il aimerait lorsque les choses se déroulent rapidement. A contrario, une personne avec un sniper, préférerait avoir un regard global sur une situation pour pouvoir anticiper, calculer et prendre une décision en connaissance de cause, être à l'abri du danger, ne pas prendre de risque et pourrait avoir un caractère patient et une tendance à se focaliser sur une tâche en particulier même si elle a conscience de ce qui gravite autour. Bien sûr, tout ceci n'est qu'un exemple, je n'ai nullement trouvé d'étude pouvant affirmer ce genre d'hypothèse. Cependant, je pense qu'il serait intéressant d'étudier cet aspect du jeu vidéo qui est complètement lié aux actions même du joueur et de son gameplay. Le risque de ce genre d'étude serait d'avoir une interprétation du comportement du joueur qui ne serait pas objective et donc qui pourrait très facilement être faussée. C'est donc pour cela que je me suis intéressée dans un premier temps à l'origine des actions de l'être humain. Pourquoi fait-il ce qu'il fait ?

II.1) Expérimentations personnelles

J'ai choisi de réaliser mes expérimentations personnelles sur le moteur de jeu Unreal Engine 4. En effet, ce moteur de jeu a différents avantages qui m'ont permis d'aborder mes tests plus facilement. D'abord parce que j'ai pu reprendre des figures 3D existant dans d'autres projets unreal pour les incorporer dans mes projets. Cela m'a évité bien des modélisations d'objets, de textures et d'effets spéciaux et donc un gain de temps considérable. Par ailleurs, j'ai pu importer des animations et des personnages du site Mixamo.com afin d'avoir un personnage répondant aux différents inputs actionnés par le joueur et des feedback visuels rendus par l'animation qui indiquent au joueur que son action a bien été engendrée. De plus l'outil de retargeting d'Unreal Engine 4 est très facile à prendre en main et m'a permis d'ajuster les animations aux différents mesh de personnage. Un second avantage de ce moteur de jeu est qu'il permet de coder des niveaux de jeu en "blueprint", c'est-à-dire de programmer de manière nodale. Ce qui permet, à une personne qui n'a pas un profil complet de développeur tel que moi, de pouvoir programmer des niveaux de jeu et des intelligences artificielles très facilement et rapidement. Par ailleurs, il existe une grande communauté de personnes qui met à disposition des tutoriels écrits et vidéos qui m'ont été d'un grand secours dans certaines situations. Enfin Unreal Engine 4 permet très facilement de passer du langage de programmation C++ au blueprint et de créer des relations entre les deux types de code, ce qui m'a permis de jongler facilement entre les deux.

II.1.1) Appréhender le joueur

Avant de quantifier les actions que le joueur réalise, je devais lister différentes actions que je pourrai implémenter dans un niveau de jeu. Ces actions devaient être suffisamment variées pour appartenir à différentes manières de jouer et ainsi satisfaire un plus grand nombre de comportements différents. Pour ce faire, j'ai mis à disposition sur le réseau social facebook un questionnaire auquel les joueurs de jeux vidéo pouvaient répondre. Les personnes ayant accès à ce questionnaire sont celles appartenant au groupe de la formation Arts et Technologie de l'Image ainsi que celles appartenant au groupe de la formation de licence en Game et Level Design de 2014 de l'iut de Paris 13. Le questionnaire comportait deux questions (disponible en annexe). La première demandait aux joueurs de décrire les actions de gameplay qu'ils préféreraient utiliser durant les phases de combat, en précisant le type d'arme qu'ils utilisent le plus et en décrivant leurs expériences de jeu si nécessaire. La seconde question leur demandait de lister les actions qu'ils souhaiteraient voir implémenter dans les jeux vidéo selon leur genre et qui enrichirait leur manière de jouer. Bien que je n'ai eu que peu de réponse à ce questionnaire (onze réponses), j'ai tout de même pu lister différentes actions et appréhender différents comportements. Ainsi j'ai pu remarquer que sur toutes les réponses obtenues, peu de joueurs jouent de manière polyvalente et tous ont leurs caractéristiques de jeu. Par exemple même si différents profils de joueur furtifs se dessinent, ils ont tous leurs particularités : l'un préfère attirer l'ennemi et aménager une zone dans laquelle il va pouvoir tendre des pièges à son adversaire, l'autre préfère se glisser derrière son adversaire pour le prendre par surprise en utilisant la topographie du terrain pour se cacher. De même les profils de joueurs jouant le rôle de soutien ou de soigneur dans les jeux de rôle ou les arènes ont tendance à jouer avec des armes longue distance dans les jeux de tir. Ce qui dénote une similitude de gameplay dans différents types de jeu. En effet, les soigneurs restent le plus souvent en retrait pour protéger leurs alliés

à distance, il semble donc logique qu'ils préfèrent les armes à longue distance dans les jeux de tir (afin d'avoir une vue d'ensemble de la situation et être le plus souvent dans des lieux hors d'atteinte des ennemis). En ce qui concerne la seconde question, bien que les joueurs abordent plusieurs notions, celle qui revient principalement est : l'environnement et les objets utilisables dans le jeu. En effet, ils soulignent cette limite qu'ont les jeux d'avoir un environnement et des objets qui ne soient pas complètement accessibles ou interactifs et qu'ils ne peuvent pas utiliser à différents usages. Par ailleurs, ils regrettent de ne pas avoir différentes manières de remplir un objectif et donc d'être très souvent cantonnés à ce que les game et level designer ont décidé comme chemin unique. Pour citer de nouveau Edward L. Deci et Richard Flaste, donner le choix à un individu exerce moins de pression à lui donner un objectif à accomplir. L'individu se sent libre de choisir la manière de résoudre le problème qui lui est posé et sa motivation intrinsèque à réaliser l'action ou l'objectif demandé est conservée. Aussi, avoir les actions du gameplay qui se génèrent en fonction des choix d'actions du joueur doit pouvoir lui permettre de conserver cette motivation à jouer et à trouver une expérience fun et intéressante.

Afin d'orienter mes tests, je devais décider du genre de jeu sur lequel je baserai mes niveaux d'expérimentation. J'ai donc choisi de créer des niveaux de type jeu de plateforme et d'action pour deux raisons :

Les plateformes permettent au joueur d'appréhender l'espace de son environnement non pas uniquement sur les axes x et y mais aussi sur l'axe z, c'est-à-dire en étant en hauteur.

Pour tester mes niveaux, j'ai besoin que les joueurs soient confrontés à une phase d'action dans laquelle ils devront affronter des ennemis. Par ailleurs, créer un niveau basé sur les jeux d'action est plus rapide que de devoir créer un niveau de jeu de rôle.

1ère méthode d'appréhension du joueur :

A partir des résultats obtenus grâce au questionnaire, j'avais différentes manières d'appréhender le comportement du joueur. Ce que j'avais fait inconsciemment au départ, c'est que je recréais une typologie de joueurs mais d'un point de vue du gameplay en anticipant et en orientant le joueur vers différentes façons de jouer même s'il pouvait les mélanger à la manière de Kingdom of Amalur Reckoning. J'avais repris les différentes actions que les joueurs m'avaient listées, j'en ai ajouté d'autres auxquelles je pensais et je les ai regroupées selon différentes catégories. Bien sûr, je me disais que le joueur pourrait mélanger ces différentes catégories de par ses choix d'action. Aussi j'ai regroupé ces actions sous les catégories :

Mobilité : qui correspond au déplacement du joueur dans l'espace et aux actions qui y sont attribuées.

Défense : qui comprend toutes les actions défensives et de support

L'attaque : qui regroupe tous types d'attaque possibles

De plus j'avais scindé ces catégories en sous-catégories, chacune comprenant les actions qui y correspondaient. Et enfin, sous chaque action étaient décrites ses différentes évolutions possibles en fonction de son utilisation par le joueur. Les actions en jaune correspondant aux actions que le joueur pourrait utiliser dès le départ dans le niveau. Voici un exemple avec les actions liées à la mobilité (l'intégralité du graphique est présent dans l'annexe) :

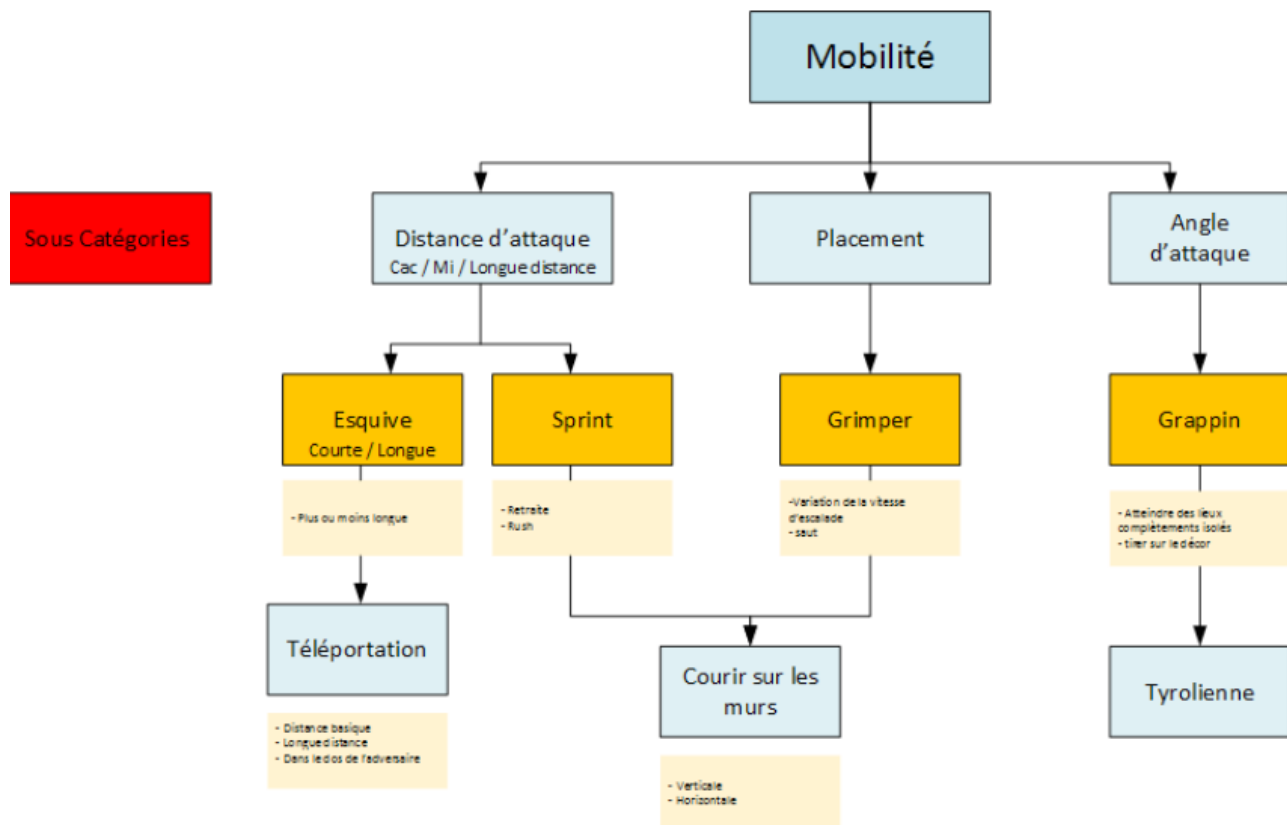


Figure 18 : Action et évolutions des actions liées à la mobilité du joueur

Même si avec ce genre de méthode le joueur pouvait mélanger les différentes catégories, il n'en restait pas moins que j'avais clairement orienté la manière de jouer de chaque catégorie. De plus, je me dirigeais vers une méthode dont j'avais précédemment décrit les limites. J'ai donc complètement revu ma méthode d'appréhension du joueur.

2nd méthode d'appréhension du joueur :

Ce que je souhaite à l'origine de ce projet, c'est que chaque joueur, individuellement, ait ses actions qui évoluent en fonction de leur utilisation. Même si tous les joueurs ont les mêmes actions en démarrant un niveau, les différentes utilisations de ces actions devraient les faire évoluer différemment. Je devais donc définir les actions que les joueurs auraient dès le commencement du niveau. L'objectif du joueur étant de vaincre des ennemis, ces actions doivent être suffisamment variées pour lui permettre d'aborder le combat de différentes manières afin d'évoluer différemment. Aussi, l'interaction du joueur en combat est régie par deux grandes variables :

Le déplacement dans l'espace : c'est-à-dire son placement au sol ou en hauteur, ainsi que la distance d'attaque avec l'ennemi visé (corps à corps, mi-distance, longue distance) et l'angle d'attaque (face à son adversaire, sur le côté ou dans le dos)

Le type d'attaque : est-ce une attaque physique avec une épée ou un arc ? Ou est-ce une attaque magique ? Si l'attaque est magique, quels sont les éléments utilisés ? Quels sont leurs effets ?

Enfin, chaque action programmée dans le jeu devrait posséder un poids. A chaque fois que le joueur utilise une action le poids de cette action doit se renforcer. De plus pour qu'une action soit débloquée par le joueur, le poids des actions nécessaires pour débloquer cette nouvelle action doit avoir dépassé

une certaine valeur. Par exemple si je détermine que pour pouvoir utiliser la compétence “courir sur les murs” le joueur doit avoir atteint le poids de 0,8 dans les compétences de “sprint” et de “grimper” ; tant que le poids de ces deux compétences n’aura pas dépassé 0,8 alors la compétence “courir sur les murs” ne sera pas disponible au joueur. J’ai donc conçu un second graphique en pensant aux relations entre chaque compétence et en déterminant sous quelles conditions ces compétences sont débloquentées par le joueur et les conditions d’activation de la compétence une fois disponible. Par exemple, la compétence “Téléportation” est disponible si la compétence “Esquive” a un poids supérieur à 0,7. Si tel est le cas alors la distance de téléportation dépend du type d’esquive et de sa longueur. Si un joueur utilise une esquive courte pour éviter un coup de l’ennemi, sa téléportation le téléportera dans le dos de son adversaire. A contrario, une esquive longue permettra au joueur de se téléporter sur de longues distances. Le graphique suivant dispose de flèches et de cases de différentes couleurs. Les cases et flèches vertes correspondent aux conditions nécessaires dans le jeu pour activer la compétence. Les cases bleues correspondent aux actions disponibles dès le commencement du niveau. Les cases oranges correspondent aux actions que le joueur peut débloquent. Les cases rouges correspondent à des attaques ultimes que le joueur peut débloquent. Les flèches oranges et rouges correspondent aux conditions nécessaires pour débloquent les différentes actions. Les flèches en pointillés correspondent à un moyen alternatif de débloquent une action.

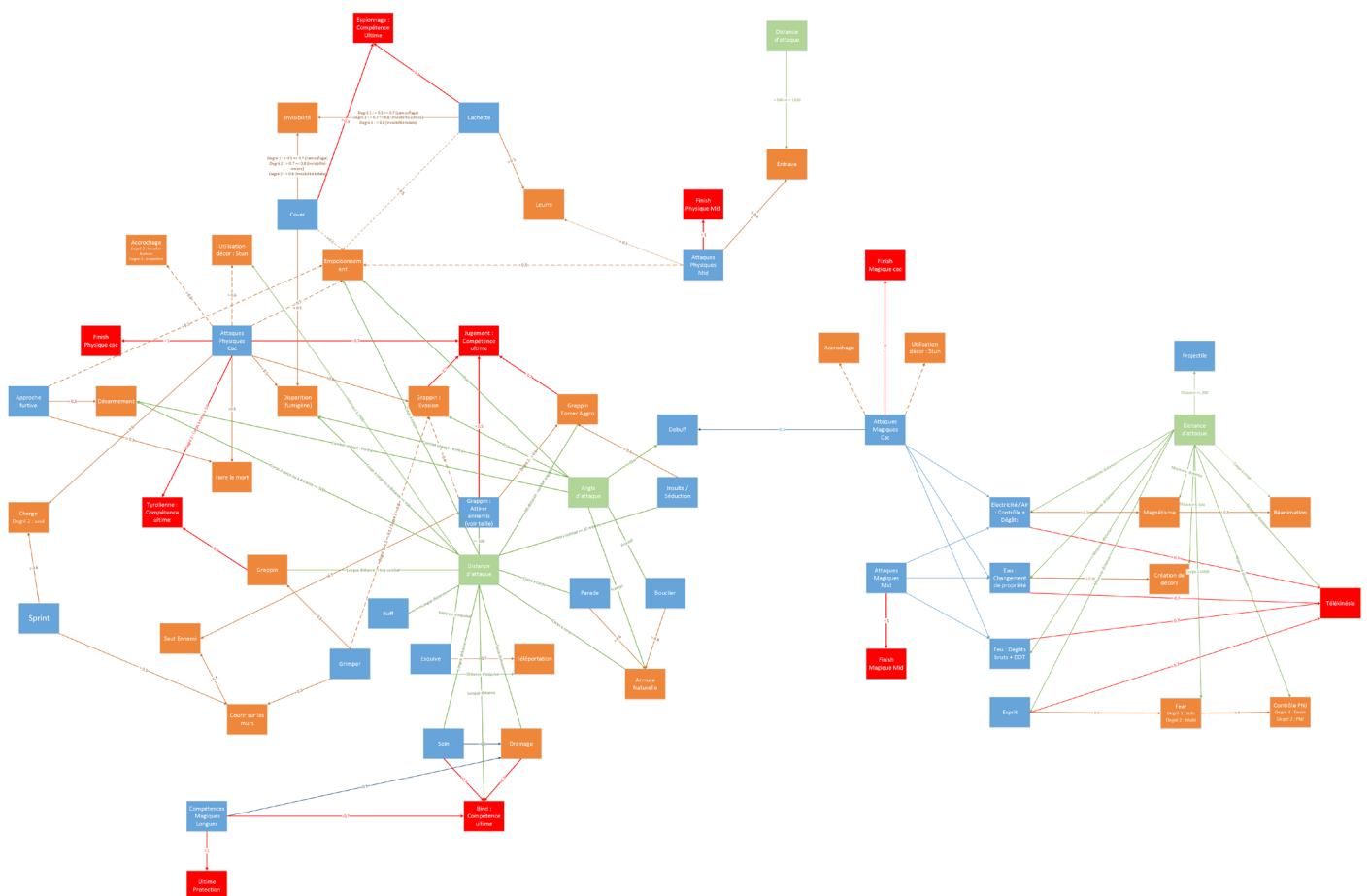


Figure 19 : Graphique montrant les liens entre chaque action, leurs évolutions, et les pré-requis pour les utiliser

L'inconvénient de ce graphique est qu'il est difficile à lire dans son ensemble. Il faut le lire compétence par compétences. Bien entendu, il pourrait comprendre bien plus de compétences que ce qu'il contient actuellement. Il représente simplement les idées que j'ai pu avoir en début de projet. De plus, je n'avais pas pour objectif de programmer toutes ces compétences à mon personnage, mais de limiter l'évolution des premières actions à quelques nouvelles. L'avantage que j'ai eu à le réaliser, c'est qu'il m'a fait prendre conscience que je devais définir les actions, leur évolution et leur disponibilité en fonction des relations et des liens qu'elles peuvent entretenir avec les autres. En un sens ce graphique me fait penser à un réseau plutôt qu'à un arbre de talents, même si les deux notions sont proches. Donner un poids aux actions m'oriente vers l'utilisation du renforcement que j'ai évoqué précédemment en terme associé à la psychologie et à l'intelligence artificielle.

II.1.2) Quantifier les actions du joueur

Le renforcement des actions utilisées par le joueur me paraissant intéressant à introduire dans mes tests, je me suis concentrée sur la programmation de cette technique. J'ai donc dû mettre en place une échelle où 1 représente le renforcement maximal de l'action et 0 représente une action non utilisée. Le juste milieu 0,5 représente l'action débloquée avec ses paramètres par défaut. Ainsi, toutes les actions disponibles dès le commencement du niveau de jeu ont un poids de 0,5. Cependant, une action qui ne sera jamais utilisée par un joueur ne doit pas lui rester disponible, car il n'en aura pas besoin. Il faut donc mettre en place un renforcement négatif des actions. C'est-à-dire programmer un timer pour chaque action qui, toutes les x secondes, diminue le poids d'une action non utilisée. Néanmoins lorsqu'une action est utilisée par le joueur, il faut réinitialiser le timer. Ainsi, les actions qui sont régulièrement utilisées par le joueur n'auront qu'un renforcement positif, leur poids augmentera et permettra de faire évoluer l'action ou d'en débloquent de nouvelles. A l'inverse, une action qui ne sera jamais ou peu utilisée par le joueur verra son poids diminuer, jusqu'à être indisponible à l'utilisation si le poids tombe à zéro. La subtilité de cette méthode vient du calibrage du renforcement : combien de points sont ajoutés en cas de renforcement positif ? Combien sont retirés en cas de renforcement négatif ? Et à quelle échéance le timer doit appliquer le renforcement négatif ? Quelle relation établir entre le nombre d'utilisations d'une action et ce qu'elle représente pour le joueur ? Il ne faut pas débloquent une nouvelle action trop rapidement, mais pour autant ne pas demander un trop grand investissement d'utilisation de cette action avant d'en débloquent une nouvelle.

Mon premier objectif a donc été de programmer ce renforcement en simplifiant au maximum son intégration dans le niveau. Ainsi, j'ai simplement réalisé un niveau test en vue à la première personne dans lequel le joueur peut réaliser différents types de déplacements et interagir avec des boîtes qui correspondent à une interaction avec un élément tiers comme un ennemi ou un élément du décor.

Ce premier test ne comprend que des actions basiques. Aussi j'ai programmé les actions :

- Sprinter (touche shift)
- S'accroupir (touche C)
- Esquiver (touche espace)
- Se déplacer (touches Z,Q,S,D)
- Interagir (touche E)

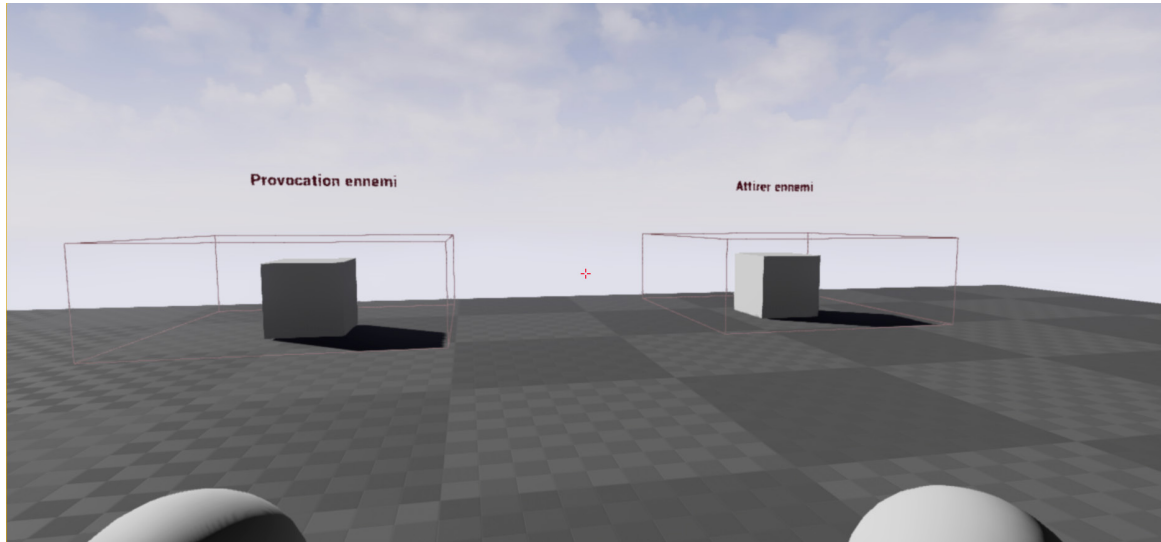


Figure 20 : Premier niveau me permettant de tester la méthode de renforcement

Lorsque le joueur réalise une action de déplacement ou d'interaction avec une boîte, il ajoute du poids à l'action qu'elle représente. Le poids par défaut est de 0,5. A contrario, une action non utilisée verra son poids diminuer toutes les 10 secondes.

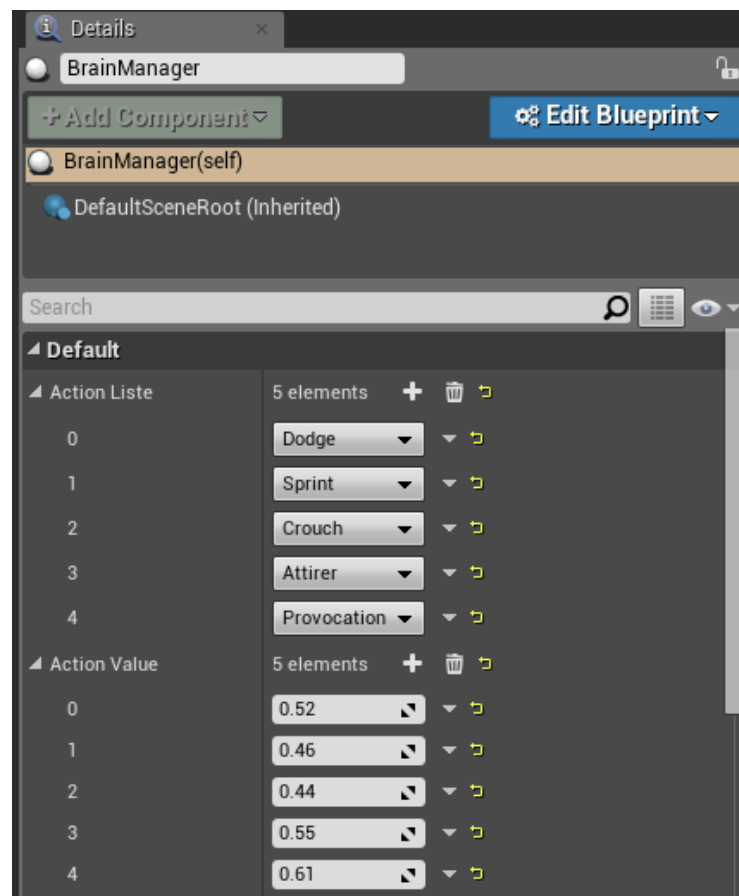


Figure 21 . Résultats de la méthode de renforcement

On voit ici clairement les actions qui ont été utilisées et ont subi un renforcement positif ainsi que celles que le joueur a délaissées et qui ont reçu un renforcement négatif. Cette méthode de renforcement a été programmée au travers de deux blueprints différents. Un blueprint équivaut à une classe dans le langage de programmation classique. Chaque blueprint correspond à une fonctionnalité du renforcement :

Le blueprint nommé “Brain Manager” recueille dans un tableau toutes les actions que le joueur peut réaliser (ces informations lui sont envoyées par le Brain Character). Il gère également le renforcement positif et négatif de chaque action.

Le blueprint nommé “Brain character” qui se charge d’initialiser toutes les actions existantes et d’envoyer les informations d’utilisation d’une action au Brain Manager afin d’entraîner un renforcement positif. Il fait le lien entre le personnage contrôlé par le joueur et le Brain Manager

Avant de programmer le renforcement, j’ai créé une liste de toutes les actions pouvant être utilisées dans ce test. Cette liste sera appelée dans les différents blueprints du renforcement.

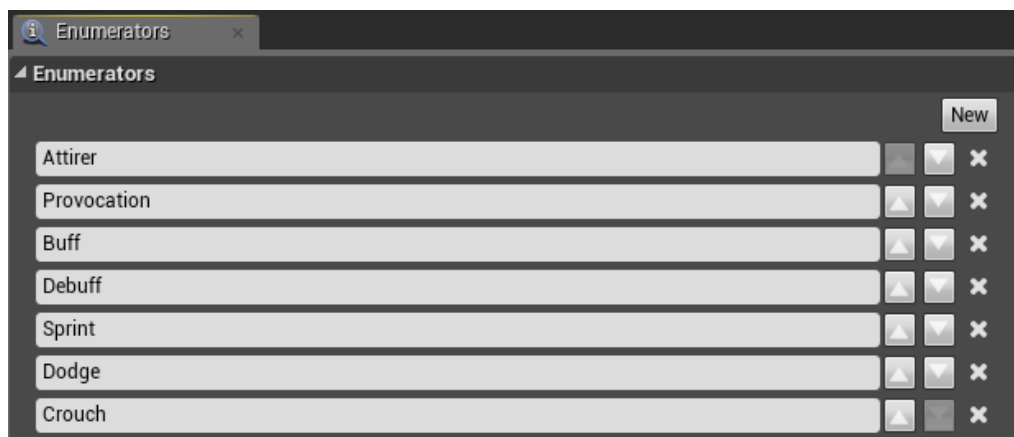


Figure 22 : Liste des actions qui seront renforcées

Fonctionnement du Brain Character

Le blueprint Brain Character est un blueprint de type Actor Component. C’est-à-dire qu’il peut être appelé sur un objet de type “Actor” ou “Character” comme le personnage d’un jeu. Au démarrage du niveau, il envoie le nom de chaque action présent dans la liste au Brain Manager ainsi que la valeur par défaut du renforcement (0,5) et la valeur par défaut du timer (0).

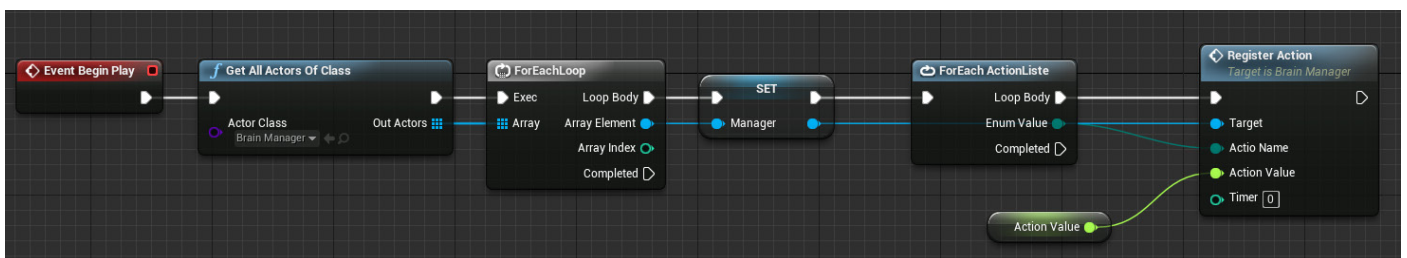


Figure 23 : Initialisation de chaque action

Lorsque le joueur utilise une action comme “esquiver”, le blueprint de son personnage fait appel au Brain Character pour lui signaler le nom de l’action utilisée. Le Brain Character envoie ensuite l’information au Brain Manager pour qu’il applique le comportement désiré.

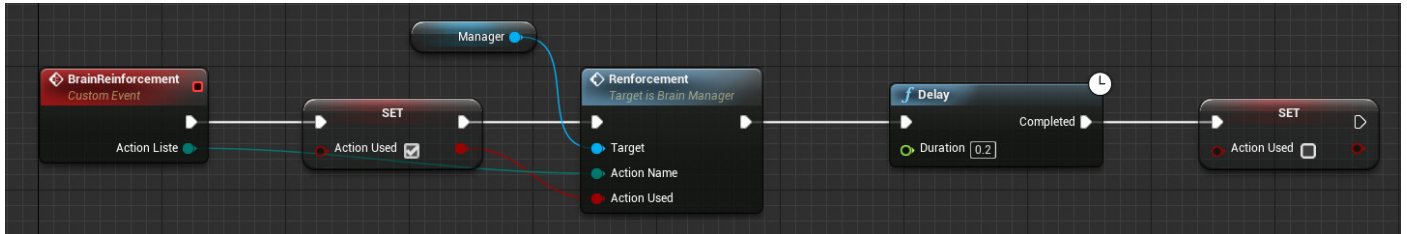


Figure 24 : Le Brain Character envoie au Brain Manager quelles actions sont utilisées

Fonctionnement du Brain Manager

Le Brain Manager est un blueprint de type Actor qui est placé directement dans la scène du niveau. Lorsqu’il reçoit les informations du Brain Character au démarrage du niveau, il les mémorise dans trois tableaux regroupant le nom, le poids de renforcement et le timer de chaque action.

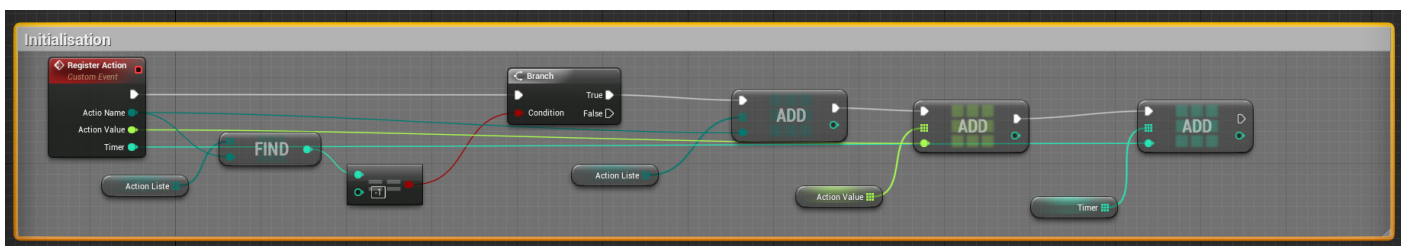


Figure 25 : Le Brain Manager initialise dans plusieurs tableau les différentes informations de chaque action : le nom, le poids de renforcement et la valeur du timer.

Lorsque le Brain Character lui envoie le nom de l’action qui a été utilisée par le joueur, le Brain manager récupère la ligne de chaque tableau correspondant au nom de cette action et y applique ces deux comportements :

Il récupère la valeur du poids de renforcement, applique un renforcement positif en ajoutant la valeur x au poids existant. Puis mémorise cette nouvelle valeur dans la ligne du tableau correspondant. Si le poids récupéré est déjà de 1 alors la valeur ne change pas, car 1 est le maximum.

Il réinitialise le timer de l’action utilisée afin qu’elle ne subisse pas de renforcement négatif.

Enfin toutes actions n’étant pas utilisées dans le temps imparti du timer voit son poids de renforcement récupéré par le Brain Manager et diminué une fois de valeur x . Le timer étant récursif, si une action n’est pas utilisée à répétition, son poids descendra jusqu’à 0. Si le poids d’une action est déjà à 0 lorsque le timer se termine, alors aucun renforcement négatif ne s’opère car 0 est le minimum. Une action qui voit son poids diminuer à 0 devient indisponible pour le joueur.

Cette méthode simple permet de récupérer en temps réel le poids de chaque action utilisée. Elle est également évolutive. En effet, d’une part elle permet aux concepteurs qui veulent rajouter des actions à leur personnage ou aux éléments tiers d’avoir ces actions automatiquement référencées dans le blue-

print qui calcule le renforcement. D'autre part, les variables permettant de gérer le poids du renforcement positif et négatif ainsi que le timer sont très simples à modifier en cas de réglages et peuvent être changées sans obligatoirement passer par le blueprint (variable publiques). Enfin, le poids obtenu à chaque utilisation est très simple à récupérer pour appliquer les règles de comportement d'évolution des compétences ou du décor.

II.1.3) Conception des règles d'évolution des actions du joueur

Au second semestre, j'ai porté mon attention sur la création de règles qui régissent l'évolution des compétences suivant le poids de renforcement des actions utilisées par le joueur. En effet, nous avons vu en premier lieu comment nous pouvions quantifier l'utilisation de chaque action. Cependant, ces données ne peuvent être utilisées seules et doivent permettre la création de règles déterminant lorsqu'une nouvelle action est débloquée par le joueur et le degré d'utilisation. Par exemple, la capacité d'invisibilité peut être débloquée selon trois degrés différents en fonction du poids de renforcement des actions requises :

- Degré 1 : le personnage peut se camoufler lorsqu'il reste immobile
- Degré 2 : le personnage peut être invisible dans les zones d'ombre et se déplacer durant son invisibilité. Nécessite d'être hors combat.
- Degré 3 : le personnage peut être invisible dans n'importe quel espace et se déplacer sous invisibilité mais uniquement lorsqu'il est hors combat.

Les règles d'évolution des actions sont regroupées dans deux documents présents dans l'annexe de ce mémoire. Le premier correspond au graphique présent dans la partie "Appréhender le joueur" de ce mémoire, le second est un document texte relatant toutes les conditions nécessaires à la disponibilité et au fonctionnement d'une action. Il s'agit de la version écrite du graphique qui comprend tous les détails de chaque action. De plus, il décrit le système de gameplay de la seconde expérience que j'ai réalisée. Dans ce nouveau test, j'ai intégré la méthode de renforcement et à laquelle j'ai ajouté les règles d'évolution des actions.

Ce second niveau de jeu est une arène dans laquelle le joueur contrôle un personnage à la troisième personne et doit affronter plusieurs adversaires. A contrario du premier test, j'ai donc importé un personnage, des animations, réalisé un décor, programmé une intelligence artificielle basique pour les ennemis, ainsi que différentes actions pour le personnage du joueur. Ce niveau est scindé en deux parties : la première permet de quantifier les actions du joueur et y applique les règles faisant évoluer ses actions. La seconde modifie l'agencement du décor du niveau et sera abordées une prochaine partie de ce mémoire.



Figure 26 : Différents lieux du niveau d'arène

Bien que j'ai voulu programmer un maximum d'actions avec un maximum d'évolutions possibles, je n'avais pas suffisamment de temps pour programmer toutes les actions décrites dans les documents. De plus, certaines évolutions ne possèdent pas tous les degrés d'activation que je souhaitais programmer. C'est le cas de l'invisibilité. Si le renforcement des actions "se mettre à couvert" et "s'accroupir" est suffisant le joueur peut débloquent l'invisibilité. Cependant, il n'y a que l'invisibilité totale (degré 3) qui a été programmée. A l'inverse, on a bien les deux types de téléportation programmés : la téléportation dans le dos de l'adversaire si le joueur utilise principalement les esquives à courte distance et la téléportation à longue distance si le joueur utilise les esquives longues. Dans ce niveau, le type d'attaque du joueur dépend de sa distance et de son orientation avec la cible (dans le dos, sur les côtés, face à l'adversaire). Lorsqu'un joueur veut attaquer un adversaire, il va se positionner par rapport à lui en fonction de l'attaque qu'il va vouloir réaliser. Ainsi, les joueurs qui aiment jouer la carte de la furtivité vont préférer approcher leur adversaire discrètement et attaquer dans le dos. C'est pourquoi il me semblait pertinent de programmer différents types d'attaques selon le placement du joueur par rapport à l'adversaire. Voici donc les différentes actions programmées avec une ou plusieurs évolutions dans le niveau.

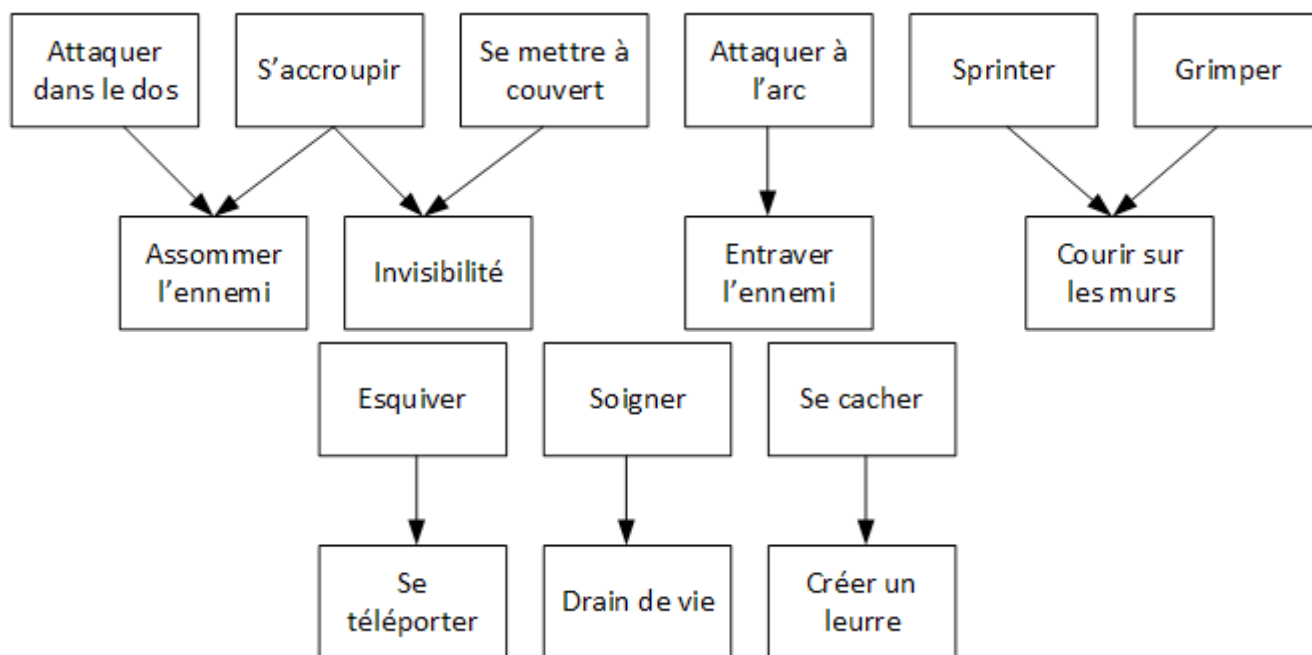


Figure 27 : Actions programmées dans le niveau d'arène et leurs évolutions

La problématique qui se posait alors était de savoir comment déterminer, mettre en place et appliquer les règles d'évolution des actions du joueur. Au départ, je me suis intéressée à la méthode de la "Fuzzy Logic" ou logique floue en intelligence artificielle. Cette méthode permet de déterminer toutes les variations d'une situation et d'y appliquer un comportement. Pour reprendre l'explication donnée dans le livre "Artificial intelligence for games" de Ian Millington et John Funge, on peut imaginer un ennemi qui tire sur le joueur lorsque celui-ci est à découvert et qui arrête de tirer lorsqu'il est caché derrière un décor. La question que l'on peut se poser est : que ce passe-t-il si le joueur est à moitié à découvert ? Déterminer des comportements en fonction d'une situation prédéterminée comme joueur à couvert et joueur à découvert peut se programmer dans un arbre de comportement (aussi appelé Behavior tree). Mais le problème de cette méthode réside dans les différentes variations du comportement du joueur : être en partie caché. La fuzzy logic permet de déterminer un comportement en fonction de ces différentes variations de situation. Elle fonctionne selon trois étapes :

- D'abord, elle prend en entrée d'input les variables obtenues dans le jeu et les transmet au moteur d'inférence. C'est l'étape de fuzzification. Dans mon cas, il s'agirait du poids de renforcement de chaque action.

- Puis elle applique les règles qui lui sont programmées dans le moteur d'inférence. Ces règles sont sous la forme de : Si "condition 1"... et... "condition 2"... alors... "résultat".

- Enfin, les résultats sortent en output du moteur d'inférence. C'est l'étape de défuzzification. Les résultats de la fuzzy logic peuvent être montrés sous la forme d'un graphique aussi nommé "Crisp". Ainsi, si je reprends mon exemple de joueur qui se cache en fonction des paramètres d'entrée (en input) et des règles on peut obtenir ce genre de graphique.

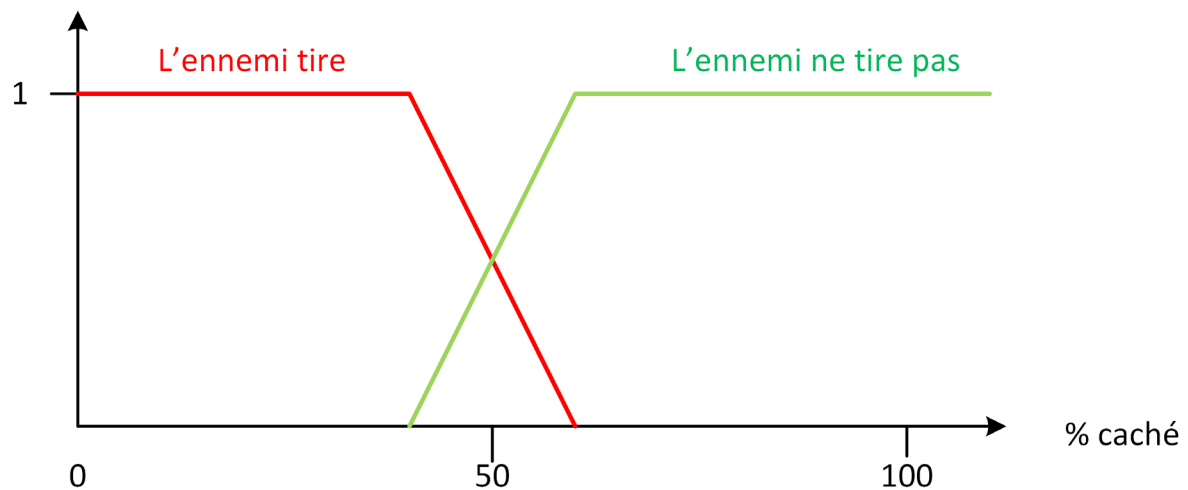


Figure 28 : Exemple de «Crisp» de fuzzy logic

On voit donc que si le degré de couverture du joueur va de 0% à 40% alors l'ennemi tirera en continu sur le joueur. Cependant, à partir de 40% la courbe "ne tire pas" augmente. On peut donc imaginer qu'à partir de 40% et jusqu'à 60% l'ennemi tirera sur le joueur en alternant avec l'action de se mettre lui-même à couvert. Cette dernière sera plus fréquente que le tir si l'on se rapproche de 60%. Enfin les 60% dépassés, l'ennemi n'attaquera plus le joueur et décidera d'une autre action (selon ce qui lui est programmé) comme se déplacer ou se cacher lui aussi.

Cette méthode me semblait intéressante pour déterminer à quel moment chacune de mes actions pourraient devenir disponibles. Ainsi je me suis intéressée à la bibliothèque "Fuzzy lite" qui a été programmée par Juan Rada-Vilela et qui est libre pour toute utilisation personnelle et à but non commerciale. Cette bibliothèque permet d'utiliser directement tous les outils de fuzzy logic sans que je n'ai besoin de les programmer moi-même. J'ai donc à partir de cette bibliothèque, créé quelques règles de fuzzy logic que je souhaitais intégrer à mon test.

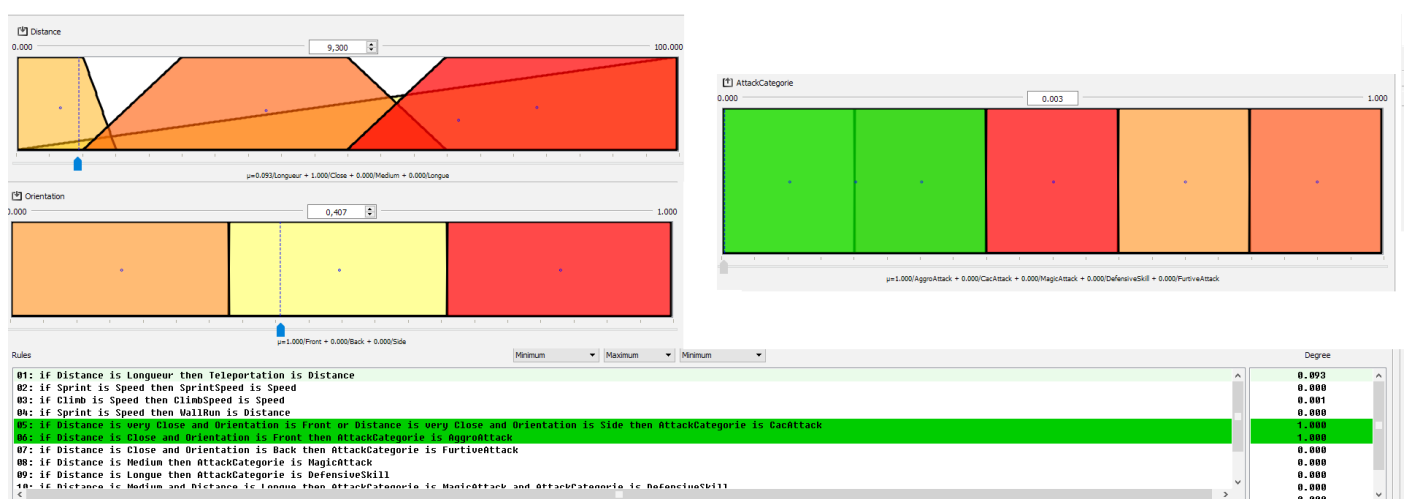


Figure 29 : Exemple de «Crisp» que j'ai réalisés pour essayer la méthode de fuzzy logic. Ici on détermine le type d'attaque du personnage joueur en fonction de la distance et de l'orientation du personnage par rapport à l'ennemi.

Cependant, un problème est rapidement apparu. Les variables que j'entrais en input me semblaient trop résolues, finies. Et mes règles pouvaient se résoudre à :

Si... “poids de l'action” $\geq$ x alors “action X disponible”.
ou
Si... “poids de l'action 1” $\geq$ x et “poids de l'action 2” $\geq$ x alors “action X disponible”.

Où x est le poids de renforcement souhaité pour débloquer une nouvelle action et X est le nom de l'action à débloquer.

Ce qui revient à prédéfinir une situation (comme pour l'arbre de comportement). En réalité bien que je comprenais l'utilité de la fuzzy logic, je n'arrivais pas à l'intégrer correctement à mon test. De plus le temps que je pouvais allouer à programmer cette méthode était relativement court, car je devais ensuite me concentrer sur la génération de l'aménagement du décor. J'ai donc décidé d'abandonner la fuzzy logic et de créer les règles basiquement selon les exemples donnés précédemment. Le nombre de règles pouvant être plus ou moins important en fonction du nombre d'actions et d'évolutions, j'ai décidé de les intégrer en langage C++ dans Unreal Engine 4.

Mes règles d'évolution d'action fonctionnent grâce à trois codes C++ :

Le premier récupère le poids de renforcement de chaque action du Brain Manager en temps réel et crée une variable associée à chacune de ces actions dans laquelle il mémorise la valeur du poids de renforcement. Il initialise également une variable booléenne pour chaque action qui peut être débloquée. Par défaut ces variables sont initialisées à “faux”. Ces variables booléennes peuvent être récupérées dans le Brain Manager.

Le second comprend toutes les règles requises pour débloquer de nouvelles actions. Comme, par exemple :

```
bool entrave = false;

bool Entrave() {
    if (bowAttack > 0.8) {
        entrave = true;
        return entrave;
    }
    else {
        entrave = false;
        return entrave;
    }
}
```

Le booléen “entrave” est initialisé à “faux”. Si le poids de “bowAttack” est supérieur à 0,8 alors la variable booléenne va se modifier en “vrai” et l'action “Entrave” sera débloquée pour le joueur.

Le troisième code fait le lien entre les deux premiers. C'est-à-dire qu'il va récupérer la variable d'action créée dans le premier code et l'envoyer vers le second code en lui disant de vérifier la condition qui y

est présente. Puis il récupère le résultat en sortie du second code et modifie la variable booléenne de premier code en fonction du résultat.

Pour illustrer par un exemple, prenons le poids de renforcement de l'attaque à l'arc qui est, pour l'exemple, de 0,85. Le Brain Manager envoie cette information au premier code C++ qui va mémoriser cette valeur dans une variable "bowAttack". Grâce au troisième code C++ cette variable va exécuter la fonction Entrave() qui lui est associée dans le second code C++. Le minimum pour débloquent l'action "Entrave" étant de 0,8, la variable booléenne "entrave" va passer de "faux" à "vrai" et va être transmise au premier code C++. De ce code le Brain Manager pourra récupérer l'information de sortie et transmettre au code blueprint du personnage qu'il peut utiliser l'action "Entrave".

L'avantage de cette méthode, c'est qu'elle est très rapide et facile à mettre en place. Il faut créer autant de règles dans le code C++ qu'il y a d'actions à débloquent. L'inconvénient majeur de cette méthode est qu'elle n'est pas très évolutive et adaptative aux différentes situations que peuvent rencontrer les joueurs. Contrairement à la fuzzy logic qui prend toutes les différentes variations d'une situation.

II.1.4) Génération du décor d'un niveau de jeu vidéo

Cette expérimentation fait suite à la création des règles permettant l'évolution des actions et se déroule dans le même niveau. Il s'agit maintenant d'agencer les différents éléments du décor du niveau pour qu'ils s'adaptent au comportement du joueur. Pour quantifier les actions du joueur dans ce niveau, il faut au préalable créer un environnement standard permettant au joueur différentes approches des situations de combat auxquelles il va être confronté. Aussi je distingue deux types de décor :

Le décor fixe qui ne sera pas modifié par algorithme. Il s'agit surtout de la forme du niveau en elle-même, les murs et dans le cas présent la structure au centre de l'arène.

Le décor modulable qui comprend tous les éléments avec lesquels le joueur va pouvoir interagir directement comme les cover, les plateformes, les cachettes, les éléments magiques, etc. L'agencement de ces éléments dans le décor est généré par un algorithme.

Je tiens à préciser que la génération procédurale dans ce niveau ne permet que de gérer les différents emplacements d'objets. En début d'année de ce master, lorsque je voulais générer un décor en fonction du comportement du joueur, je voulais en réalité concevoir moi-même l'intégralité de ce décor et en fonction des actions quantifiées, cacher ou dévoiler des objets de cet environnement. Mais dans ce cas, l'intégralité des objets de l'environnement était déjà placée dans le décor. Cependant, en faisant évoluer mes expérimentations, je trouvais plus intéressant de travailler sur un algorithme qui supprimerait, ajouterait et agencerait lui-même des objets dans l'environnement. Malheureusement, je n'ai pu consacrer que deux semaines à l'élaboration de cette génération d'environnement. Aussi je n'ai pas pu développer cet algorithme au maximum.

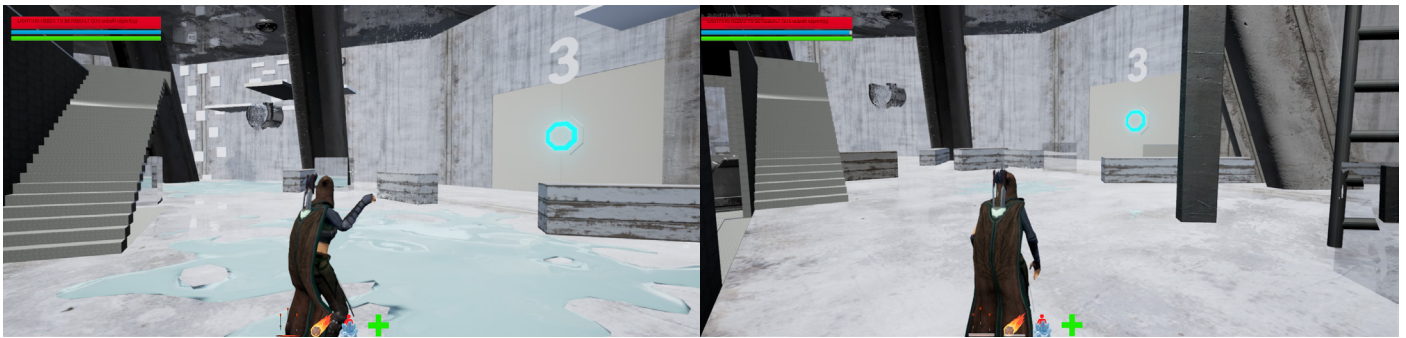
Le niveau se déroule selon ce schéma :

- Le niveau démarre, le joueur prend possession de son personnage dans un environnement que j'ai conçu au préalable et doit vaincre ses ennemis. C'est la phase première du niveau.
- Les ennemis sont vaincus, le jeu génère un nouveau décor et fait apparaître de nouveaux en-

nemis.

- Le joueur doit vaincre ces nouveaux assaillants. C'est la seconde phase du niveau.

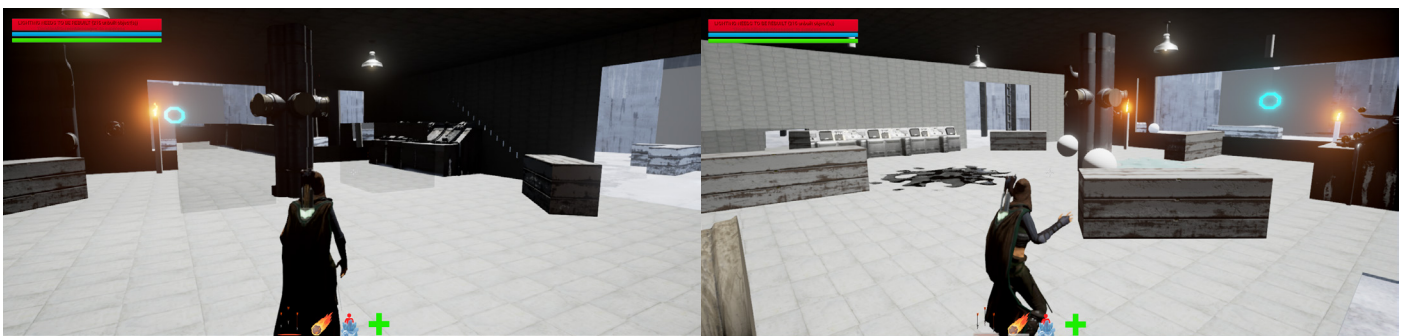
La mort de tous les ennemis du niveau déclenche la génération du décor et permet d'enchaîner sur une seconde phase d'affrontement. Lorsque le jeu génère un nouveau décor, il supprime en premier tous les objets modulables que j'ai placés auparavant, avant d'ajouter de nouveaux objets.



Avant génération

Après génération

Figure 30 : Dans ce cas présent, je n'ai pas utilisé de magie pour vaincre mes adversaires, mais beaucoup utilisé de cover en attaquant par surprise



Avant génération

Après génération

Figure 31 : Ici, j'ai joué en utilisant la magie, les poisons et les cover. En revanche je n'ai pas utilisé de «cachette» (cube transparent) dont on constate la disparition.

Comment fonctionne cette génération de décors ?

Pour générer le décor, l'algorithme se base sur le poids de chaque action récupérée à l'instant où tous les ennemis du joueur sont morts. La génération se base donc sur les données obtenues par la méthode de renforcement. Générer l'agencement des objets répond à différentes règles :

- Le nombre d'objets apparaissant doit dépendre du poids de renforcement de chaque action. Par exemple si le poids de renforcement de l'action "se mettre à couvert" est nul, aucun cover ne doit apparaître sur le terrain, car le joueur ne les utilisera pas.

- Les emplacements d'apparition des objets doivent être déterminés en fonction des différentes zones du terrain et de la pertinence avec sa topographie. De plus, il serait intéressant de connaître dans quelles zones du terrain le joueur préfère jouer afin d'orienter la génération d'objets vers ces zones (sans pour autant y générer tous les objets).

- Les éléments du décor doivent pouvoir se combiner. Par exemple un joueur qui aime jouer

furtivement et empoisonner ses cibles, doit pouvoir récolter du poison en étant à couvert ou caché.

- Il faut éviter la redondance des objets. Deux cachettes l'une à côté de l'autre sont inutiles.

Le fonctionnement des blueprints permettant la génération du décor fonctionne similairement à la méthode de renforcement. Ainsi nous avons :

- Un blueprint nommé "Terrain Manager" qui, tout comme le "Brain Manager", comprend toutes les règles de fonctionnement de l'algorithme. Il recueille dans différents tableaux les informations envoyées par les objets du décor qui serviront à agencer le nouvel environnement.

- Trois blueprints nommés respectivement "Ennemi Brain", "Target Brain" et "Terrain Brain", qui ont le même rôle que le "Brain Character" de la méthode de renforcement. C'est-à-dire qu'ils envoient au "Terrain Manager" toutes les informations concernant les objets qu'ils gèrent.

Ainsi le blueprint "Ennemi Brain" initialise chaque ennemi présent dans le niveau, dans le tableau des ennemis du "Terrain Manager" et met à jour ce tableau lorsqu'un ennemi est tué. Ainsi, en temps réel, le blueprint "Terrain Manager" vérifie en continu si tous les ennemis du niveau sont morts. Si c'est le cas, il lance la génération de décor. Sinon il continue de vérifier la présence des ennemis.

Le blueprint "Terrain Brain" est utilisé au lancement du niveau par tous les objets que j'ai moi même placés pour constituer la première phase d'arène. Il initialise dans le tableau des objets du "Terrain Manager" tous les objets existants et qui devront être supprimés à la fin de la première phase pour permettre la génération de nouveaux objets. Lorsque les objets sont supprimés du tableau, je supprime également les lignes du tableau afin de libérer la mémoire allouée.

Enfin j'ai choisi de gérer l'emplacement des objets qui apparaissent en plaçant dans mon environnement un certain nombre de "target". Lorsque l'algorithme fait apparaître un nouvel objet dans l'environnement il le fera apparaître à l'emplacement d'une target. C'est pourquoi le blueprint "Target Brain" permet de référencer dans le tableau des targets du "Terrain Manager" toutes les informations relatives à chaque target. Ces informations permettront de localiser l'emplacement d'apparition de l'objet, de détecter si des objets peuvent se combiner et d'éviter leur redondance. Il comprend toutes les informations nécessaires à la génération du décor. Aussi, les informations mémorisées dans le tableau des targets du Terrain Manager sont :

- Le nom de la target
- Le type de la target : une target située en l'air contre un mur ne doit pas faire apparaître de cover. Cela paraîtrait illogique. En revanche, elle peut permettre l'apparition de plateformes ou d'accroches permettant au joueur de grimper.
- Si la target est utilisée ou non. Cela évite de faire apparaître deux objets au même endroit, ou encore, de détecter si on peut combiner différents objets.
- La localisation de la target. En effet l'environnement est divisé en différentes parties. Il y a un décor dédié à l'élément du feu, un autre pour l'eau, le joueur peut également entrer dans la salle des machines située au centre de l'arène ou encore monter sur son toit. Ainsi on peut diriger les apparitions d'objets en fonction du type d'environnement choisi.
- Le nom de la target avec laquelle elle peut se combiner

Enfin chaque target placée dans la scène est issue d'un blueprint (de type target point) qui gère la rota-

tion et la taille de l'objet qui apparaît à son emplacement. En effet lorsque le "Terrain Manager" génère un objet à l'emplacement d'une target, il envoie au blueprint de la target la référence de l'objet qu'il vient de faire apparaître. Le blueprint va alors modifier la rotation, la taille de l'objet, ou les deux.

Exemple de génération d'objet

Prenons pour exemple la génération de cover. Lorsque les ennemis du niveau sont morts le blueprint "Terrain Manager" récupère le poids de renforcement de chaque action au blueprint "Brain Manager" et les mémorise dans de nouvelles variables. Pour générer mes cover, j'ai associé au poids de renforcement de l'action "se mettre à couvert" un pourcentage d'utilisation de l'action. Par exemple, un renforcement de l'action, situé entre 0,05 et 0,2 pourrait équivaloir à 20% d'utilisation de l'action. De ce fait, il doit y avoir environ 20% des targets qui doivent faire apparaître des cover (target dont le type y est associé). En plus des informations envoyées par le "Target Brain" dans le tableau des targets du "Terrain Manager", j'ai initialisé une colonne correspondant à une valeur choisie aléatoirement par l'algorithme pour chaque target. Si cette valeur est en dessous de 20, alors un cover apparaît à l'emplacement de la target. Et ce, pour chaque tranche de renforcement (de 0,05 à 0,2 non compris; de 0,2 à 0,5 non compris ; de 0,5 à 0,7 non compris et tout renforcement supérieur ou égal à 0,7).

Certaines apparitions d'objets sont soumises à des priorités. C'est le cas des poisons et des cachettes qui n'apparaissent qu'une fois que tous les cover ont été placés, car leur emplacement va dépendre de l'emplacement des cover. De même si l'algorithme détecte qu'un objet est apparu à l'emplacement d'une target pouvant se combiner avec une autre alors la probabilité d'apparition à l'emplacement de la target combinée est accrue.

Dans certains cas, un objet apparaîtra obligatoirement si un premier est déjà positionné à l'emplacement de sa target combinée. C'est le cas des plateformes isolées de tout mur où le joueur peut grimper. Dans ce cas une attache permettant au joueur de lancer son grappin apparaîtra obligatoirement à l'emplacement de la target.

Bilan de cette expérimentation :

Dans l'ensemble l'environnement change et l'algorithme semble bien fonctionner. Cependant, je n'obtiens pas autant de variété de décor que ce que j'espérais. Notamment en ce qui concerne le parcours aérien qui reste assez similaire même si les accès aux plateformes sont différents et que les interactions avec les éléments destructibles de l'environnement changent. Cela peut être dû au système de target pour définir les emplacements des différents objets. Délimiter des zones favorables à leur emplacement aurait pu permettre d'avoir un environnement plus varié, mais pose la question de l'agencement des objets entre eux. Notamment pour éviter que des objets se superposent. De plus, cette méthode demande énormément de temps pour tester la cohérence des apparitions d'objets. Que cela soit pour chaque localisation, taille et rotation d'objets mais également pour le rapport entre la fréquence d'apparition des objets et le renforcement des actions du joueur. Enfin ce que j'aurai souhaité ajouter dans cette expérimentation c'est de faire apparaître les objets en fonction des zones où le joueur a le plus joué. L'inconvénient que l'on pourrait voir à cela, est que rien n'affirme qu'un joueur ayant joué dans certaines zones durant la première phase du niveau continuera de jouer dans ces zones pendant la seconde phase. Et donc que cela pourrait entacher l'expérience du joueur dans le niveau de trop favoriser

certaines zones. Surtout si cela implique de constater son comportement sur seulement deux phases. Si le joueur continuait de jouer dans les mêmes zones à répétition durant un plus grand laps de temps alors il serait probablement intéressant d'aménager l'espace en fonction de ses zones de prédilection.

II.2) Transcendance : expérimentation en groupe

Transcendance (2017) réalisé par Suzanne Tosun et Jingjie Yu

Ce projet a été réalisé en groupe sur trois semaines. Pour cela je me suis associée avec Jingjie, une seconde élève de ma promotion. Nous devons combiner nos sujets de mémoire respectifs afin de concevoir une expérience. C'était pour nous l'occasion d'intégrer dans un projet réel nos expérimentations du premier semestre. A savoir, pour Jingjie, faire de la reconnaissance vocale via un moteur de jeu, et pour moi, implémenter ma technique de renforcement dans un niveau de jeu vidéo. Nous avons en commun l'envie de travailler avec Unreal Engine 4 car nous avons déjà réalisé des expérimentations auparavant et que nos deux mémoires reposaient sur l'utilisation de ce logiciel.

Nos objectifs :

Même si nos deux sujets de mémoire étaient très différents nous avons tout de même remarqué que l'utilisation de la reconnaissance vocale pouvait découler des résultats obtenus par la technique de renforcement. Nous avons donc pensé un niveau où le joueur pourrait faire apparaître des éléments du décor (pour combattre ou résoudre des énigmes) à un endroit ciblé en parlant. Cependant ces objets devraient coïncider avec le profil comportemental du joueur. Par exemple, un joueur qui se cache beaucoup doit pouvoir faire apparaître des endroits où se cacher. A la différence d'un joueur qui aime grimper, lui préférera faire apparaître des plateformes en hauteur, des attaches pour lancer son grappin, etc. Nous voulions concevoir un niveau où le joueur serait libre d'utiliser les compétences de son personnage qui conviendraient le mieux à sa façon de jouer. Le joueur évoluerait dans un niveau aux multiples possibilités où une situation aurait en réalité de nombreuses manières d'être résolue. La reconnaissance vocale ajoutant à cela la possibilité d'agencer le décor comme le souhaiterait le joueur et ainsi, en quelque sorte, créer ou modéliser une zone du niveau à sa convenance (créer sa zone de confort dans l'idée que le joueur invente sa propre résolution à une situation donnée que ce soit une situation d'action ou de résolution d'énigme).

Nous souhaitions également programmer un grand nombre de compétences au personnage incarné par le joueur. Cela afin d'augmenter les possibilités d'action et surtout les possibilités d'évolution des compétences du personnage. Le joueur pourrait alors débloquent de nouvelles compétences en fonction de celles précédemment renforcées. Malheureusement dans le temps imparti nous n'avons pu intégrer qu'un nombre très limité de compétences. De même, si le renforcement de certaines compétences permettait, dans une version antérieure du niveau, de débloquent de nouvelles compétences et même de générer du décor permettant leur utilisation ; quelques bugs tardifs dans les versions plus récentes du jeu et ne pouvant être résolus à ce niveau d'avancée dans le projet m'ont forcé à retirer ces fonctionnalités.

Mon rôle principal dans ce projet a été de modéliser et programmer le niveau ainsi que les intelligences artificielles des ennemis et également d'adapter la technique de renforcement au jeu. J'ai également travaillé sur l'aspect narratif et intégré des sons, dialogues et cinématiques au niveau.



Figure 32 : Boss de fin de niveau que le joueur doit affronter. La manière de vaincre le boss diffère selon le résultat du renforcement.

Ce que nous avons réalisé :

Dans la version finale du niveau le joueur incarne June dotée de diverses compétences permettant au joueur d'aborder les situations sous différents angles. Au fil du niveau celui-ci doit affronter différents ennemis chacun ayant leurs spécificités : certains pouvant être vaincus en étant attaqués de manière directe, d'autres ne pouvant l'être sans s'aider du décor. Ainsi le joueur pourra escalader des murs pour atteindre des zones en hauteur, faire tomber des éléments du décor sur ses ennemis, attaquer au corps à corps ou en tirant des flèches, lancer des boules de feu ou d'eau, geler une surface au sol et empêcher ses ennemis de bouger, esquiver ou encore se hisser vers un lieu ciblé grâce à son grappin. Toutes ces compétences étant disponibles pour permettre au joueur de créer sa manière de jouer, sa propre stratégie, s'approchant au plus près de son profil comportemental.

Outre l'aspect technique du projet qui me paraissait plutôt ambitieux pour une création de trois semaines, le second défi était de concevoir un niveau progressif où le joueur aurait le temps de prendre en main les différentes compétences de son personnage dans un temps et un espace très limité. Tout cela avant d'être jeté dans un environnement lui demandant de réagir à une situation plus stimulante avec un rythme plus rapide. Nous avons donc divisé ce niveau en trois parties :

- La première dans laquelle le joueur suit un tutoriel lui permettant d'appréhender les différentes compétences de son personnage. Il rencontre également les deux types d'ennemis qu'il sera amené à combattre dans la salle suivante. Le joueur n'est pas mis dans une situation de danger imminent. Pour autant, nous lui montrons comment venir à bout de ses adversaires de différentes manières.

- La seconde partie correspond à une salle dans laquelle le joueur doit affronter plusieurs adversaires en utilisant les compétences qu'il désire. Plus ces compétences sont utilisées et plus elles sont renforcées. Celles que le joueur n'utilise pas ont un renforcement négatif, c'est-à-dire qu'on va progressivement diminuer le niveau de renforcement jusqu'à zéro suivant un timer.

- La troisième partie, où le joueur doit affronter le boss du niveau. En fonction du renforcement précédemment calculé, la méthode pour vaincre ce boss est différente. Nous avons programmé deux

méthodes : si le joueur a beaucoup grimpé dans la salle précédente, il devra effectuer un parcours “aérien” pour vaincre le boss (grimper, utiliser le grappin pour atteindre des endroits escarpés, faire apparaître des plateformes, etc). Dans le cas contraire, il devra trouver un moyen de détruire le boss en restant au sol. Dans cette partie le joueur doit utiliser la reconnaissance vocale pour parvenir à ses fins. Les mots à prononcer sont différents selon la méthode utilisée pour détruire le boss. Chaque méthode demandant de faire apparaître des objets bien spécifiques. Pour autant il est possible de faire apparaître n’importe quel objet à n’importe quel emplacement, même si celui-ci ne permet pas de résoudre l’énigme correctement.



Figure 33 : Différents lieux du niveau. Respectivement : la salle d'entraînement, le premier affrontement et la salle du boss

Pour ce qui est des ennemis, il nous fallait programmer des intelligences artificielles simples mais attaquant le joueur si celui-ci est détecté. Pour cela j’ai utilisé un Behavior tree et un Navmesh. Les ennemis (boss compris) ont trois types de comportements :

Le joueur n’est pas détecté : l’ennemi se déplace à un endroit aléatoire dans la salle.

Le joueur est détecté : si l’ennemi est un mannequin celui-ci suit le joueur jusqu’à être suffisamment proche pour lui infliger des dégâts. Si l’ennemi est le boss et qu’il est à une distance inférieure à 500 alors, il suit le joueur et l’attaque au corps à corps. Si la distance est supérieure à 500 alors il jette des rochers sur le joueur.

L’ennemi est en contact avec une surface gelée : si l’ennemi est un mannequin alors il est gelé et ne peut plus bouger. Si l’ennemi est le boss alors celui-ci fait une attaque qui inflige des dégâts dans la zone autour de lui et brise la glace.

Nous avons également choisi d’intégrer de la narration dans le jeu sous forme sonore. Cela nous permettait à la fois de raconter une histoire et d’appuyer le déroulement progressif du niveau mais également de donner des informations supplémentaires et des indices sur les ennemis et les objectifs à

atteindre. Ainsi nous avons deux personnages qui communiquent entre eux et permettent une immersion plus facile et rapide du joueur. D'un côté, il y a Evan qui joue le rôle du partenaire de June et qui lui parle au travers d'une oreillette. C'est lui qui donne des informations quant aux ennemis et objectifs à remplir. Il apporte une pointe d'humour qui donne une impression de complicité avec le joueur. Via les informations qu'il apporte, ce personnage a un rôle complémentaire au tutoriel. De l'autre côté, June incarnée par le joueur, répond à Evan et suit ses indications. Un troisième personnage a été intégré en début de niveau : le commandant Jones. C'est elle qui donne une formation au joueur sur les techniques de combat et lui permet de prendre en main les compétences primaires de son personnage.



Figure 34 : Personnage non joueur qui joue le rôle de tutoriel. Il répond aux commandes vocales du joueur

Les résultats de cette expérience :

Suite à la présentation de ce projet devant les élèves et professeurs de la formation à la fin des trois semaines allouées à ce projet, nous avons pu leur faire tester notre niveau. Dans l'ensemble, même si le jeu présente différents bugs qui le rendent très difficile à finir, nous avons tout de même réussi à y implémenter au minimum tout ce que nous souhaitions. Observer différents joueurs y jouer m'a permis de constater les objectifs réellement atteints ainsi que les aspects à améliorer du jeu. Ainsi j'ai pu remarquer :

-Même si je n'avais pas eu le temps de programmer autant d'actions que je le souhaitais, il y avait tout de même différents comportements de joueur qui se distinguaient. Certains combattaient leurs ennemis au corps à corps, d'autres utilisaient plutôt la magie et préféraient se placer en hauteur pour faire tomber le décor sur leur adversaire. Les différents parcours, aériens comme terrestres, étaient utilisés.

- Malgré un tutoriel et des dialogues, nous devions ré-expliquer les commandes du jeu, et cela, pour deux raisons. La première étant que nous étions dans un environnement bruyant où il était difficile d'entendre correctement les instructions données au joueur. On m'a d'ailleurs fait remarquer qu'il aurait été agréable d'avoir des sous titres ajoutés au jeu car certains joueurs jouent également sans le son. La seconde raison est qu'une fois immergé dans le jeu le joueur cherche rapidement l'action et ne porte pas forcément son attention sur la voix du personnage non joueur qui lui parle. J'ai remarqué que les joueurs sont plus attentifs aux instructions lorsqu'elles sont données durant une cinématique car il est contraint de la regarder et ne peut agir. Peut être qu'il aurait été donc plus judicieux de montrer aux joueurs les actions possibles dans une courte cinématique puis de lui demander de les reproduire.

- Dans la version finale du projet la seule interface présente est la jauge de vie de June qui diminue en fonction des dégâts reçus et augmente lorsqu'une compétence de soin est lancée. Tout autre indication, comme la vie restant aux ennemis, est indiqué textuellement. Cette interface est incomplète. Il en aurait fallu une pour indiquer la quantité de vie des ennemis et indiquer les actions possibles au joueur. Pour les ennemis, nous aurions pu créer une barre de vie avec différents effets visuels pour indiquer le type d'adversaire que le joueur affronte. Par exemple une jauge rouge simple pour un ennemi normal et une jauge dorée pour indiquer qu'un ennemi est protégé par un bouclier. Seule la barre de vie de l'ennemi ciblé aurait été affichée. En ce qui concerne les actions, un symbole indiquant l'action et la touche à utiliser lorsque le joueur vise ou est à proximité d'un objet permettant cette action aurait permis au joueur d'être plus autonome sur le jeu.

- Le jeu manque également de feedback visuels. Notamment lorsqu'un ennemi est touché par une attaque. Il est donc difficile au joueur (surtout lorsqu'il est éloigné de l'adversaire) de savoir si son attaque l'a touché ou non. Un effet visuel pour montrer l'impact aurait permis de combler ce défaut. J'ai de plus remarqué qu'il était difficile au joueur de savoir précisément à quelle distance de l'adversaire il se situait. En effet le type d'attaque lancé dépend de la distance entre le joueur et l'ennemi. Par exemple si le joueur est situé à une distance entre 500 et 1500 il va tirer des flèches. Cependant si cette distance est supérieure à 1500, le joueur va lancer une compétence de soin. Là encore un effet visuel comme un halo lumineux autour du personnage aurait indiqué quel type d'attaque le joueur pouvait lancer. Par exemple un halo rouge si la distance est inférieure à 500, bleu si elle est située entre 500 et 1500 et vert si elle est supérieure à 1500.

- L'intelligence artificielle des ennemis était assez imprévisible. En effet le changement entre deux comportements était imprécis. Notamment lorsqu'il détectait le joueur, l'ennemi finissait d'abord son premier mouvement (qui le plus souvent était de se déplacer vers un point) avant de se focaliser sur le joueur. Cependant si celui-ci n'était plus à proximité de l'ennemi lorsqu'il finissait son mouvement alors il n'était plus détecté par l'intelligence artificielle qui, dans ce cas, continuait avec son premier comportement. Pour attirer un ennemi, le joueur devait constamment rester devant lui en espérant être correctement positionné pour focaliser l'attention de l'adversaire à la fin de son déplacement. De même lorsque l'ennemi touchait une surface gelée, il finissait son mouvement avant de se figer.

- La reconnaissance vocale était imprécise pour deux raisons. La première étant le bruit environnant lorsque les joueurs ont pu tester le jeu et qui empêchait la voix des joueurs d'être correcte-

ment perçue. La seconde étant que la distinction des mots lorsqu'ils sont prononcés est calquée sur ma propre voix. En effet lorsque nous avons programmé les mots à prononcer, ceux-ci avaient une certaine "marge d'erreur" de prononciation, en fonction (je suppose) de la tonalité de ma voix, ma façon de prononcer, mon intonation, etc. Donc si un joueur prononçait les mots différemment de ma façon de prononcer, sa voix était plus difficilement détectée par le plug-in de reconnaissance vocale. D'ailleurs, à certains moments du jeu, c'était moi qui prononçais les mots afin que le joueur puisse continuer de jouer.

Conclusion du projet :

Lorsque nous nous sommes lancées sur ce projet, nous voulions créer une expérience intégrant nos tests effectués au premier semestre. C'est pourquoi ce projet est très proche de ce que je voulais créer pour mon mémoire, même si la reconnaissance vocale n'en fait pas partie. Il illustre une partie de ma problématique : générer un gameplay grâce au renforcement. Je voulais également, au travers de ce projet, répondre à différentes questions que je me posais. A savoir, comment intégrer un tel système dans un niveau de jeu ? Est ce que cela s'intègre bien ? Si ce genre de technique n'a encore pas ou peu été utilisée, est ce parce qu'elle est inintéressante ? Qu'est ce qu'un tel système apporterait de plus à un jeu ? Je me suis senti confortée dans mon choix et dans la voie que j'ai choisi de prendre cette année, lorsque j'ai vu les joueurs interagir avec leur personnage et l'environnement du jeu différemment des autres et même, je dirai, de manière unique. Cela me renforce dans l'idée que dans un jeu nous avons tous un comportement et des décisions qui nous sont propres. Il me semble donc possible de laisser aux joueurs la possibilité de créer eux-même leur propre gameplay, leur propre façon de jouer et grâce à la compréhension de leur profil comportemental de générer le décor du jeu. C'est ce que je souhaitais réaliser au second semestre : générer le décor en fonction du profil comportemental du joueur. Le projet Transcendance m'a permis de comprendre comment intégrer un système de renforcement capable de comprendre les agissements du joueur pour faire évoluer les compétences de son personnage. Cette expérience m'a donné toutes les bases requises au second semestre pour générer mon décor et faire évoluer mon renforcement en y ajoutant de la fuzzy logic afin de rendre l'évolution des compétences plus fluide (avec une vitesse de calcul du code plus rapide) et plus automatisée (combiner de la programmation blueprint avec de la programmation C++).

II.3) Les évolutions et les risques

Réaliser ces expérimentations, m'a permis de comprendre les avantages, les inconvénients et les limites de chaque méthode et réflexion que j'ai pu avoir au cours de cette année. Ainsi j'ai pu déterminer les différentes notions à améliorer, modifier, ou peut être-même retirer.

Un de ces ajustements concerne notamment le "timer" et le renforcement négatif qui peut y être associé. En effet un joueur n'utilisera pas constamment les actions qui doivent être renforcées. Dans les jeux de rôle les joueurs passent beaucoup de temps à se déplacer dans le monde ouvert ou à marchander, parfois fabriquer des objets, etc. Pendant qu'il réalise ces activités il faut arrêter le "timer" pour éviter tout renforcement négatif injustifié. Le problème qui se pose est de savoir à quel moment on peut déclencher et arrêter ce "timer". La première réponse serait de le déclencher lorsque le joueur doit résoudre une situation donnée avec ces actions (combat, énigme, jeu de plateforme) et de l'arrêter lorsque la situation a été résolue. Mais que se passerait-il dans un jeu si le joueur devait s'absenter pendant que le "timer" tourne ? Par exemple dans un jeu d'énigme et de plateforme, si le joueur s'absente alors que son personnage est accroché à un mur, le "timer" continuerait d'engendrer un renforcement négatif sur les actions non utilisées ; ce qui serait complètement injustifié. Il faudrait donc détecter lorsque le joueur est présent ou non avec un système qui vérifierait en permanence si une action est exécutée dans le jeu : mouvement du personnage ou compétence utilisée, rotation de la caméra, utilisation de l'interface, etc.

Par ailleurs, si certains joueurs préfèrent jouer d'une seule manière et améliorer qu'une seule voie d'évolution, d'autres vont préférer évoluer de différentes manières, dans différentes voies et même parfois adapter leur comportement aux situations rencontrées. Ces joueurs doivent donc pouvoir revenir vers les actions qui ont été désactivées, car leur renforcement est tombé à zéro. Pour illustrer ces propos, imaginons un joueur qui joue de manière furtive et qui, à un moment donné, va vouloir changer de stratégie et jouer brutalement en fonçant tête baissée vers ses adversaires. Jusqu'à maintenant, les actions liées à ce jeu plus brutal, seraient désactivées et le joueur ne pourrait pas changer sa façon de jouer. Il faudrait donc imaginer que lorsque le renforcement des actions débloquées par le joueur retombe à zéro faute d'utilisation ; les anciennes actions qui ont été désactivées puissent être de nouveau disponibles avec un renforcement par défaut de 0,5. Cependant, cette méthode a le désavantage d'être progressive et un joueur qui veut changer de manière de jouer soudainement ne pourrait pas, car il faudrait le temps au renforcement négatif de refaire apparaître les actions devenues indisponibles. Néanmoins, il serait tout à fait possible, pour résoudre ce problème, que le renforcement négatif ne s'applique pas aux actions disponibles au commencement du jeu. Ainsi le joueur pourrait faire évoluer ces actions de différentes manières. Seules les évolutions de ces actions de base auraient un renforcement négatif qui s'appliquerait. Dans ce cas, si le joueur choisit d'abandonner une voie au profit d'une ou plusieurs autres alors cette première voie finirait par se désactiver et les autres, évoluer en fonction de ses actions. Pour autant, si le joueur choisit de revenir vers cette première voie, il le pourra également car les actions basiques permettant d'y accéder seront toujours disponibles. Enfin, il pourrait être intéressant de permettre au joueur de verrouiller une action qu'il aurait obtenue. Verrouiller une action l'exempterait de renforcement négatif. Ce mécanisme permettrait au joueur de garder une action qui l'intéresse même s'il sait qu'il l'utilisera peu. En effet, certaines actions sont trop occasionnelles

pour être utilisées régulièrement c'est le cas des sortilèges d'effet curatif qui permettent de dissiper un effet négatif sur un personnage. Ce genre de compétence n'est utilisé qu'occasionnellement, pour autant c'est une compétence indispensable à un joueur qui veut soutenir ses coéquipiers. Le mécanisme de verrouillage d'action serait donc très utile dans ce cas précis.

Une deuxième évolution possible de mon projet est d'intégrer de la logique floue (fuzzy logic) pour déterminer les règles d'évolution des différentes actions. Si au cours de mon année je n'ai cessé de réfléchir au moyen de l'intégrer à ma méthode de renforcement, sans vraiment comprendre comment le faire. J'ai fini par voir que je n'abordais peut-être pas la problématique sous le bon angle. Ce que je fais actuellement, c'est que le renforcement d'une ou de plusieurs actions entraîne la disponibilité d'une troisième. Or il pourrait être plus intéressant de débloquent une nouvelle action en fonction du renforcement des caractéristiques qui la composent. Utiliser les compétences renforcerait leurs caractéristiques. Cependant chaque compétence serait composée d'une combinaison de caractéristiques qui lui est propre. En fonction du renforcement de ces caractéristiques, on obtiendrait une nouvelle compétence.



Figure 35 : Exemple de modifications de renforcement des actions

Bien que les différences entre les deux méthodes soit subtiles, cette dernière permettrait une plus grande variété de combinaisons et de liens entre différentes compétences. Par ailleurs, implémenter de la logique floue sur cette méthode me paraît plus simple et plus justifiée. Enfin le renforcement des caractéristiques liées au point de dégâts des compétences permettrait de trouver plus facilement une fonction mathématique calculant l'évolution des dégâts de celles-ci. Cette méthode me semble plus évolutive et permet de moins catégoriser chaque action.

Générer du contenu vidéoludique en fonction du comportement du joueur peut se généraliser au-delà du gameplay. Notamment au travers du comportement des ennemis qui pourraient également s'adapter au gameplay du joueur. De plus en plus dans les jeux vidéo, les intelligences artificielles sont développées pour modifier leur comportement en fonction de la manière dont le joueur appréhende une situation. C'est le cas dans "The last of us", développé par le studio américain Naughty Dog dont le déplacement des ennemis dans l'environnement s'oriente en fonction des déplacements du joueur. Ainsi, lorsque le joueur tue tous les ennemis présents dans une zone, les ennemis présents dans les zones alentours vont se diriger vers celle dans laquelle le joueur joue. Même principe si le joueur essaye de s'infiltrer et d'avancer sans affronter ses ennemis : ils vont modifier leur parcours pour maintenir la difficulté du niveau et forcer le joueur à s'adapter. Ce concept est très intéressant et mériterait d'être développé en donnant la capacité à l'intelligence artificielle de modifier sa manière de combattre le joueur. C'est-à-dire que les compétences d'attaque ou d'interaction avec le joueur varieraient en fonction du profil comportemental du joueur. Pour illustrer, prenons un jeu de rôle dans lequel le joueur affronte un ennemi. Si cet ennemi est défini comme ne pouvant se battre qu'au corps à corps, alors il va se diriger vers le joueur pour l'attaquer. Dans le cas où le joueur joue lui aussi au corps à corps, l'affrontement constituera un certain challenge, et maintiendra donc l'intérêt du joueur dans le jeu : esquiver ou parer les attaques de l'ennemi, avoir suffisamment de réflexe, réaliser le meilleur enchaînement de coups, etc. Cependant, si le joueur joue à distance de l'ennemi, le seul vrai challenge, sera d'éviter que l'ennemi ne se rapproche trop. Ce qui constitue un risque, qu'à répétition, le joueur ressente un certain ennui pour le combat. Si l'ennemi pouvait adapter ses compétences au joueur alors cela maintiendrait le challenge que le joueur a dans le jeu. Plutôt que de courir après le joueur, il pourrait bondir sur lui ou bien l'attaquer à distance, avoir des compétences de soin pour contrebalancer le temps qu'il passe à se déplacer, etc.

Enfin j'ajouterais que la plus grande difficulté que j'ai rencontré dans la réalisation d'un jeu qui s'adapte au comportement du joueur, vient de l'interprétation que l'on pourrait faire de ce comportement. En effet appréhender différentes manières de jouer inclut de comprendre ces comportements et ce qui pourrait leur paraître intéressant. Cependant, on ne peut que supposer ce qu'un autre va apprécier sans être définitivement sûr de soi (surtout si le type de comportement est radicalement opposé au nôtre). Le risque est de tomber dans la catégorisation des comportements de joueurs et de proposer selon ces catégories un contenu peu varié et redondant.

Conclusion

Cette année de master a été ponctuée par la réalisation de différentes expérimentations qui m'ont permis de mieux cerner toute l'amplitude et la complexité de la création de jeu permettant au joueur de développer son propre gameplay au travers des actions qu'il réalise.

Il était important pour moi de programmer un système capable de comprendre le comportement du joueur et de s'y adapter pendant que celui-ci joue, afin d'esquiver les traditionnels questionnaires que les joueurs remplissent et qui permettent, par la suite, la création ou l'évolution d'un jeu. Aussi, je dois avouer avoir passé beaucoup de temps sur la programmation de la méthode de renforcement et sur les règles d'évolution des actions au détriment de la génération du décor. Cependant, ces méthodes étant à la base de tout autre contenu je souhaitais les développer au maximum. Au final, toutes ces expérimentations me permettent d'entrevoir les possibles évolutions et améliorations de ce genre de système. Je regrette par ailleurs de n'avoir pu faire tester toutes mes expérimentations à des joueurs. Il serait très intéressant que je puisse les leur soumettre afin d'obtenir à la fois des retours sur mes méthodes mais également sur l'équilibrage du renforcement, des règles d'évolution des actions ainsi que sur le rapport entre le renforcement et l'agencement du décor.

Néanmoins, j'ai pu grâce à cela prendre en main et expérimenter des méthodes d'intelligence artificielle que je n'avais jamais programmé auparavant. Au travers de mes deux grandes expérimentations que sont le projet Transcendance et le projet d'arène, j'ai compris, notamment grâce aux erreurs que j'ai pu y faire, comment mettre en place les différents systèmes d'intelligence artificielle et surtout comment mieux les exploiter et mieux les combiner. C'est pourquoi, réaliser ces tests étaient une étape nécessaire à la progression de mes réflexions.

Conclusion Générale

Depuis plusieurs années je m'intéresse à la psychologie associée aux jeux vidéo. Etant moi-même une joueuse de jeux en tout genre et plus particulièrement de jeu de rôle, j'ai remarqué que le comportement des joueurs en ligne correspondait à différents traits de leur personnalité. Ce qui m'a amené à me demander si le comportement des joueurs au travers de leur manière de jouer, que ce soit de par les choix d'actions, de dialogue ou du type de personnage qu'ils jouent, était révélateur d'une partie d'eux-même. Et si c'était le cas, comment optimiser leur gameplay dans les jeux (peu importe le genre). Car, actuellement, peu de jeux proposent au joueur de choisir les types d'actions ou les compétences du personnage qu'il va vouloir utiliser, débloquer ou faire évoluer. Le gameplay des jeux reste défini à l'avance pour évoluer d'une certaine manière à un instant précis. Il me semblait intéressant de donner au joueur l'impression qu'il peut créer un gameplay unique, qui ne correspond qu'à lui seul, en masquant le plus possible les délimitations et les règles d'évolution de ses actions.

J'ai choisi d'aborder ce mémoire sous un angle plus psychologique que lié au game design. C'est en me basant sur des notions de psychologie comportementale que j'ai voulu créer des mécaniques de jeu. Ainsi, je me suis d'abord intéressée aux facteurs qui régissent la prise de décision des actions chez l'être humain et à ce qui le motive à agir. Pourquoi je fais ce que je fais ? Pourquoi dans un jeu vidéo j'aime contrôler mes cibles ? Comment est-ce que j'aime les contrôler ? Qu'est-ce que cela m'apporte ? etc. Bien entendu, certaines mécaniques de jeux déjà existants ont inspirées mon travail. Puis j'ai cherché à comprendre ce que les jeux représentaient pour les joueurs. Quels liens particuliers entretiennent-ils avec lui ? Comment s'y impliquent-ils ? Et si les mondes virtuels sont aussi différents du monde réel quant aux émotions et besoins des joueurs qu'ils satisfont ? En faisant évoluer mes recherches j'ai compris que les joueurs s'impliquaient bien plus facilement à la réalisation d'un objectif dans les mondes virtuels que dans le monde réel, car la prise de risque dans les mondes virtuels est nulle. Il n'y a pas de conséquences réelles à l'échec hormis peut être le sentiment de déception rapidement remplacé par le fait que le joueur puisse recommencer son objectif. Lorsqu'il réussit il est salué tel un héros. Et tout cela sans contrainte sociale, morale ou physique qu'il aurait dans la réalité. Le joueur se sent libre de réaliser ce qu'il souhaite réaliser et de la manière programmée qui lui convient.

J'ai donc cherché grâce à l'intelligence artificielle un moyen de créer un système qui génère l'évolution des actions du joueur en fonction de sa manière de jouer. En m'inspirant des mécaniques de d'autres jeux et réalisant des recherches sur des systèmes capables de comprendre et d'analyser le comportement humain dans les univers virtuels, j'ai pu m'orienter vers les méthodes de renforcement, de logique floue et de génération procédurale. En divisant ma problématique en trois projets, j'ai pu tester rapidement chacune de ces méthodes que j'ai par la suite combinées pour essayer de créer un niveau de jeu où les actions et l'environnement du joueur évoluent selon son comportement. En réalisant ces expérimentations j'ai pu prendre conscience des avantages et limites de chacune d'entre elles et n'ai cessé de chercher un meilleur moyen de mettre en place ce système. Au final j'ai pu trouver de nouveaux axes de recherche que je souhaiterai continuer d'explorer au-delà de ce master. L'objectif premier de ces travaux était pour moi, de poser les bases d'un système générationnel de gameplay et de décor et d'en comprendre la complexité afin de le faire évoluer. Je pense en avoir compris une grande partie et continuerai d'en découvrir toujours plus au fur et à mesure de mes prochains tests, de mes prochaines

erreurs et de leur résolution.

Bibliographie

Chris **Crawford**, (1984), *The art of computer game design*, Ed. Osborne/McGraw-Hill, Californie.

Edward **Deci**, et Richard **Flaste**, (1995), *Why we do what we do : understanding self motivation*, Ed. Penguin books, New York.

Graham **Little**, (2016), *The origin of counciousness. 6th Ed.*, Ed. Institute of Theoretical and Applied Social Science, New Zealand.

Richard **Bartle**, (2003), *Designing virtual worlds*, Ed. New Riders, United States.

Ian **Millington**, et John **Funge**, (2009), *Artificial intelligence for games. 2nd Ed.*, Ed. Morgan Kaufmann, Burlington.

Marie-Hélène **Tramus**, Michel **Bret**, et Edmond **Couchot**, (2003), *La seconde interactivite : vers de nouvelles pratiques artistiques*, «A segunda interativida. Em direção a novas práticas artísticas», UNESP, Brésil.

Webographie

Articles

Benoît **Girard**, Gabriel **Robert**, et Agnès **Guillot**, (2001), « *Jeu vidéo et intelligence artificielle située* », sect. AnimatLab.

<https://www.researchgate.net/publication/228603670>

Carlo **Fabricatore**, (2007), « *Gameplay and game mechanics : a key to quality in videogames* », University of Huddersfield, sect. Conference paper,

https://www.researchgate.net/publication/236168267_Gameplay_and_game_mechanics_design_a_key_to_quality_in_videogames

Nicolas **Ducheneaut**, et Nicholas **Yee**, (2012), « *Les jeux vidéo en ligne, un miroir de la personnalité des internautes ?* » Questions de communication, no 21.

<https://questionsdecommunication.revues.org/6571>.

Conférences vidéos

Jane **McGonigal**, (2010), *Gaming can make a better world*.

https://www.ted.com/talks/jane_mcgonigal_gaming_can_make_a_better_world.

Seth **Priebatsch**, (2010), *The game layer on top of the world*.
https://www.ted.com/talks/seth_priebatsch_the_game_layer_on_top_of_the_world.

Tony **Robbins**, (2006), *Pourquoi nous faisons ce que nous faisons*.
https://www.ted.com/talks/tony_robbins_asks_why_we_do_what_we_do?

Logiciel

Juan **Rada-Vilela**, (2016), *Fuzzy Lite (version fuzzylite 4.0)*.
<http://www.fuzzylite.com/>.

Sites internet consultés

Geoffrey **Delcroix**, (2016), « *Quand les jeux vidéo analyseront votre personnalité* », LINC : laboratoire d'Innovation Numérique de la CNIL.
<https://linc.cnil.fr/fr/quand-les-jeux-video-analyseront-votre-personnalite>.

Posté par, **Hela**, (2012), « *Psychologie loi du renforcement* », Réseau encyclopédique.
<http://psychologie.savoir.fr/psychologie-loi-du-renforcement/>.

Posté par, **gregkwaste**, (27/10/2016), « *No Man's Sky – Procedural Content | 3dgamedevblog* », 3DGameDevBlog.
<http://3dgamedevblog.com/wordpress/?p=836>.

1 2 3 Test, « *Théorie des cinq grands facteurs de personnalité (Big Five)* »,
<https://www.123test.fr/theorie-des-cinq-grands-facteurs-de-personnalite/>.

1 2 3 Test, « *Test de Personnalité* », (Pour réaliser le test des big 5)
<https://www.123test.fr/test-de-personnalite/>.

Jeux vidéo cités

Développeur : **38 Studio & Big Huge Games**, (2012), Kingdom of Amalur : Reckoning, Genre : Jeu de rôle, Editeur : Electronic Arts.

Développeur : **Cristal Dynamics**, (2013), Tomb Raider, Genre : Action / Aventure, Editeur : Square Enix.

Développeur : **Dontnod Entertainment**, (2015), Life is strange, Genre : Jeu narratif épisodique, Editeur : Square Enix.

Développeur : **Gearbox Software**, (2009), Borderlands, Genre : FPS / RPG, Editeur : 2K Games.

Développeur : **Hello Games**, (2016), No Man's Sky, Genre : Action / Aventure, Editeur : Hello Games.

Développeur : **Irrational Games**, (2013), Bioshock Infinite, Genre : FPS, Editeur : 2K Games.

Développeur : **Lionhead Studio**, (2005), Black & White 2, Genre : Stratégie en temps réel, Editeur : Electronic Arts.

Développeur : **Naughty Dog**, (2013), The last of us, Genre : Survival horreur, Editeur : Sony Computer Entertainment.

Développeur : **Pearl Abyss**, (2016), Black Desert, Genre : MMORPG, Editeur : Kakao games Distributeur : Daum Communications.

Développeur : **Rockstar North**, (2013), Grand Theft Auto V, Genre : TPS, Editeur : Rockstar Games.

Développeur : **Square Enix**, (2010), Final Fantasy XIV online, Genre : MMORPG, Editeur : Square Enix.

Développeur : **The Coalition**, (2016), Gears of War 4, Genre : TPS, Editeur : Microsoft Studios.

Développeur : **Vigil Games**, (2012), Darksiders II, Genre : Action / Aventure, Editeur : THQ.

Développeur : **ZeniMax Online Studio**, (2014), The Elder Scroll online, Genre : MMORPG, Editeur : Bethesda.

Table des illustrations

Figure 1 : Boucle de comportement du personnage War	12
Figure 2 : Après avoir vécu la première situation le joueur remonte dans le temps pour revivre la même scène, mais cette fois un autre choix s'offre à lui	17
Figure 3 : Le joueur doit choisir quel chemin il va emprunter	17
Figure 4 : Dojon du jeu Final fantasy XIV online dans lequel les joueurs affrontent des monstres. Les donjons sont des environnements isolés du monde global du jeu comme le montre cette carte.	21
Figure 5 : Parade formée par les joueurs du jeu Final Fantasy XIV online	22
Figure 6 : Pour que l'expérience d'un jeu soit optimale la progression de son niveau de difficulté doit se trouver dans le «Flow channel»	23
Figure 7 : Graphique représentant les types de joueurs mis au point par Richard Bartle	27
Figure 8 : Le joueur gifle sa créature pour en changer le comportement	34
Figure 9 : La créature a modifié son comportement	35
Figure 10 : Jauge de chaque arme avec leur niveau d'utilisation	35
Figure 11 : Statistiques que le joueur peut améliorer grâce à ses point de caractéristiques	37
Figure 12 : Compétences du personnage qui se débloquent en fonction du niveau de la compétence (ici la magie noire est au niveau 17) et que le joueur peut acheter moyennant des points de compétences.	38
Figure 13 : Plus le joueur tue de belettes et plus il va acquérir d'informations sur l'animal. Comme par exemple, la barre de vie de l'ennemi qui s'affiche.	39
Figure 14 : Le joueur voit les différents décors qu'il peut faire apparaître.	40
Figure 15 : Différents univers et modélisations créés par génération procédurale dans No man's sky.	41
Figure 16 : Statistiques mondiales des choix des joueurs. L'encoche verte correspond au choix que le joueur a fait lui-même.	42
Figure 17 : Arbre de talents de sorcellerie de Kingdom of Amalur : Reckoning dans lequel le joueur dépense des points de compétence.	43
Figure 18 : Action et évolutions des actions liées à la mobilité du joueur	49
Figure 19 : Graphique montrant les liens entre chaque action, leurs évolutions, et les pré-requis pour les utiliser	50
Figure 20 : Premier niveau me permettant de tester la méthode de renforcement	52
Figure 21 : Résultats de la méthode de renforcement	52
Figure 23 : Initialisation de chaque action	53
Figure 22 : Liste des actions qui seront renforcées	53

Figure 24 : <i>Le Brain Character envoie au Brain Manager quelles actions sont utilisées</i>	54
Figure 25 : <i>Le Brain Manager initialise dans plusieurs tableau les différentes informations de chaque action : le nom, le poids de renforcement et la valeur du timer.</i>	54
Figure 26 : <i>Différents lieux du niveau d'arène</i>	56
Figure 27 : <i>Actions programmées dans le niveau d'arène et leurs évolutions</i>	57
Figure 28 : <i>Exemple de «Crisp» de fuzzy logic</i>	58
Figure 29 : <i>Exemple de «Crisp» que j'ai réalisés pour essayer la méthode de fuzzy logic. Ici on détermine le type d'attaque du personnage joueur en fonction de la distance et de l'orientation du personnage par rapport à l'ennemi.</i>	58
Figure 30 : <i>Dans ce cas présent, je n'ai pas utiliser de magie pour vaincre mes adversaires, mais beaucoup utilisé de cover en attaquant par surprise</i>	61
Figure 31 : <i>Ici, j'ai joué en utilisant la magie, les poisons et les cover. En revanche je n'ai pas utilisé de «cachette» (cube transparent) dont on constate la disparition.</i>	61
Figure 32 : <i>Boss de fin de niveau que le joueur doit affronter. La manière de vaincre le boss diffère selon le résultat du renforcement.</i>	66
Figure 33 : <i>Différents lieux du niveau. Respectivement : la salle d'entraînement, le premier affrontement et la salle du boss</i>	67
Figure 34 : <i>Personnage non joueur qui joue le rôle de tutoriel. Il répond aux commandes vocales du joueur</i>	68
Figure 35 : <i>Exemple de modifications de renforcement des actions</i>	72

Index des termes

PNJ : ou personnage non joueur. Représente tous les personnages “vivant” dans l’univers du jeu mais qui sont contrôlés par intelligence artificielle et non par le joueur.

Pattern : schéma d’actions et d’enchaînement d’actions régissant le comportement des ennemis d’un jeu. Ces actions et enchaînement d’actions répondent à des règles (exemple : si le joueur est au corps à corps faire : “action 1”, puis “action 2”, puis “action 3”). Chaque type d’ennemi a un pattern différent.

TPS : littéralement Third Person Shooter. Désigne les jeux de tir avec un angle de vue caméra à la troisième personne. Le terme est plus généralement employé pour désigner tous types de jeux dont le joueur incarne un personnage avec cet angle de caméra. Jeux de tir ou non.

Achievement : objectif secondaire que les joueurs peuvent choisir de remplir ou non. La plupart du temps les achievements n’ont pas d’impact sur l’univers du jeu ou l’histoire que le joueur va vivre. C’est un contenu additionnel indépendant qui permet aux joueurs de gagner des titres, des montures, des mascottes ou tout autre objet consommable ou décoratif.

Skin : Apparence d’un vêtement ou d’une armure de personnage. Permet de personnaliser l’apparence de son personnage.

Mascottes : Aussi appelé “pets” (appellation anglaise) qui désigne des “animaux de compagnies” virtuels qui suivent le joueur dans ses déplacements. Ils n’ont pas d’utilité première mais certains jeux vidéo proposent différentes activités avec les mascottes. A noter que toutes les mascottes n’ont pas forcément l’apparence d’un animal.

Gilde : association de joueurs ayant un but commun et qui de par ce but vont jouer régulièrement ensemble et accomplir divers objectifs.

Donjons : aussi appelé “Instances” sont des zones fermées et définies dont le déroulement du contenu reste le même autant de fois que les joueurs viennent y affronter les monstres et les boss qui s’y trouvent. Ce sont des zones dites “instanciées”. C’est à dire qu’elles sont dupliquées autant de fois qu’il y aura de groupes de joueurs voulant y accéder. Chaque groupe accédant à une copie de la zone originelle. L’accès à ces zones requiert le plus souvent des conditions particulières : être en groupe ou non, avoir débloqué son contenu via une quête, niveau du personnage ou d’équipement, etc.

Gamification : Utilisation des mécaniques et des codes propres aux jeux vidéo dans des applications n’étant pas des jeux afin de contrôler ou de modifier le comportement de la cible.

PvP : littéralement “Player versus Player” ou “Joueur contre Joueur” (JcJ). Désigne les contenus de jeux ou les jeux dans lesquels les joueurs s’affrontent entre eux

Item : tout objet que les joueurs et les personnages non joueurs peuvent ramasser, collectionner ou utiliser.

Skill : compétences actives ou passives d'un personnage. Attaques qu'il peut réaliser.

MMORPG : littéralement Massively Multiplayer Online Role Playing Game. Désigne les jeux de rôle en ligne possédant une grande communauté de joueurs interagissant dans le même univers au moyen de serveurs qui collectent et transmettent des données et mémorisent la progression des joueurs.

Classe : dans les jeux de rôle, les classes désignent différents types de personnages aux capacités, compétences et statistiques qui leur sont propres. Chaque classe oriente les joueurs vers une manière de jouer (défendre, soutenir, soigner, infliger de lourds dégâts). Exemple de classes que l'on trouve régulièrement dans les jeux de rôle : Assassin, Mage, Guerrier, Archer, Prêtre.

FPS : littéralement First Person Shooter. Désigne les jeux de tir avec un angle de vue caméra à la première personne. Le joueur voit au travers des yeux du personnage. Le terme est plus généralement employé pour désigner tous types de jeux dont le joueur incarne un personnage avec cet angle de caméra. Jeux de tir ou non.

Cover : élément du décor permettant au joueur de se mettre à couvert.

Arbre de talent : représente les différentes compétences ou caractéristiques du personnage dans lequel le joueur investit des points pour développer ou débloquent de nouvelles compétences ou améliorer les statistiques de son personnage.

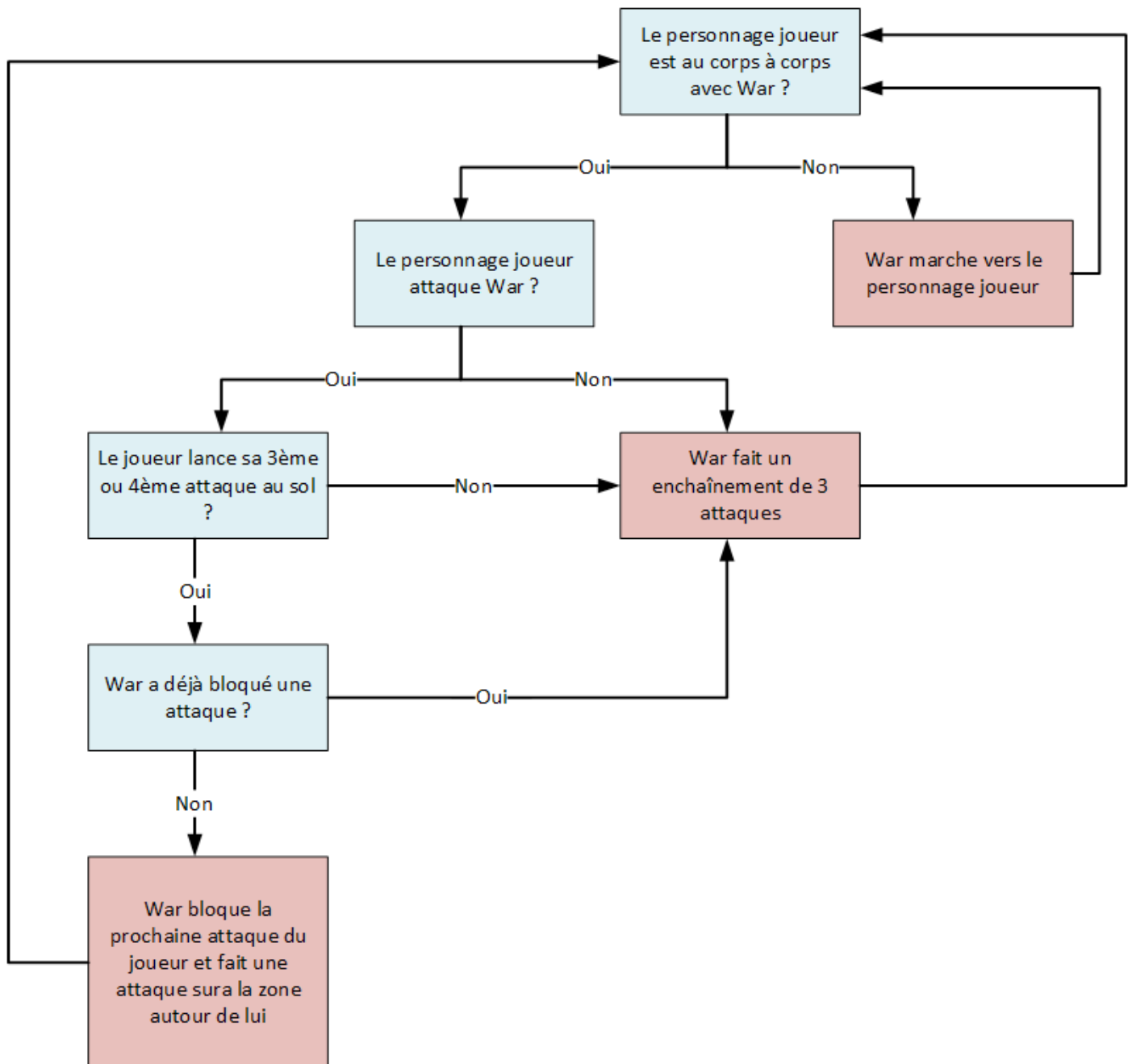
Input : désigne la voie d'entrée d'un système par laquelle les informations vont passer.

Cooldown : temps d'attente qui se déclenche lors de l'utilisation d'une compétence du personnage et qui empêche le joueur de la réutiliser avant un certain temps.

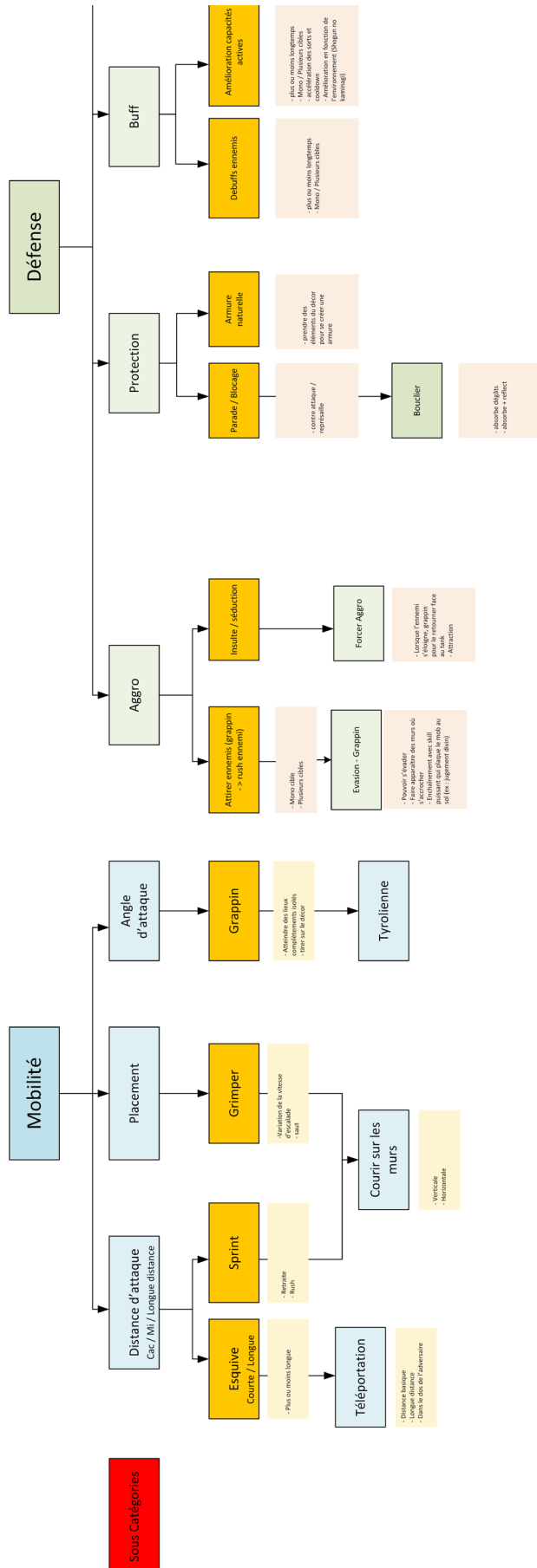
Aggro : diminutif de agressivité

Annexes

Pattern du personnage War du jeu Darksiders II

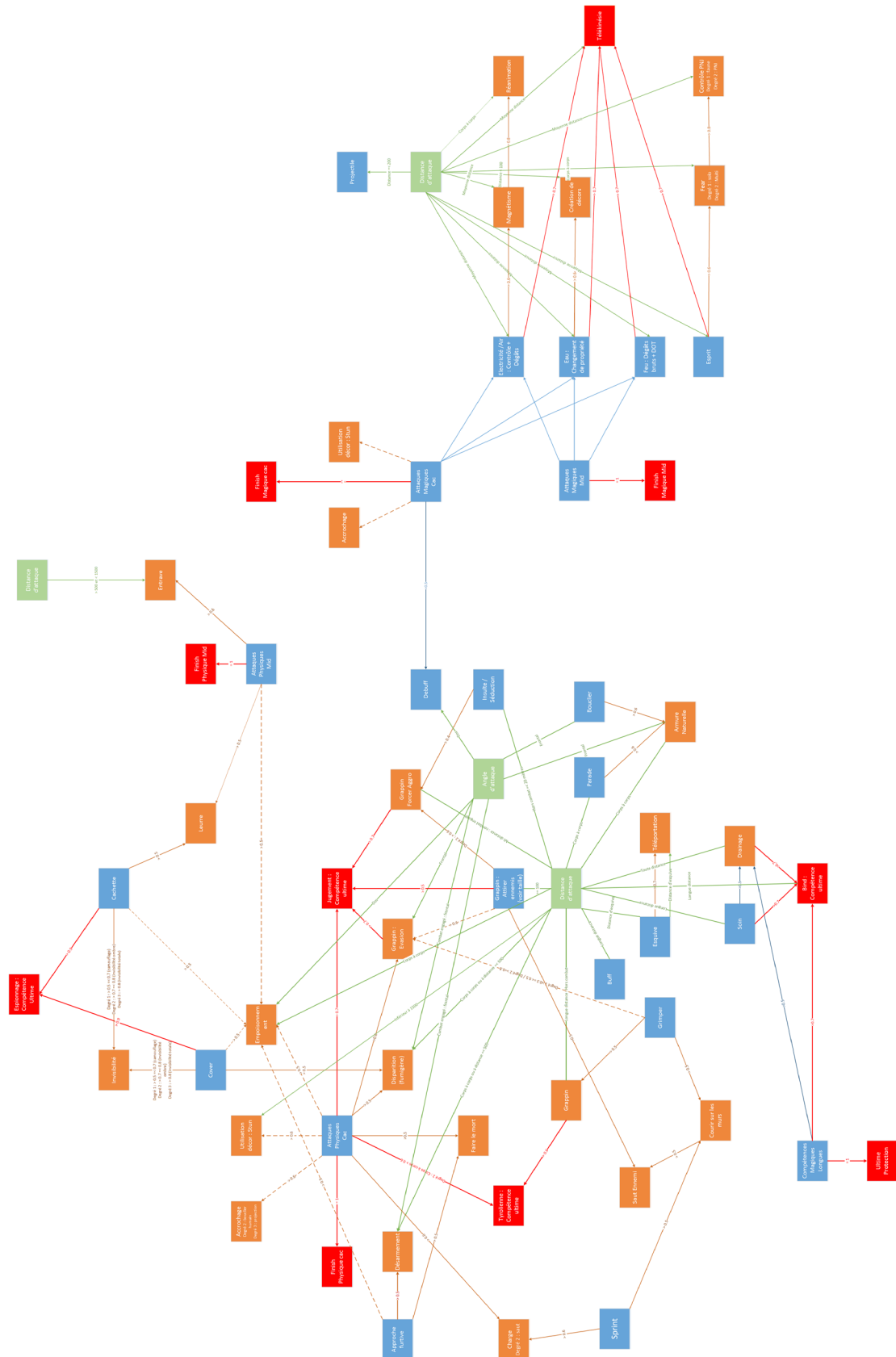


1er Arbre de talents : (divisé en 2 images)

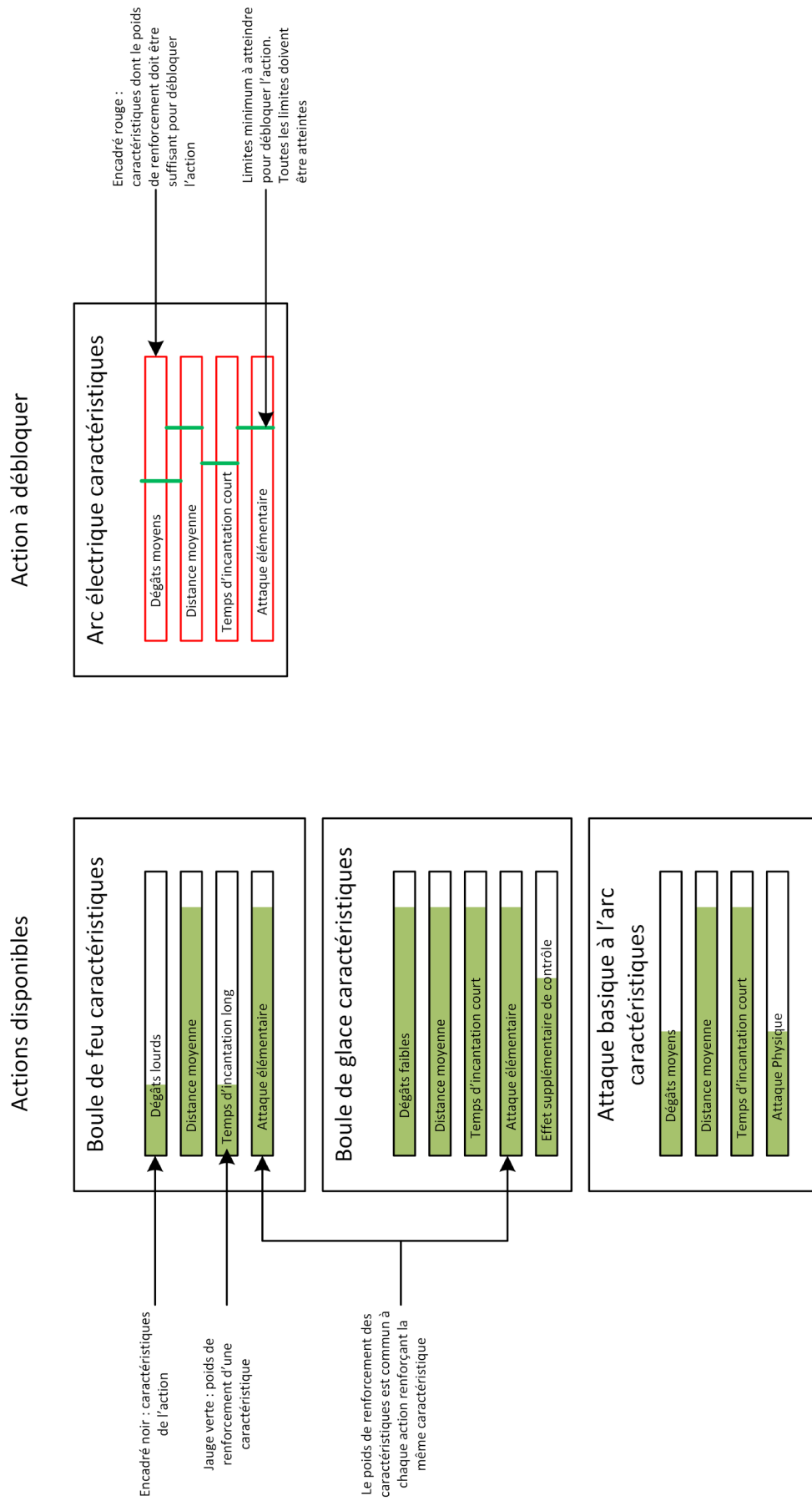




2nd Arbre de talents



Exemple de modifications de renforcement des actions



Questionnaire : Analyse comportementale de joueurs (questions)

Analyse comportementale de joueurs

Bonjour et merci de prendre le temps de répondre à ce questionnaire. Dans le cadre de mon mémoire de fin d'année je cherche à identifier différents profils psychologiques découlant des actions que les joueurs réalisent dans les jeux vidéo. On a tous une manière de jouer sur les jeux, et ce, même si nous jouons au même. Et pourtant notre comportement et notre façon d'aborder une situation est différente d'un autre joueur. Pour ce faire je vais vous demander de décrire le plus précisément possible vos expériences de jeu en illustrant ci possible avec des exemples. Faites une introspection de votre gameplay et racontez moi votre vie de joueur !

***Obligatoire**

De quelle manière jouez vous ? Précisez les actions et les capacités (actives / passives) que vous utilisez le plus régulièrement selon les expériences qu'offrent les jeu. *

Je vous demande ici de décrire votre gameplay dans tout type de jeu. Etes vous plutôt du genre à foncer tête baissée, ou à supporter vos alliés, peut être les deux, ou peut être encore aimez vous jouer de manière polyvalente ? Avec quelles armes jouez vous (longue distance, mi-distance, cac) ? Ecrivez le nom des armes (surtout pour les fps). Précisez les actions que vous exécutez le plus. Exemple : rester à couvert et contourner l'adversaire, s'infiltrer, prendre par surprise, attaquer au corps à corps, défendre, surplomber l'ennemi, etc. Si vous avez une histoire à raconter n'hésitez pas.

Votre réponse

Quelles sont les actions ou les capacités (actives / passives) qui enrichiraient votre style de jeu et que vous aimeriez voir implémenter dans un jeu vidéo ? *

Il nous est tous arrivé une fois de nous dire "Ah dommage que cette action ne soit pas possible dans le jeu". Quelle est-elle ? Exemple : "J'aimerais bien pouvoir grimper à un mur et pouvoir tirer sur mes ennemis depuis ce mur". Si elle vous permettrait de faire des combos, précisez l'enchaînement d'actions.

Votre réponse

ENVOYER

N'envoyez jamais de mots de passe via Google Forms.

Questionnaire : Analyse comportementale de joueurs (réponses)

Réponses à la question 1 :

En général je favorise les approches furtives et les stratégies. J'essaie d'utiliser un maximum d'outils pour préparer la situation (attirer des ennemis dans un coin, couper la lumière, poser des mines, des pièges, debuff sur les ennemis, améliorer mes stats, préparer des zones sécurisées etc...). Ensuite j'attaque d'abord à distance via snipers ou arcs ou sorts, pour d'une part diminuer le nombre d'opposants et les attirer dans les zones que j'ai préparée au préalable. Une fois la situation à mon avantage, si il reste des ennemis, je viens au plus proche et j'attaque en assaut.

longue distance voir très longue (snipe, avec infiltration et en hauteur)

Lorsque le jeu le permet, je jouer en général soutien (par exemple dans Overwatch) soigner, supporter. Je préfère rester à l'arrière et veiller au bon fonctionnement de l'équipe que foncer dans le tas.

En règle général, dans les jeu fps, je suis plus du genre à jouer avec des armes longues distances. Je suis du genre à contourner l'ennemi dans le dos, ou à rester caché pour surprendre plutôt que d'aller au combat de face.

Lorsque le personnage n'a pas de background j'y vais au feeling parfois discret assassin, parfois bourrin. Si le perso à un background je joue roleplay le plus souvent

J'ai un jeu plutôt solitaire et fourbe : ce que je préfère, c'est jouer l'infiltration et une fois repéré, foncer dans le tas. Je cherche toujours à approcher l'adversaire au plus près au moment où il ne s'y attend pas.

- Dans un FPS, je commence par sniper en étant de préférence planqué dans un spot proche de l'ennemi, mais discret (même si ça peut être difficile d'accès sans se faire repérer). Puis dès que j'en ai abattu quelques uns et que l'alarme est donnée, je vais dénicher les ennemis planqués derrière des éléments de décor dès qu'ils se mettent à couvert. Niveau armes, je suis toujours adepte des fusils de sniper, puis des armes type corps à corps mais toujours dans l'esprit "one shot, one kill". Jsuis pas un grand fan des mitrailleuses.
- Dans un jeu de combat, je privilégie les esquives et accrochages. Avoir l'autre à l'usure ou avec une méthode répétitive ne m'intéresse pas. Ce qui est vraiment plaisant pour moi, c'est de lui laisser le temps de se remettre après une attaque, puis de l'avoir par surprise.
- Dans un rpg, le premier personnage qui me tente est toujours l'assassin au corps à corps.
- Dans les jeux de stratégie, je suis un grand fan du brouillard de guerre et des embuscades.

Je joue plutôt de manière furtive, je ne fonce jamais dans le tas sans avoir réfléchi à mes futures actions.

En ligne, je joue support agressif. Mon rôle est de foncer tête baissée sur les objectifs pour prendre les adversaires de flanc, puis de faire des tirs de suppressions afin de ralentir la progression ennemie, prendre les objectifs et limiter les pertes alliées. Très efficace à la sulfateuse, très grandes chances de carry les parties, surtout contre des adversaires peu organisés.

Supporter les alliés / plutôt arme longue distance (snipe) et parfois courte portée (shotgun), s'infiltrer / par surprise / surplomber l'ennemi

Je ne joue pas très souvent aux jeux vidéos mais je suis plus habituée au moba qu'au fps. Je joue le plus souvent support, et je ne joue que très rarement au corps à corps. Je préfère les attaques à distance. Pour les fps comme je suis nulle, je fonce dans le tas et je me fais tuer tout de suite. Mais je me fait tuer très vite même quand je suis à couvert. Sinon j'aime bien garder des héros que je connais bien mais j'aime bien aussi en changer et en découvrir de nouveaux de temps en temps.

Je préfère le poste de damage dealer longue distance ou support, je préfère prendre avantage des erreurs des ennemis que de créer les opportunités moi même. Du coup, je préfère prendre un minimum de risque, pour forcer les joueurs impatient à prendre des initiatives et capitaliser sur une contre attaque. Mon comportement le plus habituel est de faire des dégâts petit à petit sur l'ennemi ou les adversaires, pour les forcer à perdre du terrain ou à m'attaquer de front. Mes armes favorites dans les FPS sont plutôt les armes précises, quitta à faire moins de dégâts, à partir du moment où je touche l'adversaire, c'est souvent du One Shot.

Observation, planification, action. Avec beaucoup de communication avec l'équipe si je joue en multi. J'ai une préférence pour les armes polyvalentes de base car la distance du combat est souvent contextuelle. Si j'ai la possibilité d'avoir une arme/pouvoir plus approprié/e à la situation, je saute dessus. Par exemple, si je suis le seul à défendre une position et que je me retrouve débordé, mon but est de retenir l'ennemi et de survivre en attendant les renforts. Mes possibilités sont multiples: Tirs de suppression ou directs, fumigène, grenade incendiaire...

Réponses à la question 2 :

J'aimerais pouvoir faire basculer des éléments du décor pour créer des obstacles et de couverts de manière dynamique et stratégique, afin de créer ma zone de confort.

grimper, et se cacher

J'apprécie en général d'avoir des interactions avec l'ensemble du décors. Je suis assez frustrée par exemple quand un endroit semble accessible mais pour des raisons techniques et inaccessible au joueurs. J'aimerais voir un environnement qui me permet de me cacher n'importe où, avec n'importe quel objet.

Un environnement full interactif, pas de murs invisibles. tu veux faire une chose => tu le fais ci tout.

Généralement, je trouve qu'il y a souvent beaucoup, voire trop de capacités possibles, mais des actions possibles peu variées à l'intérieur même du jeu.

Très rares sont les jeux où plusieurs solutions d'infiltration se valent (en excluant la méthode "je bourrine et en fait ça marche parce que le jeu est mal conçu").

Beaucoup de jeux de rôles ne proposent qu'une ou deux solutions pour les quêtes disponibles, ce qui est peu. Le meilleur jeu de rôle que j'ai testé propose au moins trois solutions par quête, et généralement, on a plusieurs solutions pour un seul type de personnage, et des issues différentes selon la manière dont on a monté son perso (par exemple on peut obtenir un droit de passage bloqué par des brigands en faisant une quête pour eux, en les payant, en leur volant la clef, en crochétant discrètement la porte qu'ils bloquent, en les tuant tous ou en leur faisant croire qu'on fait partie d'une confrérie de voleurs qu'ils admirent et qu'ils ont empiété sur le territoire d'un autre voleur).

Ce qui me frustre le plus souvent dans les jeux, c'est de me confronter à un univers fini et une voie toute tracée. Si des personnages différents découvrent un jeu qui n'a rien à voir, même s'il y a peu de capacités, je peux passer des décennies sur le jeu en question.

Avant d'imaginer de nouvelles capacités pour faire des combos, je préfère essayer moi-même d'imaginer ces combos, même si les concepteurs du jeu n'ont pas prévu le coup. Si ça ne marche pas, je suis juste content d'avoir essayé et je passe à autre chose. Je n'essaie même pas d'imaginer la capacité "manquante". Le joueur doit s'inventer des capacités, mais ne doit pas en inventer à son personnage.

J'aimerais pouvoir dessiner un plan sur la carte et pouvoir le partager en temps réel avec des coéquipiers IA ou humains qui suivraient ce plan

J'aimerais bien tirer au juger de manière plus sécurisée, sans avoir à me mettre à découvert. Tuer n'est pas l'objectif de la suppression, il faut juste les empêcher d'avancer et de viser. Les alliés se chargent du reste.

J'aimerais aussi que les couverts soient plus intéressants à utiliser, et avec plus de formes qu'un simple cube derrière lequel se cacher. J'aimerais aussi qu'on puisse plus souvent les créer ou les déplacer soi-même. Il me semble idiot que les personnages dans les jeux vidéo, en solo ou en multi, n'aient aucune volonté de survie, et qu'une position fortifiée se résume seulement à des sacs de sables entassés. Qu'on puisse vider un camp fortifié parce que personne n'a pensé à faire des meurtrières ou une protection à l'avant de sa mitrailleuse, c'est super idiot.

Air control (comme dans Unreal Tournament) / grimper sur un mur / tirer à travers les murs / killcam (évite d'avoir trop de campeurs)

Je me suis dit que ce serait bien de pouvoir communiquer avec les autres joueurs autrement que depuis le chat. Par la voix c'est sympa mais quand les joueurs parlent une autre langue c'est relou. Il faudrait faire des petites bulles ou inventer un truc nouveau plus sympa. Voilà !

Franchement, rien ne me viens à l'esprit, c'est justement aux Game designers de penser à ça en fonction du gameplay de leur jeu.

Pouvoir modifier son environnement. Ou que l'environnement change de lui même en cours de jeu pour offrir de nouvelles possibilités. Le meilleur exemple est la série Battlefield sous le moteur Frostbite, pour la destruction de l'environnement. Plus particulièrement l'épisode 4, qui intégrait "Levolution". Bidonville aux rues étroites qui favorisait le combat rapproché et était fatal aux véhicules, transformé en bataille semi-navale après la destruction d'un barrage engendrant une montée brutale des eaux. J'aimerais clairement retrouver de genre de fonctionnalités dans d'autres jeux.

Règles d'évolution des actions

Ce document est un document référence qui m'a aidé à concevoir les règles générales et les actions de mes niveaux de jeu expérimentaux. Par contrainte de temps toutes les actions et leurs caractéristiques n'ont pas été programmées.

I) Fonctionnement du système d'attaque

- Toute les attaques sont de type physique sauf exception avec une interaction joueur / environnement entraînant la capacité de lancer des compétences magiques
- La séparation entre l'ennemi et le joueur est découpée en trois espaces :
 - Les attaques au corps à corps ou attaques à courte distance : toutes attaques lancées dans un périmètre inférieur à 300
 - Les attaques à mi-distance : toutes attaques lancées avec une distance située entre 300 et 1500
 - Les compétences à longue distance qui sont réservées aux compétences défensives comme le "heal" et le "buff" : distance supérieure à 1500 et inférieure à 2000
 - Le joueur peut lancer des compétences magiques lorsqu'il est à proximité d'éléments naturels.
Ex : source d'eau, torche, boue, source électrique, etc...
- Les compétences magiques demandent obligatoirement au joueur de jouer avec son environnement.
- Les types d'attaques varient en fonction de l'angle d'attaque
 - Si le joueur attaque frontalement au corps à corps : capacités liées à l'agressivité + capacités d'attaque brute + capacités de protection + capacités magiques (si proche d'un élément : distance 300)
 - Si le joueur attaque sur les côtés au corps à corps : capacités d'attaque brute + capacités de protection + capacités magiques (si proche d'un élément : distance 300)
 - Si le joueur attaque dans le dos : capacités sournoises + capacités magiques (si proche d'un élément : distance 300)
 - Si la distance est inférieure à 500 le joueur peut lancer son grappin pour attirer le/les ennemi(s) (sauf si l'ennemi est trop gros pour être attiré) ou se projeter vers l'adversaire (dans le cas où l'ennemi est trop gros)
 - Si le joueur joue à mi-distance ou longue distance l'angle d'attaque importe peu
 - Les capacités défensives comme les boucliers peuvent se lancer aussi bien au corps à corps qu'à distance (peu importe la distance)
 - Les capacités purement furtives comme l'invisibilité, la pose de pièges ou encore l'espionnage sont uniquement possibles lorsque le joueur n'est pas entraîné dans un combat, n'a pas été détecté ou n'est pas dans le champ de vision de l'ennemi (voir la description des compétences)
 - Lors d'une action contextuelle : un symbole apparaît sur l'ennemi. Le joueur doit utiliser la touche d'action pour effectuer la compétence.
 - L'interface du joueur est composée d'une barre de vie, d'une barre de mana et d'une barre d'endurance
 - Les attaques magiques consomment des points de mana
 - Les esquives et mouvements de déplacement particuliers ou de parade consomment de l'endurance

- Lorsque le joueur vise un ennemi la jauge de vie de l'ennemi ciblé apparaît en haut à droite de l'écran

II) Utilisation d'objets / de décors

- La détection de l'utilisation d'un objet se fait via un raycast invisible
- Lorsque le joueur touche l'objet avec le raycast une interface apparaît pour signaler l'action avec l'objet / décor.
 - L'interface apparaît si le joueur se tient dans le champ d'action de l'objet / décor
 - Si le joueur n'est pas dans le champ d'action de l'objet / décor il ne peut pas l'utiliser
 - Exception pour certains décors qui ne nécessitent pas que le joueur les vise pour les utiliser mais simplement d'être à proximité. Exemple : être proche d'une torche pour lancer des boules de feu. (voir fonctionnement des capacités)
 - Exception pour les décors qui peuvent être détruits : le joueur a simplement besoin de tirer dessus.
- Les projectiles du joueur ont une distance maximum de 2000
- Tout décor / objet au delà de cette distance ne sera pas frappé par les projectiles

III) Fonctionnement des compétences

Lesquive :

- Deux types d'esquive : courte et longue respectivement sur les touches alt et espace
- Esquive courte : le personnage esquive en sautant dans la direction de mouvement du personnage (exemple : touche D + touche alt -> esquive courte sur le côté)
- Esquive longue : roulade dans la direction de mouvement du personnage (exemple : touche D + touche espace -> esquive sur le côté)
- Esquive longue : diminue l'endurance de 30
- Esquive courte : diminue l'endurance de 10

La téléportation :

- Evolution de l'esquive
- Liée au type d'esquive
- Si renforcement Esquive > 0.6 alors Téléportation disponible
- Le type de téléportation dépend du type d'esquive le plus utilisé
- Si l'esquive courte est la plus utilisée alors la téléportation téléporte le joueur dans le dos de l'ennemi
- Si l'esquive longue est la plus utilisée alors le joueur aura une téléportation sur une longue distance
 - La longueur de téléportation "longue distance" dépend de la distance d'attaque
 - Par défaut la longueur de téléportation "longue distance" est de 1000
 - Par défaut la longueur de téléportation "moyenne distance" est de 500
 - Cooldown 30 secondes

Le sprint :

- Le joueur peut sprinter en appuyant sur la touche shift
- Sprinter diminue la jauge d'endurance de 1 point par seconde
- Lorsque le joueur n'a plus d'endurance il ne peut plus sprinter
- Récupération d'endurance : 10 points par seconde

Charge

- Disponible si renforcement Sprint et renforcement de Attaque physique $cac > 0.6$
 - Remplace le sprint lorsque le personnage vise un ennemi
 - Si le joueur appuie sur la touche "sprint" le personnage fonce sur l'adversaire ciblé (distance maximale de 500)
 - Tout ennemi percuté est projeté en arrière sur une distance de 200 et subit des dégâts
 - Ne projette pas les ennemis lourds (ex : boss)
 - Dans le cas d'un ennemi lourd le personnage s'arrête au corps à corps de l'ennemi et lui inflige des dégâts
 - Si renforcement "Charge" > 0.8 alors le combo Charge Saut est disponible pour déclencher le combo appuyer sur la touche espace à la fin de la charge
 - Si combo enclenché le personnage saute sur les ennemis projetés.
 - Ne peut se déclencher que sur les ennemis projetés
- inflige des dégâts supplémentaires

Grimper :

- De même que pour le sprint, la vitesse varie en fonction du renforcement de l'action
- Plus l'action grimper est renforcée plus la vitesse de grimpe est élevée
- Si renforcement > 0.7 alors le joueur peut effectuer des sauts sur les murs
- Lorsque le joueur est agrippé à un mur il peut se laisser glisser en tapant sur la touche "E".
- Pas de cooldown

Courir sur les murs :

- Evolution de l'action Grimper
- Lié au sprint
- Si renforcement "Grimper" $= < 0.3$ alors Courir sur les murs non disponible
- Si renforcement "Grimper" et "Sprint" > 0.5 et $= < 0.7$ alors Courir sur les murs est disponible mais seulement sur un axe vertical
- Si renforcement "Grimper" > 0.7 et renforcement "Sprint" > 0.7 alors Courir sur les murs est disponible verticalement et horizontalement
- La distance de course sur les murs varie en fonction du renforcement du sprint
- Plus le sprint est renforcé plus la distance parcourue en courant sur les murs est longue
- Pas de cooldown

Le Grappin :

- Evolution de Grimper

-
- Disponible si renforcement Grimper ≥ 0.7
 - Utilisable uniquement hors combat
 - Il permet au joueur de bouger des éléments du décor qui sont en l'air
 - Permet de se suspendre dans les airs
 - Possibilité de se balancer et de se raccrocher à une autre prise de Grappin
 - Le joueur doit viser l'objet ou la prise de grappin pour s'y accrocher et interagir avec.
 - Si le joueur se raccroche à une autre prise il n'a pas besoin de viser la nouvelle prise mais juste de lancer son grappin qui doit s'accrocher automatiquement
 - Les interactions avec le grappin se situent en hauteur. Le joueur doit donc au préalable grimper
 - Pas de cooldown

La tyrolienne :

- Evolution du Grappin
- Liée à la Distance d'attaque
- Disponible si renforcement Grappin > 0.6
- Utilisable uniquement hors combat et en l'air (en hauteur)
- Le joueur vise le point d'accroche et enclenche la touche d'action pour créer la tyrolienne
- Pour utiliser la tyrolienne : touche action une fois celle ci posée
- Si renforcement "Attaque physique corps à corps" > 0.6 le joueur peut lâcher la tyrolienne au dessus d'un ennemi pour lui tomber dessus. Si l'ennemi est un boss : enchaînement QTE. Si l'ennemi n'est pas un boss : mort instantanée de l'ennemi
- Pas de cooldown

Grappin : attirer l'ennemi

- Lié à la distance d'attaque / l'angle d'attaque n'importe pas
- Disponible si le joueur est à une distance de 500 ou moins d'un adversaire
- Attire l'adversaire avec son grappin et inflige un étourdissement de 2 secondes
- Ne peut pas attirer les boss (pas d'étourdissement)
- Si utilisation sur le boss, c'est le joueur qui est attiré vers le boss
- Si renforcement > 0.5 : attire l'ennemi visé et tout autre ennemi à une distance égale ou inférieure à 100 de celui visé.
- Pas de cooldown

Grappin : Evasion

- Evolution de Grappin : attirer l'ennemi
 - Lié à la distance et à l'angle d'attaque : nécessite d'être au corps à corps (distance < 100) et face à l'ennemi
 - Disponible uniquement le combat engagé
 - Degré 1 si renforcement "Attirer ennemi" $> 0.3 < 0.6$ ou renforcement "Grimper" $> 0.3 \leq 0.5$: le joueur lance son grappin sur un point d'accroche et s'extraît des mains de l'ennemi. Le joueur doit viser le point d'accroche et taper sur la touche correspondante.
 - Degré 2 si renforcement "Attirer ennemi" ≥ 0.6 ou renforcement "Grimper" ≥ 0.8 : Le joueur
-

crée un point d'accroche sur un élément du décor et y accroche son grappin pour s'extraire

- Il doit viser le décor voulu et appuyer sur la touche correspondante à l'action. L'extraction est automatique

- Après l'extraction le joueur reste accroché au mur jusqu'à une nouvelle action (glisser, grimper ou jugement)

- Peut être un point en hauteur
- Distance de lancement limitée à 1500
- Cooldown 60 secondes

Insulte / Séduction :

- Disponible hors combat à une distance inférieure ou égale à 500
- Attire l'attention de l'ennemi qui attaque le joueur
- Si renforcement > 0.6 charme l'ennemi qui s'approche du joueur mais ne l'attaque pas. Dure 5 seconde

- Cooldown 30 secondes

Grappin : forcer aggro

- Evolution de Grappin : attirer ennemi et "Insulte / Séduction"
- Lié à la distance d'attaque (l'angle d'attaque n'importe pas)
- Degré 1 disponible si renforcement "Insulte / Séduction" > 0.4
- Degré 2 disponible si renforcement "Grappin : attirer ennemi" > 0.6 et renforcement "Insulte / Séduction" > 0.6

- Disponible uniquement le combat engagé et le joueur à mi-distance.
- Degré 1 : retourne l'ennemi (boss inclu) vers le joueur et le force à l'attaquer
- Degré 2 : retourne l'ennemi et l'attire directement devant le joueur. Si l'ennemi est un boss attire le joueur vers le boss

- Cooldown 60 secondes

Jugement : Compétence Ultime

- Evolution de Grappin : Evasion et Grappin : forcer aggro
- Disponible uniquement si renforcement "Grappin : Evasion" > 0.7, renforcement "Grappin : forcer aggro" > 0.7, renforcement "Grappin : attirer l'ennemi" > 0.5 et Renforcement "Attaque physique corps à corps" > 0.7

- Combo avec la compétence Grappin : Evasion
- Lorsque le joueur est accroché au mur après Grappin : Evasion il a 2 seconde pour enclencher l'action jugement

- Jugement : le joueur saute sur le boss (pas de limite de distance) et déclenche des QTE. Si réussite : engendre d'énormes dégâts. Si échec le joueur est repoussé

- Cooldown 180 secondes

Bouclier

- Il n'y a pas de distance requise pour activer le bouclier

-
- Le joueur doit obligatoirement avoir engagé un combat
 - Pour activer la compétence le joueur doit être face à l'ennemi et maintenir le clic gauche et droit en même temps
 - Dure 5 secondes
 - Le joueur ne peut pas faire d'autres actions pendant la compétence
 - Le joueur ne peut pas bouger pendant la compétence
 - Cooldown 60 secondes
 - Degré 2 : si renforcement Bouclier > 0.6 alors le bouclier renvoie à l'ennemi la somme des dégâts qu'il endosse pendant l'activation

Parade :

- Pour activer la parade Le joueur doit être au corps à corps
- Pas d'angle d'attaque requis
- Consomme 10 points d'endurance par seconde
- Lorsque le joueur fait une parade il ne peut pas faire d'autres actions mais il peut se déplacer
- La vitesse de déplacement est réduite de 50%
- La parade bloque tous les dégâts exceptées les AOE
- Pas de cooldown
- Degré 2 : si renforcement Parade > 0.6 alors le joueur peut enchaîner une représaille : coup porté à la suite d'une parade => combo

Armure Naturelle :

- Evolution des compétences "Bouclier" et "Parade"
 - Disponible uniquement si renforcement Bouclier et renforcement Parade > 0.6
 - Permet au joueur de changer la constitution physique de son personnage en copiant un matériaux visé (Exemple : pierre, métal)
 - En changeant de propriété le joueur acquiert les capacités du matériau pendant 5 secondes
 - Cooldown 120 secondes
- #### Propriété matériaux :
- Pierre : Encaisse tous les dégâts (peu importe le type et la provenance : AOE) sans dépasser 60% des points de vies totaux du joueur. Si les dégâts dépassent 60% l'armure est brisée
 - Métal : Ignore 50% des dégâts et renvoie 50% des dégâts reçus

Bufs (améliorations de capacités)

- Les Bufs sont disponibles uniquement à longue distance
 - Une fois activés le joueur bénéficie du buff pendant x secondes.
 - Pour activer un buff le joueur doit être proche un élément (eau, feu, terre, air : distance =< 200)
 - L'effet engendré dépend de l'élément qui a permis l'activation du buff
 - Le joueur peut cumuler 2 buffs et combiner les effets
 - Le temps d'effet par défaut dure 5 secondes
 - Plus le renforcement Buff est élevé plus l'effet est amplifié (puissance du buff) et dure longtemps (ajout maximum 20 secondes)
-

-
- Cooldown 30 secondes
 - Les cooldowns ne sont pas communs

Debuffs (malus ajouté à l'ennemi) :

- Les debuffs sont disponibles si Renforcement "Attaque magique au corps à corps" > 0.5
- Les debuffs sont activables uniquement au corps à corps (distance =< 100) et dans le dos de l'ennemi
- L'effet engendré dépend de l'élément qui a permis l'activation du debuff
- Le temps d'effet par défaut est de 10 secondes
- Plusieurs debuffs peuvent se cumuler.
- Plus le renforcement Debuff est élevé plus l'effet est amplifié (nombre d'ennemis / puissance du debuff) et dure longtemps (ajout maximum 10 secondes)
- Cooldown 60 secondes
- Les cooldowns ne sont pas communs

Soin :

- Sort disponible sans prérequis d'action
- Pour utiliser un soin le joueur doit être à longue distance de l'ennemi
- Utilisable hors combat
- Un soin rend 20% de la santé du joueur
- Cooldown 10 secondes
- Plus le Renforcement de Soin est élevé plus le pourcentage de santé rendu est élevé (ne peut pas dépasser 50% de santé)

Drainage :

- Evolution de Soin
- Disponible uniquement si Renforcement "Soin" > 0.5 et Renforcement "Compétences magiques longues distance" > 0.5.
- L'action drainage peut être exécutée à toutes distances inférieures à 2000 du personnage ciblé
- Le joueur doit cibler un animal vivant ou un ennemi pour aspirer sa vie
- La vie rendue au joueur dépend de l'animal drainé (Exemple : Lapin -> 5%, Cerf -> 20%)
- Cooldown 30 secondes

Bind : Compétence ultime

- Disponible uniquement si Renforcement "Soin" > 0.7, Renforcement "Drainage" > 0.7 et -
- Renforcement "Attaque magique longue distance" > 0.7
- Le joueur lie sa vie à un élément naturel du décor (arbre, animal, etc), tant que cet élément est toujours vivant (non détruit), le joueur peut ressusciter à la place de cet élément en cas de mort
- Si le joueur ressuscite l'élément lié est détruit
- Si l'élément est détruit par un ennemi ou le joueur (exemple : lors d'un combat), le joueur devra de nouveau placer son lien sur un autre élément.
- Une marque doit apparaître sur l'élément lié. La marque est visible au travers des obstacles.
- Cooldown 2 minutes

Approche furtive :

- Disponible dès le départ
- Le personnage se déplace sans faire de bruit
- Permet de s'approcher furtivement de l'ennemi sans attirer l'attention (sauf si le personnage entre dans le champ de vision de l'ennemi)
- Le joueur doit garder la touche "C" enfoncée pour se déplacer furtivement
- Si la touche est relâchée l'approche furtive est désactivée.
- Activable sans restriction de distance et de positionnement
- Pas de cooldown

Désarmement :

- Disponible uniquement si renforcement "Approche furtive" > 0.5
- Activable uniquement au corps à corps et dans le dos d'un ennemi armé.
- Pour activer : appuyer sur la touche "E"
- Lorsque activé : l'ennemi perd ses armes pendant 10 secondes -> l'ennemi est stupéfait pour 5 secondes puis attaque à mains nues pour les 5 dernières secondes avant de posséder à nouveau ses armes.
- Pas de cooldown

Faire le Mort :

- Disponible uniquement si Renforcement "Approche furtive" > 0.5 et Renforcement "Attaque physique Cac" > 0.5
- Deux situations d'activation : Combat engagé et hors combat
- Les cooldowns ne sont pas communs aux deux situations

Combat engagé :

- ne peut être activé qu'au corps à corps de l'ennemi
- lorsque activé l'ennemi se désintéresse du personnage joueur : réinitialisation de l'agressivité du joueur
- cooldown de 60 secondes

Hors combat :

- lorsque la compétence est activée le personnage s'effondre au sol en faisant du bruit qui attire
- les ennemis à proximité (périmètre de 500 autour du personnage joueur)
- lorsqu'un ennemi est attiré au corps à corps le joueur peut enchaîner avec une attaque au corps à corps qui tue l'ennemi d'un seul coup
- cooldown 60 seconde

Disparition :

- Disponible uniquement si Renforcement "Attaque physique cac" et "Cover" > 0.5.
- Deux situation d'activation de la compétence : Combat engagé et hors combat
- Les cooldowns ne sont pas communs aux deux situations

Combat engagé :

Ne peut être activé qu'au corps à corps, face à l'ennemi

-
- Jette une bombe fumigène au sol qui envoie un écran de fumée et désoriente l'ennemi
 - Périmètre du nuage de fumée de 100 autour du lieu d'impact de la bombe.
 - Les ennemis présents dans le nuage de fumée ne peuvent plus attaquer
 - Le nuage de fumée se dissipe au bout de 15 secondes
 - Cooldown de 60 secondes

Hors combat :

- Lance une bombe fumigène sur un ennemi qui envoie un écran de fumée et le désoriente
- Même effet et même périmètre que lors d'un combat engagé
- Tout autre ennemi présent dans le champ d'action de la bombe est désorienté
- Le nuage de fumée se dissipe au bout de 15 secondes
- Cooldown de 60 secondes

Empoisonnement :

- Disponible uniquement si Renforcement "Attaque physique cac" ou "Attaque physique Mid" > 0.5 et Renforcement "Approche furtive" ou "Cover" ou "Cachette" > 0.5.
- L'empoisonnement se déroule en deux temps
- 1er : Le personnage empoisonne son arme lorsqu'il est à proximité de la plante correspondant à ses besoins (1 charge poison sur l'arme)
- 2nd : Le personnage attaque l'ennemi au corps à corps ou à distance. Si l'arme est empoisonnée l'attaque inflige le poison.
- Le poison se dissipe après 10 secondes d'effet.
- Différents poisons : paralysie -> plante verte, intoxication -> plante violette, brûlure -> plante rouge
- Poison paralysie : empêche l'ennemi de bouger pendant 10 secondes. Si l'ennemi est un boss, ralentit ses déplacements de 50%
- Poison intoxication : inflige x points de dégâts toutes les 2 secondes pendant 10 secondes et rend l'ennemi fou. L'ennemi fou attaque tout personnage (joueur ou non) à proximité.
- Poison brûlure : inflige x points de dégâts toutes les 2 secondes pendant 10 secondes. Peut s'utiliser deux fois sur différents ennemis (1 charge = 2 utilisations)
- Si renforcement de empoisonnement > 0.8 alors l'arme peut porter 2 charges de poison.
- Si les deux charges ont le même poison : double le temps et les dégâts infligés par le poison
- Si les charges ont différents poison : combine les effets des deux poisons mais le temps avant dissipation reste de 10 secondes. Exemple brûlure + paralysie : paralyse l'ennemi et inflige x points de dégâts toutes les 2 secondes pendant 10 secondes. Peut être utilisé deux fois.

Accrochage :

- Disponible uniquement si Renforcement "Attaque physique cac" ou "Attaque magique cac" > 0.6
 - Le personnage joueur soulève l'ennemi attaqué du sol pendant 5 secondes
 - L'ennemi soulevé ne peut plus faire d'action
 - Le personnage joueur peut attaquer l'ennemi soulevé avec l'attaque de la seconde main mais ne peut pas enclencher de seconde compétence
 - Si Renforcement de "Accrochage" > 0.6, déclenche le combo : bouclier humain
-

-
- le joueur peut utiliser l'ennemi comme bouclier humain en appuyant sur la touche "E" après avoir utilisé accrochage. Dure 5 secondes
 - Si Renforcement de "Accrochage" > 0.8, déclenche le combo : projection
 - Lorsque le joueur utilise accrochage il peut le projeter dans la direction ciblée en appuyant sur le clic gauche de la souris
 - Ne peut se déclencher que pendant les 5 secondes du bouclier humain
 - Ne fonctionne que sur les humanoïdes
 - Si un ou plusieurs ennemis sont touchés par l'ennemi projeté, ils tombent au sol

Utilisation décor :

- Disponible uniquement si Renforcement "Attaque physique cac" ou "Attaque magique cac" > 0.6
- Le personnage arrache un élément du décor utilisable et frappe l'ennemi avec
- L'élément du décor peut également être envoyé sur l'adversaire si la distance d'attaque est inférieure à 1500
- Assomme l'ennemi pendant 3 secondes
- Pas de cooldown

Cover :

- Compétence disponible dès le départ
- Si le joueur est situé à une distance inférieure ou égale à 300 et qu'il est face à un cover il peut appuyer sur la touche "E" pour se mettre automatiquement à couvert (système gears of war)
- A couvert le joueur peut se déplacer le long du cover (gauche / droite)
- Si le joueur fait une roulade ou appuie sur la direction opposée au cover (déplacement arrière), il se détache automatiquement du cover
- Pas de cooldown

Cachette :

- Compétence disponible dès le départ
 - Lorsque le joueur est proche d'une cachette (en collision) il peut se cacher à l'intérieur en appuyant sur la touche "E"
 - Nécessite d'être à moyenne distance des ennemis (distance supérieure à 500)
 - Le joueur ne peut se cacher d'un ennemi au corps à corps. Si le joueur entre dans une cachette en étant en combat avec un ennemi, celui-ci le tir de sa cachette et engendre un stun (étourdissement) du personnage joueur. Durée du stun : 3 secondes
 - Lorsqu'un joueur est caché, les ennemis ne peuvent pas le détecter
 - Si le joueur attaque un ennemi situé à proximité de la cachette il devient visible et détectable
 - Si combat engagé, le joueur doit d'abord s'éloigner de l'ennemi avec une distance de 1000 au minimum ou s'éloigner en passant derrière des décors. Point d'interrogation qui apparaît au dessus de la tête de l'ennemi pour montrer qu'il perd la position du joueur et que celui-ci peut se cacher
 - Ne fonctionne pas sur les animaux et créatures (monstres) agressifs proches.
 - Pas de cooldown
-

Leurre :

- Compétence “Leurre” disponible si Renforcement “Cachette” et Renforcement “Attaque Physique Cac” > 0.5
- Nécessite d'être caché pour lancer la compétence
- Lorsque le joueur utilise le Leurre, une réplique de son personnage apparaît à une distance de 300 dans la direction ciblée
- Le leurre attire les ennemis présents dans la zone autour de lui (distance de 200 autour du leurre)
- Le leurre est destructible et dure 20 secondes
- Les ennemis attaquent le leurre
- Lorsque les points de vie du leurre tombent à 0 le leurre disparaît
- Si la durée du leurre est terminée alors celui-ci disparaît même si ses points de vie sont supérieur à 0.
- Si renforcement cachette > 0.8, le leurre activé est agressif et attaque les ennemis présent dans sa zone (distance de 200 autour du point d'apparition)
- Le leurre agressif possède les mêmes attaques que le personnage joueur
- Les attaques du leurre agressif infligent 50% des dégâts du personnage joueur

Invisibilité :

- 3 degrés d'invisibilité
- Degré 1 (camouflage) : disponible si Renforcement “Cover” et Renforcement “Cachette” > 0.5 et =< 0.7
- Degré 1 : nécessite que le personnage soit immobile (debout ou accroupi) et collé à un élément du décor sans le dépasser. Exemple : debout derrière un cover ne fonctionne pas il faut être accroupi, de même si le joueur dépasse d'un mur.
- Degré 1 : lorsque les conditions sont remplies la texture du personnage prend celle du décor auquel il est collé
- Degré 1 : rend le personnage indétectable pour les ennemis
- Degré 1 : fonctionne hors combat et en combat engagé
- Degré 1 : si combat engagé, le joueur doit d'abord s'éloigner de l'ennemi avec une distance de 1000 au minimum ou s'éloigner en passant derrière des décors. Point d'interrogation qui apparaît au-dessus de la tête de l'ennemi pour montrer qu'il perd la position du joueur et que celui-ci peut se camoufler
- Degré 1 : pas de restriction de temps passé camouflé.
- Degré 1 : cooldown 30 secondes
- Degré 2 : le joueur devient invisible dans les zones d'ombre du décor
- Degré 2 : si le joueur passe d'une zone d'ombre à une zone éclairée en étant invisible, la zone éclairée retire l'invisibilité du joueur.
- Degré 2 : disponible si Renforcement “Invisibilité” > 0.7 et =< 0.8
- Degré 2 : Lorsque le joueur est hors combat il peut devenir invisible dans les zone d'ombre du décor
- Degré 2 : Si le combat est engagé, nécessite les mêmes conditions que le degré 1 pour pou-

voir utiliser la compétence

- Degré 2 : pas de restriction de temps passé invisible dans les zones d'ombres
- Degré 2 : cooldown 30 secondes
- Degré 3 : disponible si Renforcement "Invisibilité" > 0.8
- Degré 3 : Le joueur peut être invisible dans toutes zones d'une map, éclairée ou non.
- Degré 3 : Lorsque le joueur est hors combat il peut devenir invisible dans les zones éclairées et non éclairées
- Degré 3 : Si le combat engagé le joueur ne peut utiliser l'invisibilité dans les zones lumineuses. Il doit obligatoirement utiliser les zones d'ombres pour utiliser sa compétence (même conditions d'activation que pour le degré 2)
- Degré 3 : le joueur ne peut être invisible dans une zone éclairée que pendant 20 secondes
- Degré 3 : cooldown 30 secondes
- Les différents degrés s'accumulent. Si renforcement cover et cachette > 0.8 le joueur peut se camoufler et devenir invisible
- Le camouflage et l'invisibilité ne fonctionnent pas sur les animaux et créatures (monstres) agressifs proches.

Espionnage :

- Compétence espionnage disponible si Renforcement "Cover", "Cachette" et "Invisibilité" > 0.7
- Nécessite d'être hors combat
- Pour activer la compétence maintenir la touche "shift"
- Lorsqu'elle est activée, la compétence espionnage permet de détecter les ennemis situés à une distance de 2500 autour du joueur (le décor apparaît en négatif et les ennemis en silhouette rouge)
- Détecte les ennemis au travers des murs
- Si la touche "shift" est relâchée, un contour rouge reste affiché autour des ennemis

Entrave :

- Disponible si Renforcement "Attaque physique mid" > 0.6
- Le joueur tire des flèches reliées à une corde qui permettent d'accrocher un ennemi à une partie du décor
- Le joueur doit d'abord tirer sur l'ennemi puis sur le décor
- Pas d'emplacement spécifique du décor pour accrocher l'ennemi
- Peut être des accroches au sol, murs, ou plafond
- Les ennemis peuvent détruire ces liens
- Le joueur peut lier l'ennemi avec plusieurs liens
- Plus il y a de liens plus l'ennemi met de temps à s'en défaire

Magie Électrique :

- Utilisable si joueur est proche d'une source électrique
 - Clic gauche : envoie un éclair sur la cible et lui inflige des dégâts
- temps d'incantation : instantané

pas de cooldown

- Clic droit : fait tomber la foudre sur la cible et la paralyse pendant 5 secondes

temps d'incantation 1 seconde

- cooldown 10 secondes

Magnétisme :

- Disponible si Renforcement "Magie Electrique" > 0.6
- Cibler un aimant pour activer
- Lorsque la capacité est activée attire et immobilise l'ennemi le plus proche.
- L'ennemi ne peut se défaire de l'attraction
- Pas de cooldown

Réanimation :

- Disponible si Renforcement "Magnétisme" > 0.8
- Invoque la foudre et permet de réanimer un allié décédé
- Temps d'incantation 5 secondes
- Pas de cooldown
- Nécessite d'être au corps à corps avec l'allié

Magie du feu :

- Utilisable si le joueur est à proximité d'une source de chaleur
- Clic gauche : envoie une boule de feu qui inflige beaucoup de dégâts

temps d'incantation 1.5 secondes

- pas de cooldown

- Clic droit : place une boule de feu explosive sur la cible qui inflige des dégâts à chaque seconde pendant 5 seconde.

- A la fin du temps la boule explose et inflige de lourds dégâts à la cible ainsi qu'aux ennemis présents dans un périmètre de 100 autour d'elle

- temps d'incantation 2 secondes
- cooldown 15 secondes

Magie Eau :

- Utilisable si le joueur est à proximité d'une source d'eau
- Clic gauche : englobe la cible (fonctionne également sur les objets) dans une bulle d'eau la soulevant du sol et l'empêchant d'avancer pendant 5 secondes
- lancement instantané
- cooldown 5 secondes

Clic droit : gèle l'eau (ou vapeur) ciblée.

- Si la cible est englobée dans la bulle d'eau, gèle la bulle et la fait retomber violemment au sol en infligeant des dégâts.

- Si la cible est une surface d'eau au sol, gèle la surface et l'ennemi présent dans la zone pendant 3 secondes et l'empêche d'avancer. Après les 3 secondes l'eau fond.

-
- Si la cible est de la vapeur : crée des projectiles qui sont envoyés dans la direction de propagation de la vapeur.
 - Lancement instantané
 - pas de cooldown

Création de décor :

- Disponible si Renforcement "Magie Eau" > 0.6
- Permet de générer du décor gelé à un emplacement du décor spécifique
- Cet emplacement est visible par une base de l'objet gelée et un shader semi transparent montrant le décor qui peut être créé
- Ce décor peut être ensuite détruit, ou fondu, etc... Exemple créer des pics de glace pour les faire tomber sur l'ennemi

Magie Esprit :

- Clic gauche maintenu : Torture la cible lui infligeant de lourds dégâts toutes les secondes pendant 3 secondes (tant que le clic est maintenu).
- La cible devient folle et attaque tout autre ennemi présent dans un périmètre de 50 autour d'elle pendant 6 secondes (tant que le clic est maintenu). A la fin de sa tourmente (ou si le sort est rompu) la cible reprend ses esprits et attaque le joueur.
- Une cible ne peut être torturée plus d'une fois
- Durant la torture le cri de la cible attire tous les ennemis présents dans un périmètre de 200 autour d'elle
- Sort de canalisation
- Durée maximum : 6 seconde
- Peut s'interrompre si le joueur relâche le clic
- pas de cooldown

Clic droit maintenu : aspire l'âme de la cible (ne fonctionne que sur les ennemis humanoïdes)

- Sort de canalisation
 - Durée maximum : 10 secondes
 - Sur cible vivante : inflige des dégâts toutes les secondes
 - Si la cible succombe absorbe son âme et remplit la jauge d'âme du joueur
 - Sur cible morte : absorbe l'âme et remplit la jauge d'âme
- le joueur ne peut accumuler plus de 3 âmes
- pas de cooldown

Fear :

- Disponible si Renforcement "Magie Esprit" > 0.6
- Nécessite de dépenser une âme pour utiliser cette compétence
- Effraie tous les ennemis présents dans un périmètre de 200 autour du joueur
- Les ennemis effrayés courent dans la direction opposée au joueur
- Dure 6 secondes

-
- A la fin des 6 secondes les ennemis reprennent leurs esprits et attaquent le joueur
 - Cooldown 30 secondes

Contrôle PNJ :

- Disponible si Renforcement “Fear” > 0.8
- 2 différents degrés
- Degré 1 : permet de contrôler la faune locale pour attaquer les ennemis ou de prendre le contrôle d’une créature
 - Si le joueur active la compétence sur la créature il en prend le contrôle, il peut alors se déplacer avec, attirer les ennemis, les effrayer (dépend de la créature contrôlée), ou les attaquer
 - Si le joueur active la compétence sur un PNJ ennemi, force la faune présente dans un périmètre de 200 autour de l’ennemi à l’attaquer.
- Degré 2 : permet de contrôler les PNJ humanoïdes
- Ne peut être déclenché qu’en étant hors combat
- Le sort est rompu si le joueur est attaqué
- La compétence peut être utilisée à une distance égale ou inférieure à 1000 de la cible
- Le champ d’action du sort lorsqu’une créature est contrôlée est de 2500 autour du joueur
- Si la créature contrôlée sort du champ d’action, le sort est rompu
- Durée du sort maximale : 60 secondes
- Le sort peut être rompu manuellement en appuyant sur la touche “E”
- Consomme une âme
- Si le joueur n’a pas d’âme en possession, le sort ne peut pas être utilisé

Télékinésie :

- Disponible si Renforcement “Magie Electrique”, “Magie Eau”, “Magie Feu” et “Magie Esprit” > 0.7
- Nécessite d’être à proximité d’une source de magie ou de consommer 3 âmes
- Permet de déplacer ou projeter des objets extrêmement lourds
- Cibler l’objet et maintenir clic gauche et clic droit puis déplacer la souris dans la direction voulue pour faire bouger l’objet
- L’objet bouge dans la direction pointée si aucun élément l’en empêche
- Tout ennemi touché par l’élément projeté meurt
- Durée du sort : 10 secondes
- Cooldown : 20 secondes

Projectile :

- Disponible dès le départ
- Le personnage jette une pierre dans la direction ciblée
- Lorsque la pierre retombe, le bruit attire les ennemis présents dans un périmètre de 200 autour de la pierre
- Les ennemis se déplacent jusqu’à l’emplacement où la pierre est tombée
- Pas de cooldown

Finish :

- Chaque type de finish est débloquent lorsque le renforcement de la compétence d'attaque associée est égal à 1. Exemple : renforcement attaque magique mid = 1, alors finish magique mid débloquent
- Lorsqu'il reste 10 % ou moins de vie à un ennemi, celui-ci à 25 % de chance de subir un finish
- Si l'ennemi peut être finish un symbole apparaît au dessus de sa tête.
- Le joueur doit être à la distance exigée par le finish et appuyer sur la touche "F". Exemple : être à distance moyenne de l'ennemi et en présence d'un élément naturel pour déclencher un finish magique à mi-distance
- Le finish fait apparaître une suite de QTE que le joueur doit réussir pour accomplir le finish
- Si le joueur manque un QTE le finish est raté et le joueur est repoussé par l'ennemi, mais il pourra retenter un nouveau finish
- Un finish réussi hôte le restant de vie à l'ennemi