

Cahier des Charges

EPITA - *Projet S2*



é

HERMITA

Produit par Epistars™

ANOUAR BELMDEJENNEH

ANTOINE BLUMENROEDER, BAPTISTE DURRINGER,
DYLAN DE ARAUJO

2022-2023

Table des matières

1	Préambule	1
1.1	Epistars™ : Les origines	1
1.2	Présentation de l'équipe	2
2	Introduction	2
2.1	Contexte du projet	2
2.2	Objet de l'étude	2
2.3	Nature du projet	3
2.4	État de l'art	3
3	Description détaillée	3
3.1	Synopsis	3
3.2	Menu et interface	4
3.2.1	Interface en jeu	5
3.3	Game design	6
3.3.1	Agencement de l'île du joueur	6
3.3.2	Description du premier niveau	7
3.3.3	Établissement du <i>housing</i>	8
3.4	Programmation	9
3.4.1	Multijoueur	9
3.4.2	Implémentation des combats	9
3.4.3	Intelligence artificielle et gestion des ennemis	10
3.4.4	Fonctionnement du <i>craft</i> et de l'inventaire	10
3.4.5	Système de sauvegarde	11
3.5	Visuel et animation	11
3.5.1	Identité du projet	11
3.5.2	Direction artistique	12
4	Conditions de réalisation	13
4.1	Délai de réalisation	13
4.2	Contraintes de développement	14
4.3	<i>Business model</i>	14
4.4	Suivi des réalisations	15
4.5	Communication client	16
4.5.1	Site web	16
4.6	Aspect technique	17
4.7	Conclusion	17
5	ANNEXE	18
5.1	Schémas UML	18
5.2	Glossaire	19
5.3	Sources	19

1 Préambule

Cette section aura la charge de vous présenter les origines de notre société mais aussi et surtout l'équipe qui s'attellera à mener à bien le projet de jeu que nous souhaitons réaliser.

1.1 Epistars™ : Les origines

La société Epistars a vu le jour lorsqu'un groupe d'amis, d'un naturel introverti, se sont rapprochés et ont créé de solides liens. Dès leur rentrée dans une école d'informatique réputée, ce groupe de quatre membres s'est vite formé durant leur journée d'intégration dans le but de devenir collègues pour réaliser un travail ambitieux sur un jeu vidéo. Le nom de la société fut choisi au cours de cette même journée pour une chasse au trésor en équipe, en fan de Star Wars épitéens ce nom était tout trouvé : Epistars (la fig. 1 illustre notre logo).

Cette proximité nous permet d'avoir une équipe soudée qui sait avancer sans contradictions. Dotée d'une forte motivation, nous avons la grande ambition de faire de la société Epistars™ un acteur majeur du monde vidéoludique dans les années à venir. Notre équipe est constituée de membres pourvus d'un riche ensemble de fortes individualités qui, de par leur complémentarité, forment un groupe solide et techniquement performant.



FIGURE 1 – Logo de la société Epistars™

1.2 Présentation de l'équipe

Notre équipe est composée des membres suivants :

- **BAPTISTE DURRINGER** (*Lead Game Designer*) : hautement compétent, fort de sa longue expérience dans le milieu vidéoludique.
- **DYLAN DE ARAUJO** (**Directeur Artistique**) : plein de talent en raison de sa passion dévorante pour toutes les formes d'art depuis son plus jeune âge.
- **ANTOINE BLUMENROEDER** (**Directeur d'Architecture**) : fêru de maths depuis toujours et doté d'une exceptionnelle intuition dans la modélisation de problèmes.
- **ANOUAR BELMEDJENNEH** (**Directeur Technique**) : fort de ses expériences acquises et de sa maîtrise des différents outils, est le plus qualifié pour diriger ce projet.

Nous aurons à cœur tout au long du processus de développement d'offrir une prestation de qualité.

2 Introduction

Ce cahier des charges a pour but de définir et présenter les modalités de réalisation du projet selon les critères voulus par le client. La mission qui nous a été confiée est la réalisation d'un jeu vidéo qui proposera une aventure solo ou en coopération jusqu'à quatre. Ce document fixe ainsi le contexte du projet, les impératifs de la mission mais également son organisation et tout ce que cela implique. *Les termes techniques employés ainsi que les sources de toutes les images qui ne sont pas réalisées par notre équipe sont disponibles en annexe.*

2.1 Contexte du projet

La société Epistars™ a choisi de mettre ses compétences au service de l'École Pour l'Informatique et les Techniques Avancées dans le but de mener à bien la réalisation d'un logiciel de loisir type jeu vidéo pour la validation de ce qui est appelé en interne « *Projet de S2* ». Cette opportunité nous permettra d'affiner nos compétences et notre technicité, mais aussi et surtout de renforcer notre capacité à travailler en équipe sur un projet complet.

2.2 Objet de l'étude

L'idée de la réalisation de ce jeu nous est venue rapidement dès que nous avons pris conscience de la liberté dont on dispose. N'ayant pas pour habitude de réaliser un projet d'une telle envergure, nous avons choisi d'être plus ambitieux car nous croyons en son potentiel. Nous faisons le choix de faire un jeu en deux dimensions plutôt qu'en trois, ce qui nous permettra de nous focaliser sur la création d'un contenu riche, au sein d'un univers original.

Nous voulons réaliser un jeu qui oppose action nerveuse et quiétude reposante car cette alchimie nous a séduit. Étant tous dotés d'un goût pour l'aventure et le défi, ce projet a trouvé écho dans nos cœurs. De plus, nous avons tous une certaine nostalgie pour les jeux de notre enfance. Ce sera l'occasion de retranscrire ce qui nous faisait vibrer plus jeunes tout en y ajoutant un regard plus mature. L'objectif est d'utiliser pleinement les différentes possibilités offertes par le format du jeu vidéo.

2.3 Nature du projet

Ce projet consiste en un jeu que nous réaliserons dans un style pixel art, il se présente sous la forme d'un monde de *fantasy* fait d'îlots flottants (avec de légères touches *steampunk*¹) en deux dimensions avec une vue isométrique². Notre jeu aura une composante *housing*³ et appartiendra au genre du RPG. Lorsqu'on lance le jeu, le joueur apparaît dans une zone qui lui servira de base et lui donnera accès aux autres îlots qui constituent son monde. Son objectif sera d'explorer son univers afin de réunir les ressources qui lui seront nécessaires pour faire évoluer sa base tout en ayant à combattre les créatures hostiles qui se dresseront sur sa route.

Hermita vise principalement à convaincre un public mature amateurs de jeux vidéo indépendants, avide d'aventure et d'exploration sans prise de tête. Notre jeu a pour objectif d'offrir une expérience apaisante et satisfaisante, sans que le joueur y passe tout son temps, nous voulons qu'il choisisse de le lancer pour passer un bon moment.

2.4 État de l'art

Le jeu que nous réalisons appartient au genre du RPG (*Role Playing Game*), très répandu dans le milieu du jeu vidéo et qui se définit par quatre principes fondamentaux : l'évolution du personnage, l'exploration, l'influence sur son monde et finalement les combats.

Le genre du RPG existe depuis maintenant plus de cinquante ans et ce bien avant l'arrivée du jeu vidéo. En effet, dès 1970, les jeux de rôles papier comme Donjon et Dragons ont inventé le concept. Ce n'est qu'en 1986, que sort le premier jeu vidéo du genre RPG : Dragon Quest sur Famicon (une console Nintendo) qui a eu un énorme succès à l'époque.

Hermita, n'a évidemment pas pour ambition de détrôner les titres établis que sont par exemple Zelda, Final Fantasy, The Elder Scrolls ou encore Dark Souls. Nous souhaitons vous présenter un jeu plus simple dans ses ambitions, offrant certes moins de possibilités qu'un triple A⁶ mais qui seront plus en adéquation avec nos objectifs et les contraintes qui nous incombent.

Pour imaginer notre jeu, nous tirons nos inspirations de différents univers pour pouvoir poser l'histoire, les décors, ou encore le gameplay. Ainsi des jeux tels que Terraria, Palia, Don't Starve (Together), Diablo ou encore League of Legends nous ont aidé à fixer le *game design*⁴ et le *level design*⁵. Au delà des jeux vidéo, notre projet est aussi librement inspiré d'oeuvres telles que les mangas Radiant et Sword Art Online, le film d'animation Le Château dans le Ciel ou même de diverses citations de Jacques Nteka Bokolo afin de fixer l'univers, l'histoire et les décors de notre jeu.

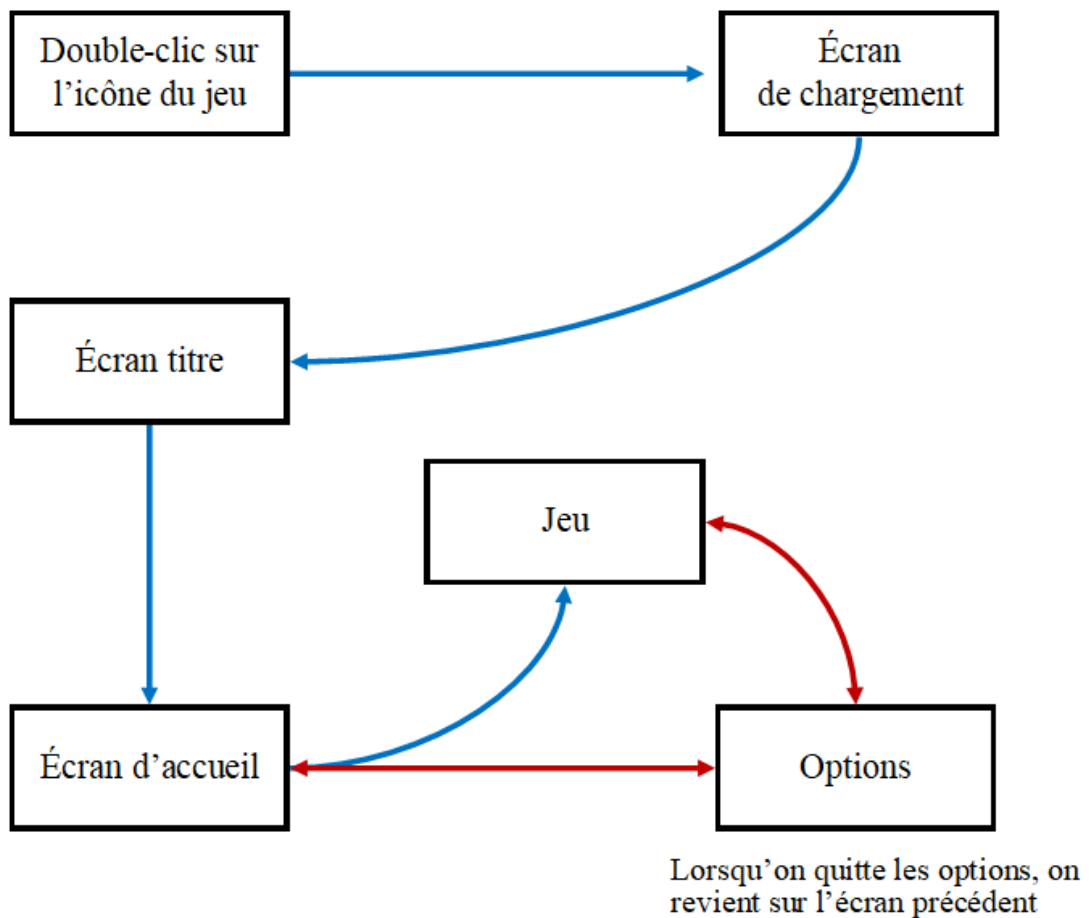
3 Description détaillée

3.1 Synopsis

Dans un monde de magie parsemé d'îles volantes, jouez un mage dont le vaisseau s'est écrasé à l'écart de toute vie intelligente. Accompagnez-le dans une aventure d'action et d'exploration pour l'aider à regagner sa vie normale en l'aidant à réparer son vaisseau au fur et à mesure qu'il progresse en faisant face à la solitude. . .

3.2 Menu et interface

Les menus et interfaces sont considérés comme l'intermédiaire entre l'utilisateur et le jeu, l'implémentation et les choix qui seront fait pour ces derniers auront une importance capitale notamment pour le plaisir de jeu. Le joueur sera constamment à leur contact durant les contextes d'utilisation. Ainsi, nous consacrerons une attention particulière à la création des différents écrans qui guideront le joueur lors de ses sessions de jeu (la fig. 2 illustre l'interaction entre les différents écrans).



A \longleftrightarrow B : On peut aller de l'écran A à l'écran B et inversement

A \longrightarrow B : On ne peut qu'aller de l'écran A à l'écran B

FIGURE 2 – Représentation de la relation entre les différents écrans

3.2.1 Interface en jeu

Dans Hermita, le joueur peut se déplacer sur huit axes et effectuer des *dashes*⁹ qui lui permettent d'esquiver des attaques, ceux-ci sont limités par la présence d'une barre d'endurance (affichée à proximité du personnage) qui se recharge progressivement lorsque le joueur ne *dash* pas pendant un certain laps de temps. Pendant qu'il sera en jeu, il faudra qu'il puisse facilement accéder aux options, à son inventaire et qu'il voit ses statistiques et les sorts qu'il a choisi d'assigner. Le joueur aura à sa disposition les éléments suivants :

- une **BARRE DE VIE (HP)** : subdivisée en plusieurs « sous-barres », de telle sorte à ce que le joueur ne puisse pas mourir d'un coup. En effet, il lui sera impossible de perdre plus d'une sous-barre complète. Cette barre de vie se régénère quand le joueur est hors combat pendant un certain temps ou à l'aide de potions et de sorts.
- une **BARRE DE MANA (MP)** : limite le nombre de sorts que peut envoyer le joueur et le contraint à utiliser des potions, ou à temporiser en attendant qu'elle se régénère dans un combat. Compte tenu du fait qu'une barre de mana peut vite s'avérer frustrante dans un jeu basé sur l'envoi massif de sorts, celle-ci sera implémentée de manière à ce que le combat reste satisfaisant. De plus le mana se régénère très vite hors combat.
- un **INVENTAIRE DE SORTS** : permet de savoir quels sorts sont disponible pour le joueur.

Il faut que tous les éléments évoqués soient accessibles et facilement lisibles en un coup d'œil car le jeu exigera de prendre rapidement des décisions fondées sur la prise en compte de nombreux facteurs, c'est dans cette optique que l'interface en jeu se voudra épurée autant que possible. De nombreuses optimisations seront certainement de rigueur tout au long du développement pour assurer un plaisir de jeu satisfaisant (la fig. 3 représente une première idée d'implémentation de cette interface).

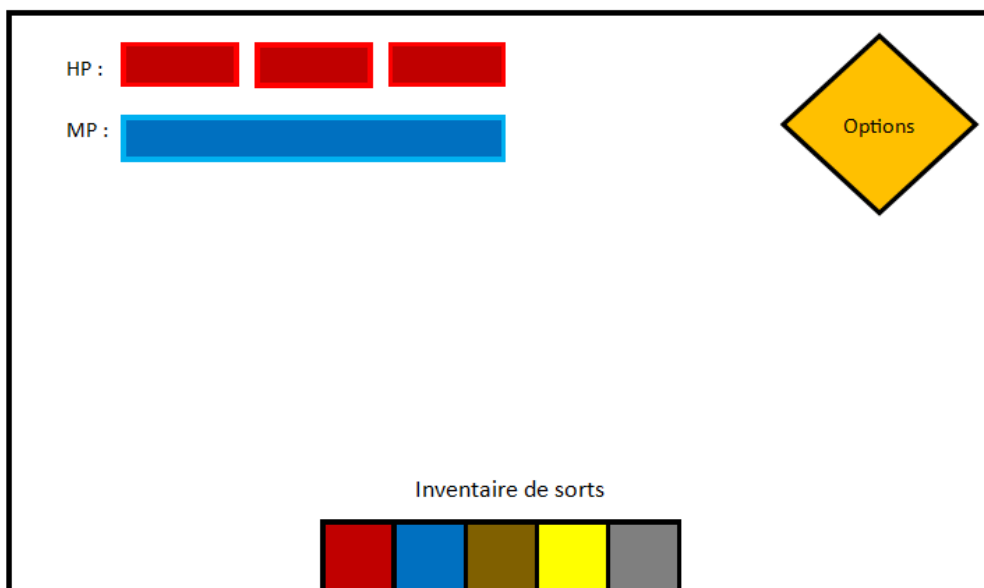


FIGURE 3 – Représentation de l'interface

3.3 Game design

3.3.1 Agencement de l'île du joueur

Hermita est basé sur la dualité entre magie et technologie. Le monde, isolé de toute vie intelligente depuis des siècles, a vu la nature reprendre ses droits, ainsi, elle constitue une part importante des zones à explorer. L'univers est forgé par les quatre éléments que sont : le feu, l'eau, la terre et la foudre, tout tournera autour de ces derniers. Il y a deux cœurs de jeu principaux :

- le **Housing** : sur l'île de base du joueur, dans laquelle il apparaît dès le début du jeu après la cinématique d'introduction.
- les **COMBATS** : sur les différentes îles à explorer, qui auront chacune une identité qui leur sera propre, accessibles à l'aide du portail présent sur l'île de base du joueur.

Le *level design*⁵ de l'île du joueur sera épuré pour mettre l'emphase sur la tranquillité qu'elle doit procurer en total contraste avec le monde extérieur. Le joueur y retrouvera :

- ses **BÂTIMENTS** : établi, atelier et laboratoire. Seul l'établi est directement disponible, les autres se débloquent à mesure que le personnage augmente de niveau.
- son **PORTAIL** : pour rejoindre d'autres îles et accéder au multijoueur.
- sa **MONTGOLFIÈRE** : écrasée, il aura pour objectif de la réparer afin de pouvoir repartir.

L'île sera d'une forme plutôt circulaire sur laquelle seront répartis les différents bâtiments évoqués ainsi que quelques éléments de décors naturels (la fig. 4 propose une représentation de l'île du joueur).

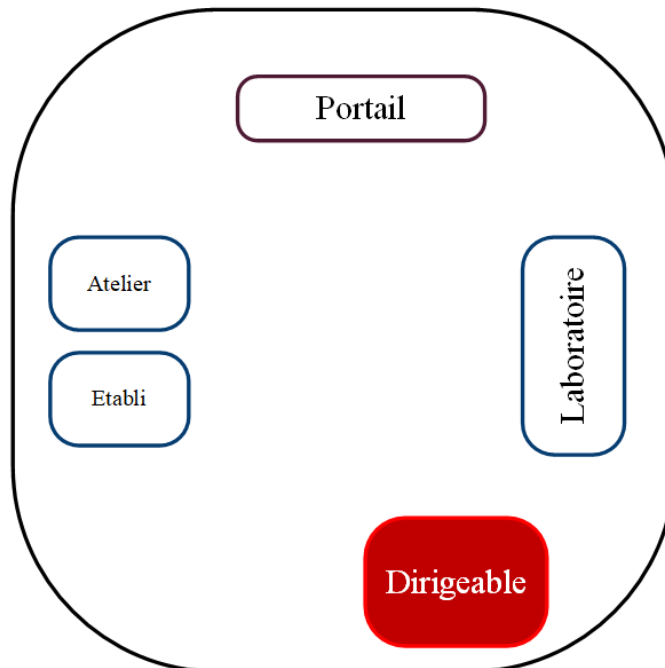


FIGURE 4 – Représentation de l'île du joueur

3.3.2 Description du premier niveau

Le premier niveau sera constitué de trois îles flottantes (la fig. 5 propose une représentation du premier niveau).

Le joueur apparaît sur la petite île au nord, il y aura en son sein un temple en ruine sur lequel il trouvera trois piliers qui serviront à l'invocation du *boss*¹¹ final, une fois qu'il aura vaincu les différents *boss* de zone. Ici, le joueur ne sera confronté qu'à quelques monstres, qui lui serviront à tester le système de combat. Cette petite île est reliée par un pont à l'île centrale.

L'île centrale est séparée en deux parties :

- **UNE ZONE DE RUINES ET DE DÉBRIS** : il n'y aura qu'un point d'intérêt majeur qui est la cachette du *boss*¹¹ de zone.
- **UNE ZONE FORESTIÈRE** : sur laquelle se trouve le monstre permettant d'invoquer le *boss* de zone. Le joueur y trouvera : des camps de gobelins, des points d'eau et un village (principalement des huttes). Les décors seront principalement composés de vastes espaces de verdure avec de très grands arbres, pour donner un aspect assez sombre et dense à la forêt.

Enfin, le joueur accède au sud à un pont fortifié menant à la dernière île, celle-ci est une **ZONE MONTAGNEUSE** dont le principal point d'intérêt est un grand village central, entouré de montagnes, peuplé par un clan goblin guerrier. Au fond du village il y a l'emplacement du *boss*¹¹ de zone, à côté de la montagne. Les décors seront plus secs et arides avec des parties rocheuses.

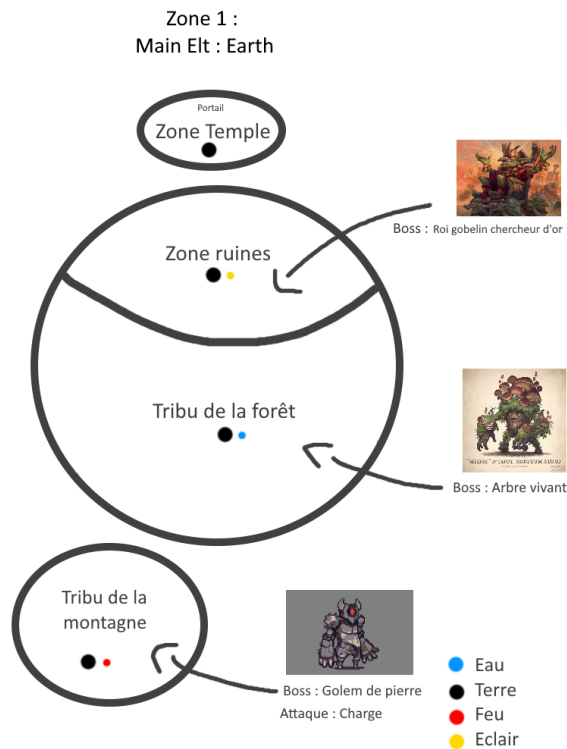


FIGURE 5 – Représentation du premier niveau avec les différents boss de zone

3.3.3 Établissement du *housing*

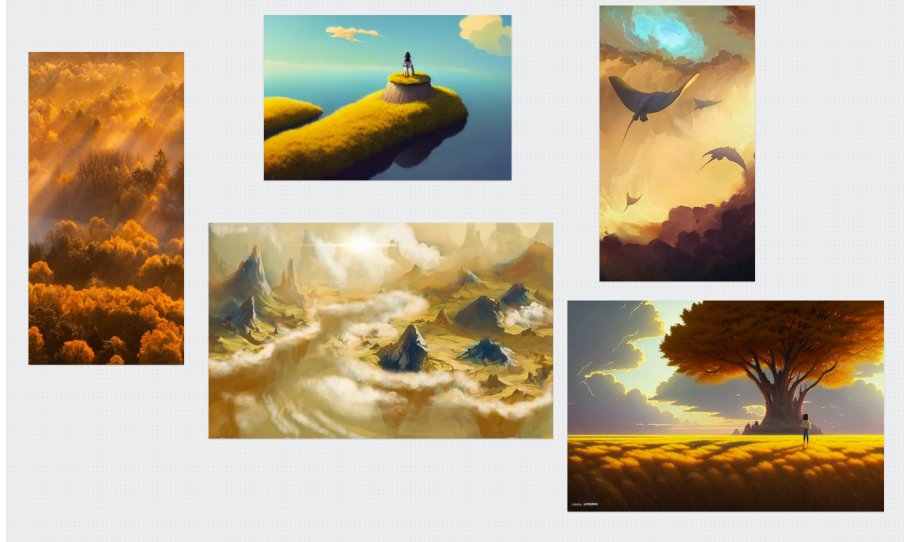


FIGURE 6 – Images de référence pour l'île du joueur [1]

Lorsque le joueur sera sur son île de base, plusieurs actions lui seront accessibles, il pourra notamment réaliser différents types d'*items*¹³ à partir des objets qu'il aura récupéré lors de ses aventures à l'extérieur de sa base. Son but sera de confectionner des éléments qui lui permettront de faciliter son exploration.

Le système de *craft*¹⁰ de Hermita se compose de trois blocs interdépendants associés chacun à un des bâtiments présents sur l'île du joueur (résumé dans la tab. 1).

- **L'ÉTABLI** : servira à transformer les différentes gemmes élémentaires brutes en gemmes polies, à ouvrir les géodes récupérées sur les golems afin d'obtenir une gemme ou une perle élémentaire aléatoire et enfin à transformer les poussières élémentaires en perle et les perles en orbes.
- **L'ATELIER** : permet les *crafts* des potions et des différentes pièces d'équipement. Ces *crafts* utilisent les *loots*⁸ récupérés sur les *mobs*⁷ ainsi que les gemmes et ressources élémentaires forgées dans l'établi.
- **LE LABORATOIRE** : aura pour rôle de permettre au joueur de créer les différents sorts en associant entre-eux des poussières, des perles ou des orbes liées au différents éléments. Un sort est constitué d'un élément principal auquel peuvent s'ajouter jusqu'à deux éléments secondaires. Le niveau du sort est déterminé par les *items*¹³ ayant servi à le créer.

Bâtiment	Produits
Établi	Gemmes polies, perles et orbes
Atelier	Pièces d'équipement et potions
Laboratoire	Les différents sorts

TABLE 1 – Résumé des différents bâtiments

3.4 Programmation

3.4.1 Multijoueur

Afin d'implémenter le multijoueur, même si l'hébergement de notre jeu ne nécessite pas de serveur, on ne peut pas connecter directement des ordinateurs sans qu'il soient en réseau local dans la plupart des cas et nous ne pouvons pas nous permettre de demander aux joueurs d'ouvrir les ports de leur routeur. Nous avons donc besoin d'un relai entre le joueur qui héberge la partie et les autres.

Pour répondre à cette problématique, nous avons décidé de ne pas opter pour Photon mais plutôt d'utiliser Unity Relay qui permet (avec la version gratuite) d'avoir jusqu'à 50 joueurs mis en relation en simultané. Cette limite pourra être repoussée lorsque Epistars™ entamera une levée de fonds. Pour simplifier l'implémentation du multijoueur dans Unity, nous avons opté pour la solution Netcode qui est principalement destinée à faciliter la création de jeux coopératifs à petite échelle. Cette bibliothèque prend en charge des éléments tels que la gestion des scènes, l'animation ou encore la physique.

L'association des outils Netcode et Relay correspond parfaitement à nos besoins puisque le jeu « Boss Room » pris en exemple par Unity est similaire en termes de fonctionnalités réseaux et utilise les mêmes outils.

3.4.2 Implémentation des combats

Le joueur possède différentes statistiques déterminées par son niveau et son équipement :

- **ATTAQUE** : permet de calculer les dégâts que font ses sorts.
- **VITESSE** : détermine la célérité avec laquelle il se déplace.
- **RÉSISTANCE** : réduit les dégâts qu'il subit.

Dans le cas où le joueur est mis KO il faut qu'il puisse réapparaître ce qui implique l'implémentation de la réapparition (*respawn*) du joueur. Pour le fonctionnement des combats, trois systèmes sont actuellement à l'étude (résumés dans la tab. 2).

- le **PREMIER** : consiste à se déplacer avec les touches assignés par le joueur (WASD et touches directionnelles par défaut), sélectionner les sorts (une touche par sort) et utiliser la souris pour viser (le sort est lancé en direction du pointeur de la souris)
- le **DEUXIÈME** : consiste à se déplacer avec les touches assignés par le joueur (WASD et touches directionnelles par défaut), sélectionner le sort avec une touche unique qui change le sort assigné en tournant modulo le nombre de sort et à utiliser la souris pour viser et tirer.
- le **TROISIÈME** : consiste à se déplacer en cliquant avec la souris, sélectionner les sorts (une touche par sort) et à utiliser la souris pour tirer (le sort est lancé en direction du pointeur de la souris).

Déplacement	Lancer les sorts	Visée
Touches assignées	1 touche par sort	Souris
Touches assignées	1 touche pour changer de sort	Souris
Souris	1 touche par sort	Souris

TABLE 2 – Résumé des différents systèmes à l'étude

3.4.3 Intelligence artificielle et gestion des ennemis

L'apparition des ennemis nécessite une part de pseudo-aléatoire pour éviter de rendre l'expérience répétitive pour les joueurs. Il s'agit aussi de les faire apparaître de façon raisonnable et avec des placements logiques.

Pour faciliter le *level design*⁵, nous avons choisi de gérer leur apparition avec des « *spawners* » qui contiendront des listes d'ennemis avec leur probabilité d'apparition et leur niveau de base. Cette solution nous permettra de simplifier la création des niveaux en nous permettant de placer des petits groupes d'ennemis aléatoires dans des zones spécifiques. La plupart des ennemis ne constituent que de maigres obstacles et ne seront pas d'une grande difficulté, nous avons cependant une toute autre vision pour les combats contre les *boss*¹¹. Ces derniers doivent constituer un réel défi pour les joueurs, pour cela nous élaborerons des *boss* aux mécaniques¹² originales dotés de redoutables statistiques afin qu'ils représentent un véritable challenge.

Enfin, nous voulons qu'Hermita soit un jeu vivant. Les ennemis ne doivent pas juste suivre les joueurs, ils doivent aussi avoir des réactions naturelles. Pour leur donner vie, nous passons par des arbres de comportement, un système d'aggro¹⁴ et un algorithme de recherche de chemin tel que A*. Cette recherche de chemin est implémentée par les « Navigation Meshes » sur Unity.

3.4.4 Fonctionnement du *craft* et de l'inventaire

Le joueur choisit en premier lieu sur son île personnelle un maximum de cinq sorts qu'il a confectionnés pour les mettre dans son inventaire de sorts, il ne pourra plus les changer une fois qu'il aura quitté sa base. Les sorts sont un aspect essentiel du jeu, en effet ils sont le seul moyen dont dispose le joueur pour faire des dégâts aux ennemis.

Dans Hermita, (comme mentionné précédemment) il y a quatre éléments, chaque sort et la plupart des ennemis sont liés à un ou plusieurs de ses derniers (même si certains seront neutres). Ainsi, les sorts feront plus ou moins de dégâts en fonction de l'élément de l'ennemi touché. En combat, il y a également les potions qui sont cruciales notamment en ce qui concerne la gestion de la vie et du mana du joueur, elles peuvent être craftées à l'atelier ou bien lootées sur les *mobs*⁷. Les différents *items*¹³ que l'on peut fabriquer existent en plusieurs catégories (la tab. 3 résume leurs différents types et la manière de les produire).

Type	Matières premières
Bâtons	Bras mécaniques, bâtons vivants, gemmes polies
Robes	Peaux animales
Colliers	Colliers basiques, gemmes polies
Potions	Poussières élémentaires

TABLE 3 – Résumé des différents types d'items

Le joueur dispose d'un inventaire « infini » dans le sens où nous n'envisageons pas de lui imposer une limite. De plus, il est possible de regrouper des items de nature identique et les stocker ensemble (ils ne constituent qu'une place d'inventaire).

3.4.5 Système de sauvegarde

En raison de la richesse que nous souhaitons apporter à ce projet, il sera impossible d'y jouer entièrement en une seule fois, de ce fait, il sera nécessaire à un certain stade d'avancement de pouvoir sauvegarder la progression du joueur à savoir : son inventaire, ses statistiques et l'état de sa base afin de pouvoir y revenir facilement. C'est ainsi que se pose la nécessité d'un système de sauvegarde. Deux options furent considérées :

- la classe **PLAYERSPREFS** : elle a pour avantage d'être intégrée à l'écosystème Unity en plus d'être relativement sécurisée mais présente le défaut d'être un peu trop limitée pour l'utilisation que nous voulons en faire car elle ne permet de stocker que trois types de données (entier, chaîne de caractère et flottant).
- les fichiers **JSON** : la sauvegarde sous forme de fichier JSON présente l'avantage d'être d'une grande flexibilité tout en étant relativement simple à appréhender, les fichiers présentent les données sous la forme de couples « key : value ».

C'est la deuxième solution qui sera retenue pour fournir une implémentation de la sauvegarde dans le projet.

3.5 Visuel et animation

3.5.1 Identité du projet

Le projet Hermita fonde sa direction artistique sur la dualité entre la magie considérée comme naturelle dans le monde et l'aspect artificiel des machines *steampunk*¹ amenées par le joueur lorsqu'il vient perturber ce monde a priori bien réglé. Nous voulions proposer une identité assez originale pour notre jeu et le choix du dirigeable pour faire office de logo nous est venu lors d'une réunion dans le but d'être, par son design, un premier pas pour la direction artistique du jeu. Au vu de l'importance que représente cet aspect, le logo Hermita a vécu une importante évolution (représentée fig. 7).

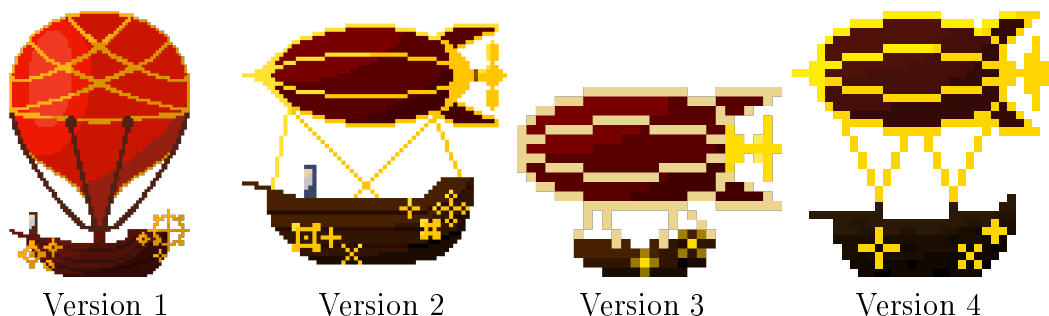


FIGURE 7 – Évolution du logo Hermita

3.5.2 Direction artistique

Le design du jeu sera conçu en pixel art mais sous forme isométrique² (la fig. 8 indique nos références) exception faite de la cinématique de départ qui sera, quant à elle, réalisée dans un style 2D plus « classique ». Le style général marquera l'opposition entre la nature emplies de magie des environnements et les machines (dans un style *steampunk*¹) pour les autres éléments.

Comme le jeu se scinde en deux parties (*housing*³ et aventure), chacune avec leur propre atmosphère, cela se traduira évidemment par des choix de direction artistique bien différents :

- La partie **HOUSING** : devra rappeler, de par son thème et ses couleurs, l'isolement du personnage mais aussi la tranquillité du lieu. Pour ce faire nous utiliserons des couleurs chaudes et nous travaillerons les jeux de lumière, par exemple sur les rayons du soleil, les ombres portées des nuages ou encore en ajoutant des feuilles aux couleurs automnales tombant au sol. L'objectif sera de rappeler au joueur qu'il se situe dans une zone sécurisée, où il n'est pas en proie au danger.
- La partie **AVENTURE** : devra évoquer la richesse des différentes îles qui composent l'univers du jeu. En effet, chaque île est constituée de zones qui possèdent toutes leur propre atmosphère. Ainsi, et à titre d'exemple, lorsqu'une forêt sera représentée, le vert sera la couleur majoritaire et l'accent sera mis sur sa densité pour renforcer l'immersion, tandis que pour un temple, son côté sombre sera prédominant et son aspect intimidant et massif sera mis en exergue par des couleurs plus ternes et brutes, des effets de lumières seront également ajoutés pour faire ressortir certains détails.

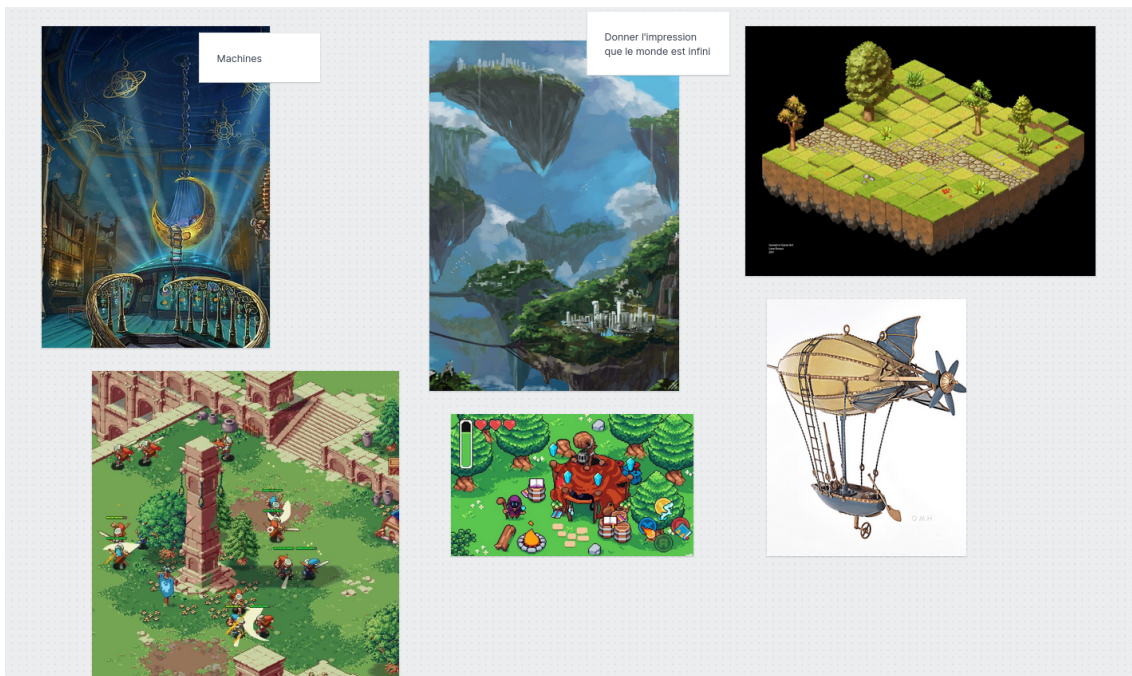


FIGURE 8 – Image de référence pour la direction artistique [2]

4 Conditions de réalisation

4.1 Délai de réalisation

<i>Soutenance</i>	<i>Première</i>	<i>Deuxième</i>	<i>Finale</i>
Programmation			
Multijoueur			
Héberger une partie	**	***	***
Rejoindre une partie	**	***	***
Combat en multijoueur	*	**	***
Implémentation des combats			
Gestion des déplacements	**	***	***
Lancer des sorts	*	**	***
Interface	**	***	***
Intelligence artificielle et gestion des ennemis			
Apparition des ennemis	*	**	***
Mécaniques des boss	*	**	***
Intelligence Artificielle	*	**	***
Fonctionnement du craft et de l'inventaire			
Intégration de l'établi	**	***	***
Création des sorts	*	**	***
Fabrication de l'équipement	*	**	***
Élaborations des potions	*	**	***
Inventaire fonctionnel	*	**	***
Système de sauvegarde			
Stockage de l'inventaire	*	**	***
Enregistrement des stats	*	**	***
Retenir l'état de l'île de départ	*	**	***
Game design			
Agencement du monde	**	***	***
Établissement du housing	**	***	***
Système de combat	**	***	***
Visuels et animations			
Personnage	**	***	***
Ennemis	*	**	***
Environnement	*	**	***
Site web			
Page d'accueil	**	***	***
Mise en page	*	**	***
Système de blog	***	***	***

TABLE 4 – Évolution de l'avancement du projet par soutenance

Tâche commencée : * || Tâche avancée : ** || Tâche terminée : ***

4.2 Contraintes de développement

La société Epistars en tant que prestataire se doit de respecter les règles imposées par le client en plus des règles spécifiques à son activité de développeur. Elle s'engage donc à respecter les prérogatives suivantes :

- L'ENTIÈRETÉ DU CODE SERA RÉDIGÉE EN LANGAGE C# POUR LE MOTEUR UNITY
- LA PLATEFORME CIBLE SERA WINDOWS
- UN MODE MULTIJOUEUR SERA RAPIDEMENT ET FACILEMENT ACCESSIBLE
- DES ÉLÉMENTS D'INTELLIGENCE ARTIFICIELLE SERONT IMPLÉMENTÉS
- UN SITE WEB PRÉSENTANT LE PROJET SERA CRÉÉ
- UNE DOCUMENTATION CLAIRE SERA FOURNIE LORS DE LA LIVRAISON DU PRODUIT
- LE CODE SERA CLAIR, STRUCTURÉ ET COMMENTÉ

4.3 *Business model*

Le *business model* de ce projet (résumé tab. 5) a été établi en tenant compte des restrictions de circonstances, ainsi notre objectif ne sera pas la rentabilité. La raison pour laquelle nous avons pris cette décision s'inscrit dans une démarche à long terme qui vise la conquête d'une part importante du marché vidéoludique. Notre entreprise indépendante se trouve cependant en défaut de renommée et nous sommes conscients que les joueurs ne nous font pas encore confiance. Pour proposer une expérience attrayante, nous avons fait le choix de rendre le jeu entièrement gratuit (*free to play*). De cette manière, nous voulons construire une communauté solide de joueurs pour nous soutenir quand nous ferons des mises à jour ou de prochains opus qui eux, auront vocation à être rentable. Une mission d'optimisation des coûts s'ajoute donc naturellement.

Identification du besoin	Solution	Coûts prévisionnels
Hébergement de partie	Utilisation de Unity Relay	0.00€
Hébergement du site web	Utilisation de Github Pages	0.00€
Création du contenu	Asset gratuit ou produit par l'équipe	0.00€
Matériels nécessaires	Utilisation des équipements de l'entreprise	0.00€
Marketing	Publicité sur les réseaux sociaux	50.00€
Temps et motivation	Café à 0.40€ sur une période de 6 mois	480.00€
Prix de l'électricité	Stagiaire qui pédale	-95.00€
Prix total		435.00€

TABLE 5 – Récapitulatif des coûts

4.4 Suivi des réalisations

	Anouar	Antoine	Baptiste	Dylan
Programmation				
Multijoueur		R	S	
Héberger une partie		X		
Rejoindre une partie		X		
Combat en multijoueur			X	
Implémentation des combats	R		S	
Gestion des déplacements	X			
Lancer des sorts	X			
Interface	X			
Intelligence artificielle et gestion des ennemis		R	S	
Apparition des ennemis		X		
Mécaniques des boss			X	
Intelligence artificielle		X		
Fonctionnement du craft et de l'inventaire			S	R
Intégration de l'établi			X	
Création des sorts				X
Fabrication de l'équipement				X
Élaborations des potions				X
Inventaire fonctionnel		X		
Système de sauvegarde	R		S	
Stockage de l'inventaire	X			
Enregistrement des stats	X			
Retenir l'état de l'île de départ	X			
Game design		S	R	
Agencement du monde			X	
Établissement du housing			X	
Système de combat			X	
Visuels et animations			S	R
Personnage				X
Ennemis				X
Environnement				X
Site web	S	R		
Page d'accueil		X		
Mise en page		X		
Système de blog		X		

TABLE 6 – Répartition des tâches sur le projet

R : Responsable || S : Suppléant || X : Se charge de la tâche

4.5 Communication client



FIGURE 9 – Outils utilisés pour la communication de l'équipe

4.5.1 Site web

Nous utiliserons Svelte comme *framework* puisqu'il fait partie des plus efficaces et performants. Pour le générateur de sites cependant, nous avons fait le choix d'Astro plutôt que Sveltekit (pensé pour Svelte) pour le fait qu'il présente des fonctionnalités comme le « partial hydration » et la présence du Markdown pour une modification de contenu bien plus simple et agréable. Comme nous ne générons que du code statique, nous n'avons pas besoin de SSR (*server side rendering*) ce qui simplifie l'hébergement qui, dans notre cas, se fera grâce à des plateformes gratuites se contentant d'héberger des fichiers comme par exemple les Github Pages.

Les tâches à réaliser pour le site :

- **UN LAYOUT** : définissant la composition de l'ensemble des éléments présents sur toutes les pages (animations, barre de navigation, ...)
- **UNE PAGE D'ACCUEIL** : comprenant des visuels riches et de jolies animations pour donner l'ambiance de l'univers de notre jeu Hermita. C'est ici que se trouvent les CTA (*call to action*) qui vont guider l'utilisateur afin de lui donner une expérience agréable.
- **UN SYSTÈME DE BLOG** : afin de suivre l'activité de nos développeurs. Pour offrir une transparence à notre client, nous rendons public ce suivi et notre équipe a pour mission de tenir une sorte de journal de bord en faisant le point sur chaque tâche dès qu'elle est réalisée. Il y aura une page pour chaque post et une qui les regroupe sous forme de liste.
- **UNE PAGE DE TÉLÉCHARGEMENT** : regroupant les différents liens permettant de télécharger le jeu et les autres documents liés au projet et à sa réalisation.
- **UNE PAGE D'INFORMATIONS** : encyclopédie qui sera une sorte de wiki donnant des informations sur le jeu
- **DES CRÉDITS** : présentant l'équipe et tous les logiciels et ressources utilisés pour nous aider dans la création du jeu.

Framework UI	Générateur de site statique	Édition de contenu
 Svelte	 Astro	 Markdown

FIGURE 10 – Outils utilisés pour réaliser le site

4.6 Aspect technique


Moteur de jeu	IDE	Visuel	Retouche
 Unity	 Visual Studio	 Procreate	 Photoshop

FIGURE 11 – Logiciel techniques utilisés pour réaliser le jeu

4.7 Conclusion

Le jeu Hermita est un projet de développement de jeu vidéo ambitieux qui vise à créer un monde unique pour les joueurs. Notre objectif est de créer un jeu qui soit accessible à tous et qui proposera une expérience captivante. Nous avons travaillé dur pour créer un univers unique et cohérent pour Hermita, axé sur la dualité entre la magie naturelle de ce monde et l'aspect artificiel des machines amenées par le joueur.

De plus, la densité en contenu et les nombreuses possibilités offertes par notre jeu permettront à l'ensemble de notre équipe d'enrichir ses compétences en C# et plus globalement en termes de création de jeu vidéo. Enfin, cette expérience nous permettra de développer notre capacité à gérer des *deadlines* et à travailler en équipe notamment en apprenant à communiquer et à être ouvert aux idées des autres.

Le jeu Hermita sera disponible sur notre site web et sur GitHub à partir du 6 juin 2023. Nous espérons que vous serez aussi excités que nous à l'idée de découvrir ce que ce jeu pourra vous offrir !



Notre mage vous attends...

5 ANNEXE

5.1 Schémas UML

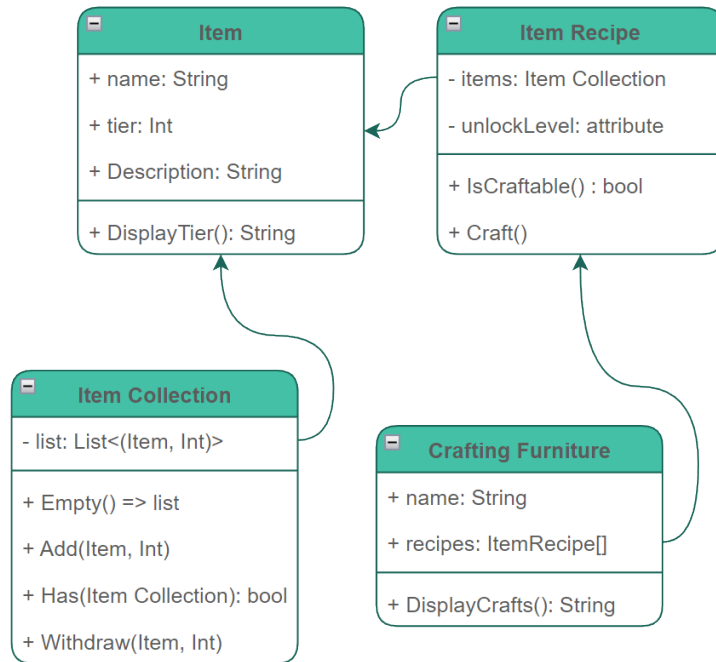


FIGURE 12 – Idée d'implémentation pour le housing

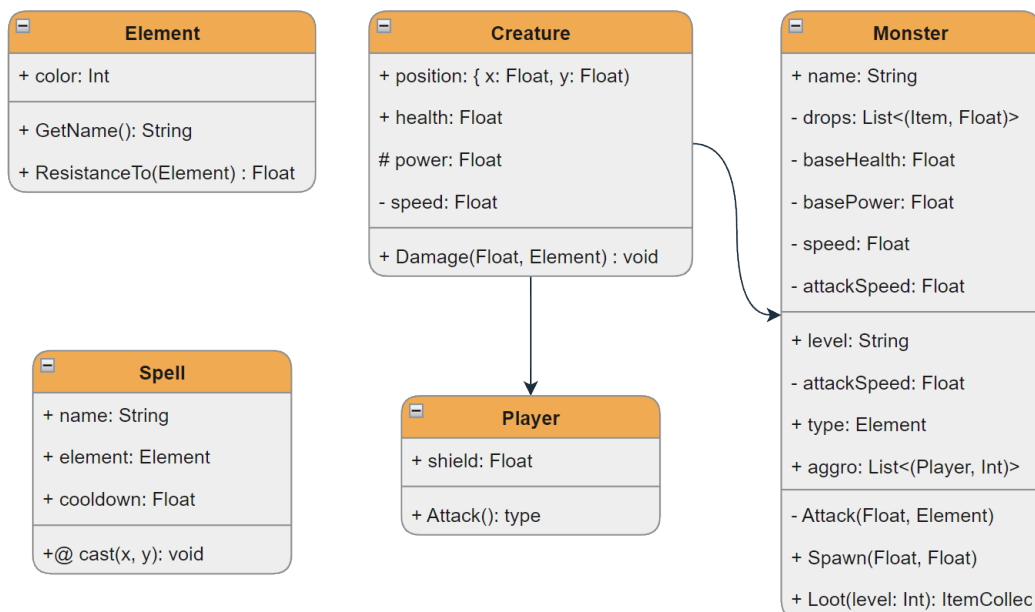


FIGURE 13 – Idée d'implémentation pour le système de combat

5.2 Glossaire

- [1] **Steampunk** : Sous-genre de la science-fiction se déroulant généralement au XIXe ou au début du XXe siècle, et devant son nom à l'utilisation massive de technologies en avance sur leur temps, principalement à base de machines à vapeur.
- [2] **Vue isométrique** : Représentation en perspective d'un objet, où les trois arêtes principales (qui correspondent aux trois dimensions de l'objet) forment des angles égaux de 120 degrés du point de vue de la caméra.
- [3] **Housing** : Élément de *gameplay* consistant à gérer et aménager une parcelle privée appartenant au joueur.
- [4] **Game design** : Consiste à la fois à créer les règles du jeu et faire en sorte qu'il soit fun. Cela passe par définir tous ses aspects, tels que les défis, le scénario, le rythme du jeu, les méthodes pour gagner, les niveaux de difficultés et l'ensemble des personnages (jouables par l'utilisateur et non joueurs).
- [5] **Level design** : Conception des niveaux ou cartes d'un jeu vidéo en fonction des éléments graphiques et des éléments de *game design* à intégrer dans le niveau.
- [6] **Triple A** : Décrit un jeu qui atteint le top 10 des ventes sur le marché, la performance commerciale l'emporte sur les questions de budget.
- [7] **Mobs** : Traduction anglaise de créature ou monstre, désigne plus globalement tous les personnages non joueurs de notre jeu.
- [8] **Loot** : Butin récupéré sur les *mobs*.
- [9] **Dash** : Déplacement très rapide sur un court instant.
- [10] **Craft** : Action de fabriquer. Peut se traduire par fabrication ou artisanat.
- [11] **Boss** : Grande créature se trouvant en fin de zone ou de niveau et ayant des mécaniques particulières le rendant plus difficiles à tuer que les *mobs* classiques (donne souvent une récompense spéciale).
- [12] **Mécaniques des boss** : Ensembles des sorts, déplacements et techniques utilisés par les *boss* au cours de leurs combats.
- [13] **Item** : Objet qui peut être collecté dans un jeu. Peut être relativement permanent (par exemple les armes et les armures) comme consommable, avec des effets positifs ou négatif.
- [14] **Système d'aggro** : Système permettant au monstre de définir quel joueur il va attaquer.

5.3 Sources

- [1] **Images de référence pour l'île du joueur** :
 - <https://playgroundai.com/>
- [2] **Images de référence pour la direction artistique** :
 - <https://twitter.com/daoklyn/status/1250219511514632193>
 - <https://www.goodreads.com/review/show/3903241322>
 - <https://www.pinterest.es/pin/328481366551231330/>
 - <https://www.pinterest.es/pin/574420127479091155/>