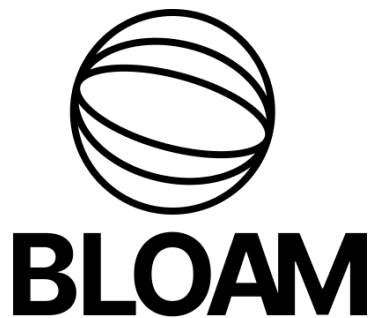


Rapport de soutenance

Benjamin Poussart, Mattéo Galus, Alexandre Abbruzzese
Oren Guetta, Léon Moneger

15 janvier 2025



Sommaire

Résumé	3
Introduction	4
Etat de l'Art	6
1 Conception et Développement	8
1.1 Scénario (Alexandre et Oren)	8
1.1.1 L'inspiration et la conception	8
1.1.2 Processus de Rédaction	8
1.1.3 Défis Rencontrés	9
1.1.4 Conclusion	9
1.2 Site web (Alexandre)	10
1.2.1 Préparation et choix des outils	10
1.2.2 Inspirations pour le design	10
1.2.3 Problèmes rencontrés et solutions apportées	10
1.2.4 Conclusion	11
1.3 Design de niveaux (Oren et Léon)	11
1.3.1 Processus de Rédaction	11
1.3.2 Pourquoi cet univers et cette mécanique de jeu ?	12
1.3.3 Les défis rencontrés	14
1.3.4 Conclusion	15
2 Implémentation	17
2.1 Réseau (Benjamin et Léon)	17
2.2 Programmation (Mattéo et Benjamin)	18
2.2.1 Défis rencontrés et solutions	18
Conclusion	21
Sources	23
2.3 Scénario :	23
2.4 Programmation réseau :	23
2.5 Programmation solo :	23

Résumé

Notre projet se nomme Final Dream. Il s'agit d'un jeu vidéo de plateformes en 2 dimensions. L'objectif est de créer un jeu original pouvant être joué à 1 ou 2 joueurs. Nous voulons implémenter un réseau de type LAN (local area network) pour la partie multijoueur. Pour l'IA, nous voulons ajouter des ennemis qui peuvent suivre le ou les joueurs pour attaquer. Nous avons décidé d'utiliser Unity pour développer notre jeu, car il nous a semblé que ce logiciel offrait une plus grande facilité d'utilisation que Godot. Pour le "version control", il existe une fonctionnalité intégrée à Unity pour remplir ce rôle. Néanmoins, nous avons privilégié GitHub pour ceci car la documentation est plus complète et de l'aide sera plus facilement obtainable. Notre jeu est inspiré d'autres jeux en 2 dimensions de plateformes, par exemple pour toutes les mécaniques propres aux jeux de plateformes, comme les déplacements basiques, ou les combats contre des ennemis. Les tâches sur lesquelles nous devons travailler pour cette soutenance sont : les spécificités techniques traitées par Benjamin Poussart et Mattéo Galus, le design de niveaux par Oren Guetta et Léon Monéger, le concept et la vision par Alexandre Abbruzzese et Oren Guetta, les graphismes par Léon Monéger et Alexandre Abbruzzese, la programmation du gameplay multijoueur par Benjamin Poussart et Alexandre Abbruzzese et enfin le réseau par Benjamin Poussart et Léon Monéger. Nous avons pu terminer le concept et la vision qui désigne entre autres le scénario du jeu. Nous avons aussi pu finir les spécificités techniques qui listent les différentes mécaniques de jeu que nous voulons implémenter ainsi que les outils que nous utilisons pour faire le jeu. Néanmoins, cette liste changera probablement avec l'avancement du projet. Cependant, nous sommes en retard sur le design de niveaux, dont la réalisation n'est pas encore terminée, ainsi que sur le réseau, la programmation multijoueur, et les graphismes. Effectivement, nous n'avons pas encore trouvé comment mettre en place un réseau LAN sur unity, ou comment implémenter un mode de jeu à 2 joueurs. De plus, la conception des décors et des sprites pour les personnages n'a pas encore commencé, contrairement à ce qu'indique notre diagramme de Gantt. Nous avons toutefois commencé le développement du jeu, avec quelques programmes pour faire avancer, reculer, ou sauter le personnage.

Introduction

Dans le cadre de la création d'un jeu vidéo, notre entreprise, la BLOAM Corp. s'est tournée vers un jeu vidéo de plateforme en 2 dimensions. De nombreuses inspirations nous ont amenées à faire ce choix, toutefois, nous cherchons tout de même à réaliser un jeu original. Un des intérêts de ce projet est de se mettre au défi, afin de perfectionner nos compétences en informatique et aussi en gestion de projet. Ce rapport de soutenance vise à présenter ce que nous avons réalisé depuis la soutenance fonctionnelle, aussi bien dans la programmation que dans la conception et le développement, c'est-à-dire ce qui est encore au stade de concept et non de réalisation concrète. La réalisation d'un projet ne se passe rarement comme prévu. Nous risquons effectivement de subir quelques décalages au niveau de la planification des tâches. Néanmoins, nous prévoyons toujours de pourvoir à la fin un jeu vidéo complet, qui répond aux attentes du client. Les buts et intérêts de ce projet sont multiples. Tout d'abord, la réalisation d'un jeu vidéo est un projet complexe pour les néophytes en matière de programmation, mais aussi en design, réseau, et autres aspects. Le processus de création de jeu vidéo implique une grande organisation et passe par plusieurs étapes : la conceptualisation du jeu, la définition des mécaniques, la création des éléments graphiques, et l'implémentation. L'ajout d'une intelligence artificielle ainsi que d'un mode multijoueur rajoute de plus un défi à surmonter en termes de programmation. L'outil principal que nous utilisons pour réaliser ce projet est Unity. Faire un jeu en utilisant ce logiciel implique de se familiariser avec lui. Pouvoir utiliser de tels logiciels est un atout non négligeable dans le domaine de la programmation. Plus généralement, ce projet nous permettra de développer nos compétences en informatique mais aussi en gestion de groupe. Effectivement, l'organisation qu'implique ce type de projet ne peut qu'améliorer notre capacité à travailler en groupe, en plus de parfaire nos compétences techniques, comme dit précédemment. Nous devons planifier et organiser les différentes tâches que nous nous sommes données, mais aussi suivre de près l'avancement de ces tâches et communiquer sans arrêt pour pouvoir être capable de gérer tout type d'imprévus, que ce soit au sein du groupe que dans le travail à fournir. Ce projet nous apprend à gérer notre temps, et aussi dans la répartition des responsabilités. Nous apprendrons également à gérer des problèmes

techniques, tels que des bugs ou d'autres défis, en comprenant quel est le problème et en cherchant en groupe des solutions. Dans ce rapport, nous allons présenter les avancées techniques et conceptuelles sur notre projet. Nous allons d'abord faire un point sur l'état de l'art des jeux vidéo de plateformes, car il est intéressant de voir quelle "formule" est utilisée pour chaque jeu, et cela peut nous permettre de tirer certaines inspirations, notamment aux niveau des mécaniques de jeu. Ensuite, nous allons présenter la vision que l'on a de notre projet pour le moment, avec des éléments déjà prévus et visualiser comme le scénario du jeu, ou encore la conception des différents niveaux de notre jeu. Dans cette partie, nous allons également parler des éléments que nous prévoyons probablement d'implémenter dans le futur. Néanmoins, ces éléments sont susceptibles de changer, voire même de disparaître complètement, parce que notre visualisation n'est pas complète, et il nous faudra procéder au fur-et-à-mesure de l'avancement du projet. Dans la partie suivante, nous présenterons plus précisément les différents éléments de notre jeu que nous avons d'ores et déjà implémentés. Il s'agit de programmation en C# sur unity, et consiste en la mise en place d'un joueur et des déplacements basiques de celui-ci. Enfin, nous finirons par conclure, et nous montrerons les sources qui nous ont aidés à avancer, ainsi qu'une annexe contenant du code que nous avons réalisé.

Etat de l'Art

De nombreux jeux de plateforme ont laissé une empreinte indélébile dans l'histoire du jeu vidéo, tels que *Super Mario Bros.*, largement acclamé par la critique et les sites spécialisés comme étant l'un des meilleurs jeux de tous les temps. Aujourd'hui encore, il demeure l'un des jeux vidéo les plus connus au monde. D'autres titres de la franchise, comme *Super Mario 64* et *Super Mario Galaxy*, ont également marqué l'évolution du genre. La série *Rayman* a aussi eu un impact significatif avec des jeux comme *Rayman Origins* et *Rayman Legends*. Parmi les autres grands noms du genre, on peut citer *Sonic*, *Donkey Kong*, *Crash Bandicoot*, *Mega Man*, *Ratchet & Clank*, ainsi que des jeux moins conventionnels tels que *Chrome Dino* (le jeu du dinosaure de Google). Tous ces jeux, largement appréciés par les amateurs de jeux vidéo, ont contribué à définir les caractéristiques clés du genre, à savoir :

- **Gameplay simple et intuitif** : La simplicité du concept – sauter et se déplacer entre des plateformes – permet une prise en main rapide et accessible à tous les joueurs, quel que soit leur niveau de compétence.
- **Contrôles précis** : Les meilleurs platformers se caractérisent par des commandes précises et réactives, donnant au joueur un contrôle immédiat et fluide sur les actions de son personnage. *Super Mario Bros*, *Céleste* et *Hollow Knight* en sont d'excellents exemples.
- **Conception créative des niveaux** : Les niveaux sont souvent conçus de manière imaginative, avec des environnements variés, des obstacles originaux et des énigmes intégrées au déplacement. *Super Mario Galaxy* a, par exemple, révolutionné ce concept avec ses niveaux basés sur la gravité.
- **Esthétique et créativité visuelle** : Les platformers offrent une grande liberté artistique. Des jeux comme *Rayman Legends*, avec ses graphismes colorés et artistiques, ou *Ori and the Blind Forest*, avec son univers visuel onirique, démontrent à quel point ce genre peut être visuellement captivant.
- **Mécaniques de saut et de déplacements uniques** : Le saut étant l'élément central du gameplay, les joueurs doivent souvent maîtriser différents types de sauts (doubles, muraux, etc.). Des mouvements acrobatiques, tels que la course,

la glissade ou l'escalade, viennent parfois enrichir le gameplay. Par exemple, le double saut dans *Rayman* ou le saut mural dans *Super Mario*.

- **Divers environnements interactifs** : Les plateformes elles-mêmes peuvent constituer des obstacles en se déplaçant, disparaître ou être bloquées. Les jeux *Celeste* et *Donkey Kong Country* mettent l'accent sur la variété des obstacles à surmonter.
- **Les objets collectables et les niveaux d'exploration** : des objets comme les étoiles dans *Super Mario 64* ou les bananes dorées dans *Donkey Kong 64*, apportent une dimension d'exploration et incitent les joueurs à découvrir tous les secrets d'un niveau.
- **Éliminer les ennemis par des actions simples** : Souvent, les ennemis sont éliminés en sautant dessus ou en les frappant. D'autres jeux de plateforme, tels que *Hollow Knight*, intègrent des mécanismes de combat plus complexes.
- **Multijoueur local ou en ligne** : Certains jeux de plateforme proposent des modes multijoueurs, permettant à plusieurs joueurs de coopérer ou de s'affronter dans les mêmes niveaux. *New Super Mario Bros. U* et *Rayman Legends* en sont de bons exemples.

L'équipe, ayant déjà eu l'occasion de jouer à certains jeux de plateforme qui les ont marqués, a décidé de s'inspirer de certains d'entre eux pour ce projet, parmi lesquels :

- *Hollow Knight* (considéré comme l'un des meilleurs jeux de tous les temps)
- *Super Mario Galaxy*
- *Inside* (considéré comme l'un des meilleurs jeux de tous les temps)
- *Dead Cells* (jeu français ayant reçu d'excellentes critiques)
- *Donkey Kong Country Returns 3D*

1 Conception et Développement

1.1 Scénario (Alexandre et Oren)

Pour le projet de première année, nous avons créé un scénario pour un jeu vidéo intitulé Final Dream, développé comme une intrigue onirique et introspective sur les thèmes de la rédemption, du pardon et de l'acceptation de soi. Cela nous a aidé à appliquer les techniques narratives du jeu et en même temps à nous plonger dans la construction d'un univers attrayant pour les joueurs.

1.1.1 L'inspiration et la conception

Ce scénario s'est inspiré de certains jeux en termes de richesse narrative ou de conception qui nous ont marqués. Il s'agit notamment de :

- *Hollow Knight* : Le jeu nous a inspiré par son approche de la narration minimaliste, la façon dont les détails sur le monde et l'histoire font progressivement surface au joueur. Cela nous a fait développer ce scénario avec des zones grises, provoquant ainsi l'imagination des joueurs.
- *Inside* : J'ai vraiment apprécié l'ambiance sombre et mystérieuse que ce jeu offre. Cela nous a donné l'idée d'utiliser des environnements symboliques comme une forme par laquelle les émotions et les conflits internes des personnages peuvent être exposés.
- *Unravel* : Utiliser des objets du quotidien pour évoquer des souvenirs et des émotions nous a inspiré à intégrer des éléments symboliques dans le gameplay.
- *Dead Cells* et *Super Mario Galaxy* : plus du côté visuel et du design, mais en substance, ces jeux nous ont inspirés à essayer de rendre le monde de Final Dream aussi immersif que possible.

1.1.2 Processus de Rédaction

Le scénario a été initié en réfléchissant aux thèmes sur lesquels il est censé s'appuyer. L'intention principale était de mettre en lumière le regret et la rédemption humains dans le cadre des rêves. Ces idées fondamentales étant posées, nous avons procédé

au compartimentage du scénario en quatre actes, chacun portant son propre objectif narratif clairement défini. Nous avons consulté des ressources en ligne qui pourraient nous aider à écrire des scénarios de jeux vidéo. Par exemple, game developer propose des articles sur la structure narrative dans les jeux qui se sont révélés très utiles. Nous avons structuré notre récit en s'inspirant d'un modèle narratif universel qui décrit le parcours initiatique d'un individu. Ce schéma, présent dans de nombreuses cultures et récits, offre une base solide pour élaborer des histoires captivantes et profondes. Nous y avons fait référence, mais nous ne l'avons pas suivie à la lettre car elle nous a fourni des idées précieuses pour structurer les transformations des personnages principaux Elena et Milo. Trouver l'équilibre entre récit et gameplay était essentiel, mais le processus se mesurait en actes. Un scénario de jeu doit développer une phase de gameplay équivalente qui fait ressortir les défis émotionnels d'Elena. Par exemple, l'acte 1 se déroule dans un lieu de désordre qui reflète son esprit fragmenté. Cela influence directement le type d'énigmes et d'interactions que nous proposons au joueur.

1.1.3 Défis Rencontrés

Un des défis majeurs a été de rendre l'histoire engageante tout en respectant les contraintes de notre projet scolaire. Le temps de développement limité nous a poussé à simplifier certains aspects du scénario sans sacrifier la profondeur émotionnelle. Par exemple, certaines scènes initialement prévues ont été fusionnées pour éviter une surcharge de contenu. Un autre défi a été de trouver une voix unique pour chaque personnage, surtout Milo, qui évolue tout au long de l'histoire. Pour cela, nous avons imaginé plusieurs brouillons de dialogues et testé différentes approches avant de trouver un ton qui fonctionne.

1.1.4 Conclusion

L'écriture de ce scénario a été une expérience enrichissante, nous permettant d'explorer la narration interactive tout en consolidant nos compétences en storytelling. Nous sommes fiers du résultat obtenu et convaincus que l'histoire de Final Dream pourra captiver les joueurs en leur offrant une expérience émotionnelle et immersive.

1.2 Site web (Alexandre)

Dans le cadre du projet de jeu vidéo, Alexandre a été chargé de développer le site web. Voici une description de la démarche, des inspirations qui ont guidé ce travail, des problèmes rencontrés et des solutions mises en place.

1.2.1 Préparation et choix des outils

Avec une expérience préalable en développement web, l'équipe disposait de bases utiles pour débiter. Pour ce projet, Vue.js a été choisi comme framework JavaScript principal. Ce choix s'explique par une familiarité précédente avec cet outil et sa grande praticité. Toutefois, les dernières utilisations de ce Framework remontant à longtemps ont poussé Alexandre à relire ces anciens projets pour comprendre les bases et les fonctionnalités essentielles du framework. En complément, une stack classique de technologies web a été utilisée : HTML, CSS et JavaScript. Pour la mise en forme, Tailwind CSS a été intégré, offrant une approche utilitaire pratique pour styliser rapidement les éléments d'interface. La documentation officielle de Tailwind a constitué une référence majeure.

1.2.2 Inspirations pour le design

Pour la direction artistique du site web, le site de l'entreprise Anthropic a servi de modèle. Ce site, avec un design épuré et fonctionnel, correspondait à l'image recherchée pour le projet. Certains éléments visuels et structurels ont été adaptés pour présenter le travail de manière claire. En parallèle, le site W3Schools a été consulté pour des informations précises sur les commandes HTML, CSS et JavaScript. YouTube a également fourni des tutoriels utiles pour comprendre des méthodes de développement web sur vue.js.

1.2.3 Problèmes rencontrés et solutions apportées

1. Rafraîchissement des connaissances sur Vue.js

- Problème : un manque de maîtrise initiale du framework Vue.js
- Solution : Analyse des anciens projets et suivi de tutoriels en ligne pour actualiser les compétences.

2. Intégration de Tailwind CSS

- Problème : Difficultés initiales avec l’approche utilitaire de Tailwind CSS.
- Solution : Utilisation de la documentation officielle et d’exemples pour comprendre comment structurer le code.

3. Gestion du temps

- Problème : Difficultés pour gérer les échéances.
- Solution : Mise en place d’un planning clair avec des tâches décomposées et des échéances réalistes.

1.2.4 Conclusion

Le développement de ce site web a permis de combiner des compétences existantes avec l’apprentissage de nouvelles techniques. En utilisant Vue.js, HTML, CSS, JavaScript et Tailwind CSS, un site efficace a été créé pour présenter le projet. Les ressources consultées ont joué un rôle important, et les obstacles rencontrés ont été surmontés grâce à des solutions adaptées.

1.3 Design de niveaux (Oren et Léon)

1.3.1 Processus de Rédaction

Dans le cadre de la création du design de niveaux de notre jeu vidéo, la rédaction a été une étape cruciale pour établir la base narrative et ludique du projet. L’objectif principal était de concevoir des environnements qui non seulement servent l’histoire du jeu, mais qui soient également cohérents avec les mécaniques de gameplay. Le travail s’est divisé en plusieurs phases :

- **Phase de brainstorming** : Nous avons d’abord exploré les grands thèmes du jeu, qui se concentrent sur la psychologie de l’héroïne, Elena, et ses luttes internes. En collaboration avec mon camarade Léon, nous avons déterminé que chaque acte du jeu correspondrait à une étape spécifique de la vie d’Elena : l’enfance, l’adolescence, l’âge adulte et enfin la rédemption et l’acceptation. Ces transitions ont servi de base pour structurer les environnements et les mécaniques de jeu associées.

- **Création de l'univers narratif** : Pour chaque acte, nous avons imaginé des lieux symboliques. Le premier acte, "Les Souvenirs d'Enfance", prend place dans un parc, un espace symbolique de l'innocence, mais aussi de la nostalgie. Ce choix de décors influençait directement les interactions entre les personnages et les objets, comme par exemple le miroir brisé représentant la fracture entre innocence et réalité. L'évolution du décor au fil des actes a été pensée pour renforcer l'aspect psychologique du jeu.
- **Élaboration des mécaniques de gameplay** : La coopération entre Milo et Elena a orienté la création des mécaniques de gameplay. L'idée de proposer un gameplay coopératif, où chaque joueur a des tâches spécifiques et complémentaires, s'est imposée comme un choix fondamental. Par exemple, dans l'Acte 1, Milo doit résoudre des énigmes liées aux objets du passé d'Elena, tandis qu'Elena, en tant que personnage inconsciente, réagit aux éléments et événements du monde autour d'elle. Cette interaction entre les deux joueurs crée une dynamique où la coopération devient essentielle pour progresser, chaque joueur ayant un rôle unique à jouer pour permettre à l'autre d'avancer dans son exploration.

1.3.2 Pourquoi cet univers et cette mécanique de jeu ?

Dès les premières discussions autour du projet, l'idée de plonger dans les luttes internes d'un personnage est apparue comme une direction naturelle. L'aspect psychologique, en particulier l'idée de se confronter aux différentes étapes de la vie d'un individu, a offert une base solide et pleine de potentiel pour l'élaboration du gameplay et de la narration. En explorant la vie d'Elena à travers ses souvenirs, ses peurs et ses aspirations, il est rapidement devenu évident que l'univers du jeu devait être onirique, où chaque environnement représente une phase importante du parcours émotionnel et psychologique de l'héroïne. L'univers a ainsi été conçu comme un miroir de ses luttes internes, avec des transitions de lieux qui reflètent les étapes majeures de la vie d'Elena : un parc calme et nostalgique pour l'enfance, un environnement scolaire étouffant pour l'adolescence, une ville en ruine pour l'âge adulte, et enfin un lieu de lumière et de réconciliation pour la rédemption. Cette progression visuelle et symbolique de l'environnement a non seulement renforcé l'impact émotionnel du jeu, mais aussi

ancré chaque acte dans un contexte à la fois narratif et ludique. Les transitions entre ces mondes ont été pensées pour illustrer la tension qui existe dans l'esprit d'Elena, une quête pour se retrouver et surmonter les peurs qui l'empêchent d'aller de l'avant. En ce qui concerne la mécanique de jeu, l'idée de proposer une expérience coopérative, mais aussi adaptable pour une expérience solo, s'est imposée. Le joueur peut incarner Elena seul ou être accompagné de Milo, le personnage "guide". Ce dualisme dans l'expérience de jeu permet de diversifier l'approche tout en maintenant une unité émotionnelle forte. Lorsque le jeu est joué en solo, le joueur contrôle Elena et doit gérer ses interactions avec l'environnement et les objets pour avancer. En coopération, Milo vient soutenir Elena, l'aidant à surmonter certains obstacles ou à résoudre des énigmes spécifiques. Ce système permet de conserver une immersion solitaire tout en offrant une dynamique sociale lorsque le jeu est joué à deux. Ce choix de gameplay a présenté des défis importants, notamment celui de maintenir une expérience fluide et gratifiante dans les deux configurations. Trouver un équilibre entre une mécanique qui fonctionne bien en solo et en coopération sans simplification excessive a demandé des ajustements constants. Nous avons veillé à ce que les énigmes et les interactions ne dépendent pas uniquement des actions de Milo et Elena, mais soient plutôt organisées autour d'actions complémentaires qui varient selon le mode de jeu. Cela a permis de garantir que l'expérience reste enrichissante, quelle que soit la configuration choisie. Les inspirations pour cet univers et cette mécanique ont été nombreuses. Des jeux comme *It takes two* et *A Way Out* ont montré que des expériences narratives peuvent être conçues autour de personnages complémentaires, tout en permettant une interaction à la fois indépendante et interconnectée, ce qui a influencé la structure du jeu. *It takes two* a particulièrement inspiré la création d'une connexion émotionnelle forte entre les personnages, tout en offrant la possibilité de vivre cette dynamique de manière flexible, en solo ou en coop. Par ailleurs, des jeux tels que *Limbo* ont prouvé qu'il est possible de transmettre des thèmes profonds à travers des environnements minimalistes et une mécanique de gameplay simple mais percutante. Cela a inspiré la décision de concevoir un univers où chaque décor devient un personnage à part entière, illustrant l'évolution de l'état d'esprit d'Elena. Le choix de l'univers onirique a également été nourri par des réflexions autour de l'importance des objets symboliques dans l'histoire, à la manière de *The Legend of Zelda*, où les objets

et artefacts sont utilisés non seulement pour avancer dans le gameplay, mais aussi pour déclencher des réflexions émotionnelles ou narratives. Un exemple marquant est celui du miroir brisé dans l'Acte 1, symbolisant la fracture entre innocence et réalité d'une manière visuellement forte mais aussi fonctionnelle. Cet objet n'est pas simplement un élément de gameplay, mais un symbole de la perte et de la reconstruction, un thème qui se développe tout au long du jeu.

1.3.3 Les défis rencontrés

Lors de la rédaction du design des niveaux, plusieurs défis ont émergé, principalement liés à la cohérence entre la narration et le gameplay, ainsi qu'à la création d'interactions fluides entre Milo et Elena.

Maintenir la cohérence narrative : L'un des défis majeurs a été de s'assurer que l'évolution des décors et des mécaniques de jeu restait en parfaite harmonie avec le parcours émotionnel d'Elena. Par exemple, dans l'Acte 2, où l'adolescence est représentée par une atmosphère sombre et oppressante, il était essentiel que les environnements ne reflètent pas uniquement les luttes internes d'Elena, mais aussi les enjeux sociaux et scolaires de cette période. La recherche d'un équilibre entre un gameplay engageant et une histoire cohérente a été complexe. Il a fallu veiller à ce que les mécaniques de jeu renforcent l'état d'esprit d'Elena, tout en restant suffisamment immersives pour que le joueur se sente immergé dans l'univers sans que l'histoire perde sa crédibilité.

La mécanique de coopération : Concevoir des énigmes où Milo et Elena doivent interagir de manière complémentaire a représenté un autre défi important. Les tâches de chaque joueur devaient être ajustées à plusieurs reprises pour garantir que la progression soit fluide et logique, quelle que soit la configuration choisie. Par exemple, dans l'Acte 3, où les joueurs doivent agir simultanément pour résoudre des dilemmes moraux, il a été délicat de définir les moments où leurs actions devaient s'entrelacer et ceux où elles devaient être indépendantes. L'objectif était d'éviter une simplification trop évidente, tout en permettant au joueur seul de pouvoir avancer dans des situations où la coopération semble essentielle.

Création d'une expérience émotionnelle : Un autre défi majeur était de faire en sorte que le joueur ressente les émotions d'Elena, qu'il s'agisse de la tension, de la peur

ou de la réconciliation, à travers ses interactions avec Milo. Les énigmes ne devaient pas être de simples obstacles de gameplay, mais aussi des moyens d’approfondir la compréhension émotionnelle du joueur sur les luttes internes du personnage. Nous avons dû nous assurer que chaque action et interaction ait un poids émotionnel, même lorsque les personnages ne communiquent pas directement entre eux.

1.3.4 Conclusion

Cette partie design de niveaux a été une exploration passionnante de la manière dont un environnement ludique peut non seulement soutenir un gameplay, mais aussi refléter l’évolution émotionnelle d’un personnage. À travers la conception des niveaux, l’objectif principal était de créer des espaces immersifs qui servent à la fois de cadre et de miroir des luttes internes d’Elena. Chaque environnement a été pensé pour symboliser une étape de son parcours personnel, tout en soutenant les mécaniques de jeu et en contribuant à l’expérience émotionnelle du joueur. Cette approche nous a permis de jouer avec la notion de dualité, notamment à travers le système coopératif, où la complémentarité entre les personnages renforce la dynamique de l’histoire. Les choix artistiques et mécaniques ont été largement influencés par des jeux comme *It takes two*, *Limbo*, qui ont démontré qu’il est possible de créer une expérience profonde à travers des mécaniques simples mais percutantes. De même, l’influence de *The Legend of Zelda* s’est matérialisée dans l’utilisation d’objets symboliques, où chaque artefact, tout en étant essentiel au gameplay, renforce également la narration. L’un des principaux défis a été de maintenir une cohérence parfaite entre l’évolution narrative et les mécaniques de gameplay. Par exemple, l’Acte 2, centré sur l’adolescence d’Elena, a nécessité une attention particulière pour équilibrer l’ambiance visuelle et émotionnelle avec les enjeux du gameplay. Concevoir des énigmes cohérentes pour chaque mode de jeu (solo ou coopération) a également demandé des ajustements constants, afin que chaque interaction, qu’elle soit indépendante ou complémentaire, soit significative tant du point de vue narratif que ludique. Ce projet a mis en lumière l’importance d’une conception où l’environnement, les personnages et le gameplay s’entrelacent pour créer une expérience émotionnelle riche. Chaque détail a été pensé pour non seulement faire avancer le joueur dans le jeu, mais aussi pour l’impliquer dans le cheminement intérieur

du personnage. Tout cela a abouti à une expérience de jeu qui cherche à capturer les luttes, les aspirations et les réconciliations d'Elena de manière fluide et immersive. Bien que le résultat soit satisfaisant, plusieurs pistes restent ouvertes pour enrichir davantage l'expérience. Une exploration plus poussée des choix moraux, ou l'intégration de mécaniques encore plus subtiles, pourrait approfondir l'interconnexion émotionnelle du joueur avec les personnages. Ce projet a, en tout cas, permis de mieux comprendre la puissance de l'intégration entre gameplay et narration dans la construction d'un univers immersif.

2 Implémentation

En plus de la conception, ainsi que du développement, nous avons également implémenté plusieurs éléments dans notre projet. Tout d’abord, nous avons mis en place un réseau très basique sur unity pour connecter plusieurs joueurs entre eux. Ensuite, nous avons mis en place les mouvements basiques des personnages, c’est-à-dire se déplacer et sauter.

2.1 Réseau (Benjamin et Léon)

Pour le réseau, nous avons utilisé sur unity un package appelé “NetCode for GameObjects”. Ce package permet justement de mettre en place un réseau pour réaliser un jeu vidéo multijoueur. Il existe d’autres packages unity pour gérer le multi joueur comme “Photon” ou “Mirror”, cependant nous pensons que NetCode est plus adapté pour un réseau de type LAN. Pour mettre en place ce réseau, nous avons d’abord installé le package via le package manager de unity. Ensuite, nous avons créé un GameObject vide dans notre scène, que nous avons appelé “NetworkManager”. Dans ce dernier, nous avons ajouté un composant appelé Network Manager, qui permet justement de gérer les connexions des autres joueurs. Il nous a fallu ensuite gérer notre joueur. Nous l’avons déplacé dans un dossier appelé “prefabs” et l’avons supprimé de la scène. Nous avons alors pu mettre ce joueur comme prefab par défaut dans notre Network Manager. Enfin, nous avons ajouté un composant “NetworkObject” à notre joueur pour pouvoir le connecter au réseau. Maintenant, quand nous lançons notre projet, si nous cliquons sur “Start Host”, notre joueur apparaît sur l’écran, et nous pouvons le déplacer. Cependant, le rigidbody du joueur n’est plus associé aux autres éléments du jeu, car le joueur apparaît après le lancement du jeu. Une des difficultés que nous rencontrons à présent est de tester notre multijoueur, car il nous faut à chaque fois plusieurs joueurs. Une autre difficulté rencontrée a été de se renseigner pour mettre en place un début de multijoueur. Les ressources en ligne sont très variées, et il est difficile de se repérer lorsque l’on est débutant dans le domaine.

2.2 Programmation (Mattéo et Benjamin)

2.2.1 Défis rencontrés et solutions

La création d'un jeu vidéo au sein d'une entreprise est une entreprise complexe qui nécessite une coordination dans l'équipe. Les défis rencontrés sont nombreux, mais grâce à des solutions adaptées, il est possible de les surmonter. Voici une analyse détaillée des principaux problèmes rencontrés et des solutions mises en œuvre dans le cadre de notre projet de développement de jeu vidéo.

Manque de Connaissance de Unity au Sein de l'Équipe

Problème : Les membres de l'équipe n'avaient pas d'expérience avec Unity, la plateforme choisie pour le développement.

Solutions :

Documentation sur d'autres projets : En effet de nombreux jeux vidéos créés à l'aide de Unity sont disponibles et cela nous permet d'élargir nos connaissances et d'observer la manière dont le jeu à été codé.

Documentation Officielle : Consultation approfondie des ressources fournies par Unity et sur Internet pour mieux comprendre la plateforme.

Tutoriels en Ligne : Utilisation de tutoriels et de vidéos éducatives pour approfondir les compétences de base et technique sur Unity.

Difficultés dans la Mise en Place des Mécaniques de Jeu

Problème : L'implémentation des fonctionnalités de base, telles que le saut, les déplacements latéraux et une caméra dynamique, s'est avérée plus complexe que prévu.

Solutions :

Recherche de Scripts Préexistants : Analyse et adaptation de scripts disponibles pour comprendre leur fonctionnement et les intégrer au projet.

Tests Itératifs : Mise en place de cycles de tests réguliers pour identifier et corriger les problèmes liés aux mécaniques de jeu.

Documentation : De nombreuses documentations sont disponibles pour nous permettre de comprendre les principales mécaniques.

Visionnage de tutoriel : Enormément de tutoriel sont disponibles pour nous permettre de mieux comprendre les bases, ces tutos sont les bases de notre jeu il est donc

extrêmement de comprendre ces scripts.

Recherche de Thèmes pour les Niveaux

Problème : La conception de niveaux attrayants et cohérents avec l'univers du jeu.

Solutions :

Discussions entre membre de l'équipe : Des discussions ont eu lieu pour savoir si certains avaient des préférences et/ou des idées précises.

Mise en place des thèmes : Les thèmes mis en place sont en rapport avec le scénario déjà crée ce qui facilitent la recherche.

Mise en Place du Mode Multijoueur

Problème : Le développement d'un mode multijoueur a introduit des complexités supplémentaires, notamment en matière de synchronisation et de performance.

Solutions :

Documentation : En effet , de nombreuses documentations sont mis en place pour nous permettre de comprendre et faciliter l'implémentation du mode multijoueur qui est une des tâches les plus complexes du projet.

Visionnage de tutoriel : Même si de nombreuses documentations sont disponibles, les vidéos tutoriels sont plus explicites et nous permettre de mieux comprendre.

Développement d'une d'Ennemi Suivant le Joueur

Problème : La création d'un ennemi qui suit et interagit de manière réaliste avec le joueur a été un défi majeur.

Solutions :

Visionnage de tutoriel : Plusieurs vidéos tutoriel nous ont permis de comprendre les bases de ce type d'algorithme et ont facilité l'implémentation . Par exemple : récupérer la distance et l'angle entre le joueur et l'ennemi etc..

Consultation de la documentation : En effet la documentation nous a falicité l'implémentation d'un algorithme complexe comme celui-ci. Grâce à la documentation nous avons découvert l'existence de différentes classes dédiées à des tâches complexes.

L'écriture du scénario

Problème :Ecriture du scénario est un enjeu majeur lors de la création d un jeu vidéo. Il permet de se démarquer et d'être plus attrctif.

Solutions :

- Discussion de groupe : Discuter en groupe nous a permis de nous mettre d'accord sur quels aspects du scénario il était nécessaire de plus ou moins développer.
- Une histoire originale : En effet notre scénario s'inscrit sur une histoire originale avec le parcours des rêves ce qui permet à notre jeu d'être plus attractif.
- Plusieurs Plot-Twist : Notre scénario comporte de nombreux plot-twist.

Problèmes avec les scripts

Création de script : Lors de la création des premiers scripts, les membres de l'équipe ne connaissaient pas le comportement des scripts . Nous avons donc dû nous documenter.

Erreur de script : Au cours du codage de plusieurs nous avons rencontré plusieurs erreurs inconnues et nouvelles . La résolution de ces erreurs est aussi passée par la documentation. D'autre part certains étaient syntaxiquement corrects mais le comportement n'était pas celui attendu ce qui a rendu la tâche encore plus complexe.

Conclusion

Le développement de ce jeu vidéo a mis en lumière l'importance de la formation continue, de la collaboration interdisciplinaire et de l'adaptabilité face aux défis techniques et créatifs. Grâce à une approche structurée et à l'engagement de l'équipe, nous avons transformé ces obstacles en opportunités d'apprentissage, renforçant ainsi notre expertise collective et la qualité du produit final.

Conclusion

Les différentes tâches de ce projet que nous avons réalisées depuis la soutenance méthodologique sont donc surtout des tâches conceptuelles, dans le sens où elles n'ont pas directement impliqué une certaine implémentation à un niveau technique, à ce stade du projet. Néanmoins, nous cherchons vraiment à visualiser ce à quoi notre jeu va ressembler, avant de donner une priorité à la programmation.

Nous avons vu que notre jeu s'inspire de différents autres jeux de plateformes, de plus des classiques du genre (tels que Dead Cells, Hollow Knight ou Super Mario Bros). Cette inspiration se traduit par le reprise des mécaniques propres aux jeux de plateformes (se déplacer de gauche à droite, sauter), mais également d'éléments plus spécifiques, comme un effet de parallaxe que nous prévoyons d'implémenter, et qui est présent dans de nombreux jeux du genre. Cependant, nous voulons tout de même créer un jeu original et amusant, avec des décors et une histoire unique.

Le scénario du jeu, réalisé par Alexandre Abbruzzese et Oren Guetta, présente le récit de Milo et d'Elena, à l'intérieur des rêves de celle-ci. La mécanique des rêves est essentielle au jeu, car il s'agit du point d'ancrage de l'histoire et aussi ce qui rend le jeu original. Les rêves permettent de représenter des paysages fantastiques et nous permettent de faire preuve de créativité dans la conception des niveaux. Une des difficultés rencontrées fut de devoir raconter une histoire à caractère émotionnel tout en restant dans le cadre du projet scolaire. Effectivement, pouvoir mesurer et quantifier une tâche de telle manière à ce qu'elle puisse être réalisée dans les temps est une capacité cruciale dans ce projet, il faut donc parfois faire des compromis sur certaines parties de l'histoire par exemple.

Pour la programmation, nous avons posé les bases de la programmation multijoueur en mettant en place un Network Manager pour gérer les connexions des joueurs. Ceci nous a permis d'avoir un début de multijoueur. Cependant il est difficile de tester ce que l'on a fait à ce niveau car il faut plusieurs joueurs connectés au même réseau. De plus, le multijoueur présentera d'autres difficultés que nous rencontrerons plus tard dans l'année, liées à la synchronisation entre les joueurs, et à d'autres aspects de la programmation du jeu. Plus spécifiquement, il nous faudra décider comment connecter

deux joueurs entre eux lorsque le jeu est lancé : il est possible de faire un système d’ami pour se connecter, comme dans Fortnite, ou alors faire un système de code généré aléatoirement que l’on crée pour faire en sorte de rejoindre un autre joueur, comme dans Kahoot. Néanmoins ces aspects n’ont pas été encore abordés.

Enfin, en terme de programmation solo, nous avons pour l’instant réussi à mettre les mouvements de base (se déplacer et sauter). Il restera cependant beaucoup à faire à ce niveau, et aussi sur tout ce qui concerne les graphismes, les sprites des personnages, les décors... et que nous n’avons pas encore fait.

Globalement, nous nous sentons un peu perdus par rapport au projet. Nous sommes des débutants dans le domaine de la création de jeux vidéos, et nous avons du mal à respecter la planning que nous avons fait et à finir les tâches dans les temps. Comme nous ne savons pas si tel ou tel élément prend du temps ou pas à faire, notre organisation en pâtit. Actuellement, la majorité du temps que nous passons sur projet consiste à concevoir des éléments sans les implémenter et aussi faire des recherches sur l’utilisation de unity ainsi que sur la programmation du jeu, et cela prend beaucoup de temps. Cela peut parfois être décourageant pour certaines personnes du groupe, mais les efforts finissent par payer. Notre problème d’organisation n’est pas ciblé et concerne tout le groupe. Il est probablement causé par un problème de communication et de compréhension de ce que l’on doit faire. Pour remédier à ce problème, la solution la plus efficace est de plus communiquer, ainsi que passer plus d’heures sur le projet, et prendre plus d’avance sur les tâches à accomplir. Effectivement, nous avons tendance à faire certaines tâches très tard par rapport à ce qui était prévu de base. C’est un problème dont nous avons conscience et nous allons faire des efforts pour y remédier.

Sources

2.3 Scénario :

<https://www.gamedeveloper.com/design/the-evolution-of-video-games-as-a-storytelling-medium-and-the-role-of-narrative-in-modern-games>

<https://objectif-scenario.fr/voyage-du-heros/>

<https://www.jeuxvideo.com/forums/42-31-46706943-1-0-1-0-comment-ecrire-un-scenario-de-jeu-video.htm>

2.4 Programmation réseau :

<https://docs.unity.com/>

<https://docs-multiplayer.unity3d.com/netcode/current/tutorials/get-started-ngo/>

<https://fr.sharpcoderblog.com/blog/building-multiplayer-networked-games-in-unity>

2.5 Programmation solo :

<https://stackoverflow.com/questions/25350411/unity-2d-jumping-script>

<https://www.youtube.com/watch?v=hQE8lQk9ikE> (Design Patterns)