

Les Cœurs Brisés

Projet Ascension

Rapport de soutenance



Sommaire

Table des matières

1. INTRODUCTION

2. TÂCHES RÉALISÉES

3. AVANCEMENT DU PROJET

3.1. Avances/ Retards

4. LES TÂCHE À VENIR

5. CONCLUSION

1. INTRODUCTION

Ascension est un jeu vidéo de plateforme qui met en scène un personnage joueur progressant à travers un univers montagneux. Inspiré par le documentaire Kaizen publié il y a quelques mois par le youtubeur Inoxtag, ainsi que par des jeux tels que Celeste ou Bread & Fred, nous avons fait de ces références notre principale source d'inspiration.

Depuis notre première soutenance, le projet a connu des avancées significatives, notamment sur le plan fonctionnel et technique. Nous avons peaufiné les mécaniques de jeu afin d'améliorer l'expérience du joueur. L'objectif reste inchangé : gravir une montagne en franchissant plusieurs niveaux semés d'obstacles et d'ennemis, rendant l'ascension progressive et stimulante.

Lors de la première présentation, nous avions principalement axé notre travail sur les fonctionnalités de base, mettant temporairement de côté l'aspect graphique. Aujourd'hui, nous avons commencé à intégrer des éléments visuels plus développés et à concevoir une direction artistique cohérente qui reflète notre univers de jeu.

Par ailleurs, nous prévoyons toujours d'intégrer des intelligences artificielles dans la programmation et le game design afin d'affiner le comportement des ennemis et enrichir l'interactivité du jeu. Cette étape, qui interviendra dans les mois à venir, permettra de rendre le gameplay plus dynamique et engageant.

Notre équipe, Les Cœurs Brisés, continue de travailler avec rigueur et passion. Grâce à une organisation efficace et à une répartition claire des tâches, nous avons pu avancer sereinement dans le développement de notre jeu sur Unity. Voici un récapitulatif des rôles de chacun :

- Yazid ADGHOGHI : programmation des déplacements du joueur, du multijoueur, et de la création des niveaux
- Maxime ELOPHE : game design et création des menus
- Jean-Baptiste GUÉRIMAND : conception des camps de base entre les niveaux, comportement des IA ennemis, Interfaces d'achats et d'utilisations des objets/bonus.
- Ugo BOLDRON : scénario et narration, création des niveaux

Chaque membre continue d'apporter son expertise tout en collaborant pour assurer la cohésion du projet.

Nous estimons actuellement l'avancement de notre projet à environ 40%. Malheureusement, nous sommes pas satisfait du rendu à cause de quelques retards accumulés , mais nous nous sentons capable de doubler d'efforts pour que le projet puisse aboutir a un rendu intéressant. L'objectif est de proposer une expérience immersive et satisfaisante qui répondra à nos attentes et à celles de nos futurs joueurs.

Nous avons hâte de vous présenter la suite de notre travail et d'obtenir vos retours pour continuer à améliorer Ascension.

2. TÂCHES RÉALISÉES

Chacun des membres de notre équipe à contribuer à la création du jeu en respectant le tableau d'avancement réalisé au préalable (voir 3. Avancement du projet.) ainsi que la répartition des tâches en suivant le tableau ci-dessous.

Tâche	Yazid ADGHOGHI	Ugo BOLDRON	Maxime ELOPHE	Jean-Baptiste GUERIMAND
Programmation				
Déplacement du joueur	S	R		
Comportement des IA ennemis			S	R
Gestion et effets des objets/bonus		S	R	
Multijoueur/interactions entre joueurs	R			S
Game Design				
Conception des niveaux		S	R	
Conception des camps de bases (entre les niveaux)		S		R
Interfaces d'achats et d'utilisations des objets/bonus		S	R	
Scénario	S	R		
Animations des évènements		R	S	
Sound design	R			S
Site Web				
html	S			R
css	R			S

R = Responsable S = Suppléant

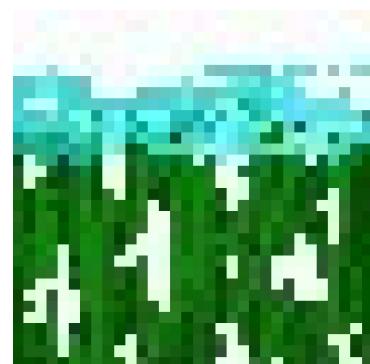
Ugo BOLDRON :

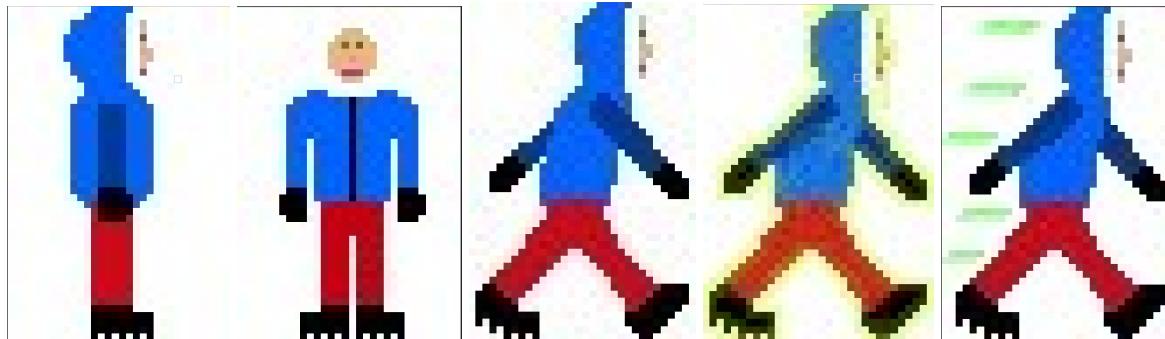
Le rôle de Ugo BOLDRON pour la première était d'écrire le scénario du jeu au complet avec les moindres détails et à quel moment du jeu cela allait se produire. Le jeu étant basé sur le projet d'ascension de l'Everest de Inoxtag, Ugo BOLDRON a donc re visionné le documentaire d'Inoxtag pour l'aider à créer les différents niveaux avec leurs différents obstacles et pour s'inspirer au maximum de la vie réelle. Pour commencer, Ugo a commencé à faire un grand croquis avec une grande montagne (voir l'image ci-dessous) pour nous donner une idée de l'apparence du jeu et du chemin à faire pour arriver au sommet. Puis il a commencé à rentrer dans les détails en faisant plein de croquis des différents niveaux, avec des événements présents dans l'Ascension d'Inoxtag, mais aussi avec les événements qu'il a imaginé lui-même . Il y a donc un scénario précis pour chaque niveau et cela peut donc influencer les niveaux suivants. Une fois cela fait, Ugo BOLDRON a donc expliqué qu'après chaque niveau il y avait un camp de base où on pouvait se reposer pour pouvoir attaquer le prochain niveau. Mais dans ces camps de base on peut également acheter un objet qui va nous servir pour terminer le prochain niveau. Ugo BOLDRON a donc également réfléchi à comment le joueur va faire pour s'orienter dans le jeu. Il a donc écrit des petits textes comme «utiliser l'échelle pour franchir la crevasse» pour l'aider. Ces informations seront importantes pour le joueur comme ça il sera ce qu'il faut faire et il aura donc toutes les cartes en main pour pouvoir passer le niveau. Grâce à ce scénario nous savons donc les événements de chaque niveaux et nous pourrons donc créer les niveaux sur Unity tout en suivant le scénario.



Croquis représentent la grande montagne avec les premières idées du scénario (à gauche) et deux croquis de différents niveaux (à droite)

Pour la deuxième soutenance Ugo avait pour but de réaliser différents assets, des animations pour le Dash, et pour le Boost de vitesse mais également améliorer le saut faire la cinématique de début et les animations entre les premiers niveaux. Pour créer les assets UGO à utilisé l'application PixiEditor





Yazid ADGHOGHI :

Yazid ADGHOGHI avait plusieurs missions :

La gestion des déplacements:

La gestion des déplacements des joueurs, un aspect essentiel dans le développement d'un jeu vidéo. Au début de son travail, il avait choisi d'utiliser la classe KinematicObject pour concevoir ce système de mouvement. Cependant, il a rapidement rencontré des problèmes liés à la documentation de cette classe, ce qui a rendu l'implémentation plus difficile que prévu. Face à ces complications, Yazid ADGHOGHI a décidé de modifier son approche en optant pour la classe MonoBehaviour. Cette classe est non seulement mieux documentée, mais elle s'avère également plus appropriée pour répondre aux besoins spécifiques de son projet.

L'implémentation du système de mouvement a présenté divers défis, en particulier concernant le mécanisme de saut. Au départ, le personnage ne sautait pas de manière naturelle et avait tendance à léviter dans les airs. Ce problème a conduit Yazid ADGHOGHI à réaliser qu'il ne pouvait pas se fier uniquement à un système basé sur la gravité. En conséquence, il a pris la décision de développer une méthode alternative afin d'optimiser le saut du personnage. Au cours de plusieurs tests, il a été en mesure d'identifier les valeurs idéales pour différents paramètres de déplacement. Après des ajustements et des essais, il a conclu qu'une vitesse

(Speed) de 2 et une force de saut (Jumpforce) de 3 permettaient d'obtenir un équilibre satisfaisant, offrant ainsi une expérience de jeu fluide et agréable.

Par la suite, le système de mouvement a été amélioré avec l'ajout de nouvelles fonctionnalités. Un mécanisme de dash a été intégré, permettant aux joueurs d'effectuer des déplacements rapides sur de courtes distances. Ce type de mouvement dynamique ajoute une dimension supplémentaire au gameplay, rendant les interactions plus intéressantes. De plus, pour accroître encore l'expérience de jeu, un système de bonus de vitesse temporaire a été introduit. Ce système permet aux joueurs d'accélérer significativement leurs déplacements durant une période limitée, ajoutant ainsi une couche stratégique au jeu, car les joueurs doivent savoir quand utiliser ces boosts pour maximiser leur efficacité.

En ce qui concerne l'optimisation des performances, Yazid ADGHOGHI a décidé qu'il n'était pas nécessaire de travailler spécifiquement sur les mouvements rapides comme le dash. En effet, après avoir testé le système, il a constaté que celui-ci fonctionnait de manière satisfaisante dans son état actuel, sans latence ou ralentissement. Cela a permis de garantir que les joueurs puissent profiter des fonctionnalités de mouvement rapide sans compromettre la fluidité du jeu.

```
void Sauter()
{
    if (jumpTimer <= 0 && (Input.GetKeyDown(KeyCode.Space) || Input.GetKeyDown(KeyCode.Z) || Input.GetKeyDown(KeyCode.UpArrow))
    {
        transform.position += Vector3.up * jumpForce;
        jumpTimer = jumpCooldown;
    }
    if (jumpTimer > 0) jumpTimer -= Time.deltaTime;
}
```

Fonction de saut

```

↳ Frequently called  ↳ 1 usage  ↳ xyubot +1
void Deplacement()
{
    Vector3 move = Vector3.zero;
    if (Input.GetKey(KeyCode.Q) || Input.GetKey(KeyCode.LeftArrow))
    {
        move -= transform.right;
        if (Input.GetKey(KeyCode.F))
        {
            transform.position += transform.right * (-5 * (speed * Time.deltaTime));
            return;
        }
    }
    if (Input.GetKey(KeyCode.D) || Input.GetKey(KeyCode.RightArrow))
    {
        move += transform.right;
        if (Input.GetKey(KeyCode.F))
        {
            transform.position += transform.right * (5 * (speed * Time.deltaTime));
            return;
        }
    }
    transform.position += move * ((isDashing ? speed + dashSpeed : speed) * Time.deltaTime);
}

```

Fonction de déplacement

```

↳ Frequently called  ↳ 1 usage  ↳ xyubot
void HandleDash()
{
    if (Input.GetKeyDown(KeyCode.DownArrow) && !isDashing) (isDashing, dashTimer) = (true, dashDuration);
    if (isDashing)
    {
        dashTimer -= Time.deltaTime;
        isDashing = dashTimer > 0;
    }
}

```

Fonction de fonctionnement du dash



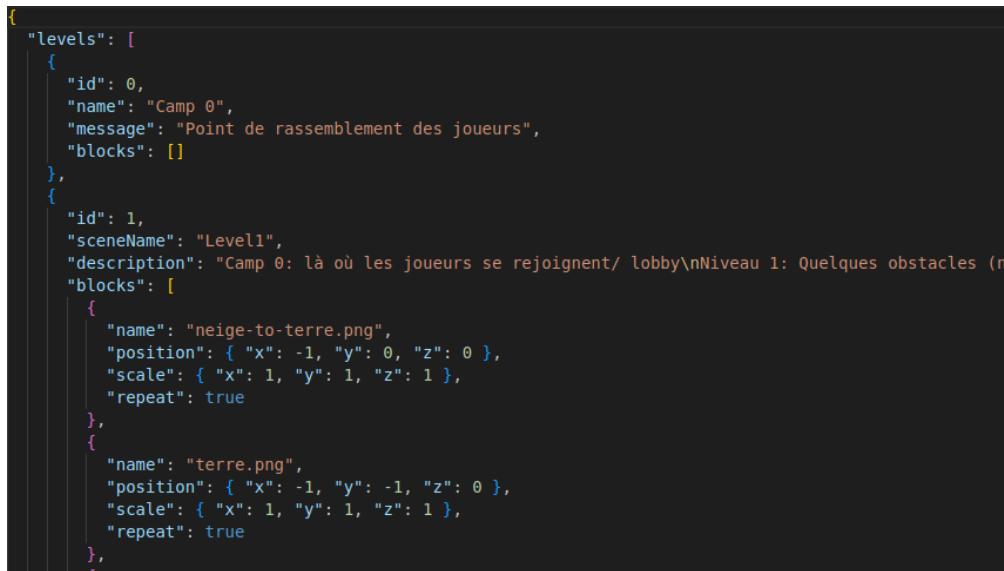
Assignment des variables

Génération de Niveaux avec JSON :

L'utilisation de JSON pour structurer les niveaux est une excellente idée, offrant une grande souplesse et facilitant les modifications sans altérer le code source. Cette approche permet également une meilleure gestion des données et une intégration facilitée avec d'autres outils ou systèmes.

Concernant ce que j'ai pu accomplir, la structure est claire et sérialisable. Vos classes Block, LevelData et LevelList sont bien définies et s'intègrent facilement avec JSON. Le chargement est efficace grâce à la méthode LoadLevels(), qui permet d'importer les niveaux en toute fluidité depuis le fichier JSON dans StreamingAssets. La gestion des répétitions est optimisée avec la fonction Repeat0(), qui simplifie la création de blocs récurrents, un atout pour des plateformes ou éléments répétitifs. L'instanciation et la configuration sont dynamiques : la méthode CreateBlocks() assure la création des objets, l'application des textures et la configuration des propriétés physiques. Enfin, le système est extensible, offrant la possibilité d'ajouter facilement de nouveaux types de blocs ou de mécanismes grâce à la flexibilité du JSON.

Il existe cependant quelques pistes d'amélioration. La paramétrisation des répétitions pourrait être plus avancée, car actuellement Repeat0() est limitée à 20 copies. Ajouter un attribut repeatCount dans le JSON offrirait une plus grande flexibilité. Les blocs pourraient proposer davantage de variantes graphiques afin d'enrichir la diversité visuelle. L'intégration de propriétés avancées, telles que la forme, la taille, la rotation ou le mouvement, améliorerait également le dynamisme du jeu. Enfin, un éditeur visuel de niveaux permettrait de simplifier la conception des environnements en générant automatiquement le JSON.



```
{
  "levels": [
    {
      "id": 0,
      "name": "Camp 0",
      "message": "Point de rassemblement des joueurs",
      "blocks": []
    },
    {
      "id": 1,
      "sceneName": "Level1",
      "description": "Camp 0: là où les joueurs se rejoignent/ lobby\nNiveau 1: Quelques obstacles (neige et terre)",
      "blocks": [
        {
          "name": "neige-to-terre.png",
          "position": { "x": -1, "y": 0, "z": 0 },
          "scale": { "x": 1, "y": 1, "z": 1 },
          "repeat": true
        },
        {
          "name": "terre.png",
          "position": { "x": -1, "y": -1, "z": 0 },
          "scale": { "x": 1, "y": 1, "z": 1 },
          "repeat": true
        }
      ]
    }
  ]
}
```

Début du fichier .json et premiers blocs placés

Mode Multijoueur avec Photon :

L'intégration de Photon PUN pour le mode multijoueur est une approche pertinente, simplifiant la gestion de la connexion, de la communication et de la synchronisation. Votre implémentation gère correctement la connexion à Photon et l'instance des joueurs dans la salle "Salle1", avec un code clair et concis. Toutefois, plusieurs améliorations peuvent être envisagées. Plutôt que des positions de spawn aléatoires, il serait plus intéressant de définir des points précis dans le JSON. La synchronisation des mouvements pourrait être améliorée en utilisant PhotonView et IPunObservable afin d'assurer une fluidité optimale dans les déplacements et les animations des joueurs. La gestion des déconnexions pourrait être optimisée en affichant un message, en supprimant le joueur déconnecté ou en mettant la partie en pause. Une interface utilisateur pour le lobby permettrait aux joueurs de voir les autres connectés, de choisir un niveau et de démarrer la partie ensemble. Un système de "Master Client" pourrait être mis en place, désignant le premier joueur connecté comme responsable du chargement et de la synchronisation du niveau. Enfin, l'ajout d'un chat basique améliorerait la communication entre les joueurs.

La synergie entre la génération de niveaux et le mode multijoueur pourrait enrichir l'expérience de jeu. Le "Master Client" pourrait générer un niveau aléatoire à partir des données JSON en début de partie. Il serait également intéressant d'ajouter des événements interactifs dans le JSON, déclenchant des actions en multijoueur, comme l'ouverture d'une porte lorsqu'un objectif est atteint par tous les joueurs.

N'hésitez pas à poser des questions plus spécifiques sur certains aspects de votre projet, tels que la synchronisation des animations, la gestion des collisions en

multijoueur ou la création d'un éditeur de niveaux. Je serais ravi de vous apporter des conseils et des exemples concrets adaptés à vos besoins.

```
public class NetworkManager : MonoBehaviourPunCallbacks
{
    void Start()
    {
        Debug.Log("Connexion à Photon...");
        PhotonNetwork.ConnectUsingSettings();
    }

    public override void OnConnectedToMaster()
    {
        Debug.Log("Connecté au serveur Photon !");
        PhotonNetwork.JoinLobby(); // Rejoint automatiquement un lobby
    }

    public override void OnJoinedLobby()
    {
        Debug.Log("Connecté au lobby !");
        PhotonNetwork.JoinOrCreateRoom("Salle1", new RoomOptions { MaxPlayers = 4 }, TypedLobby.Default);
    }

    public override void OnJoinedRoom()
    {
        Debug.Log("Joueur rejoint la salle !");
        Vector3 spawnPos = new Vector3(Random.Range(-5f, 5f), 0, 0);
        PhotonNetwork.Instantiate("Player", spawnPos, Quaternion.identity);
    }
}
```

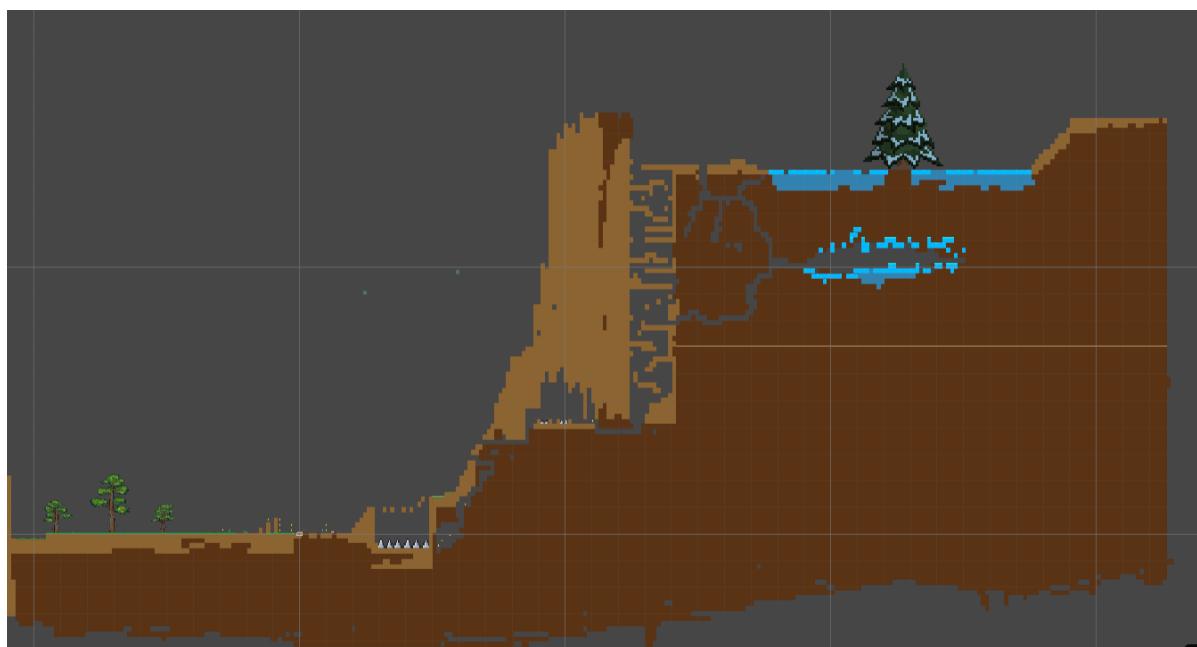
Script NetworkManager.cs

Maxime ELOPHE :

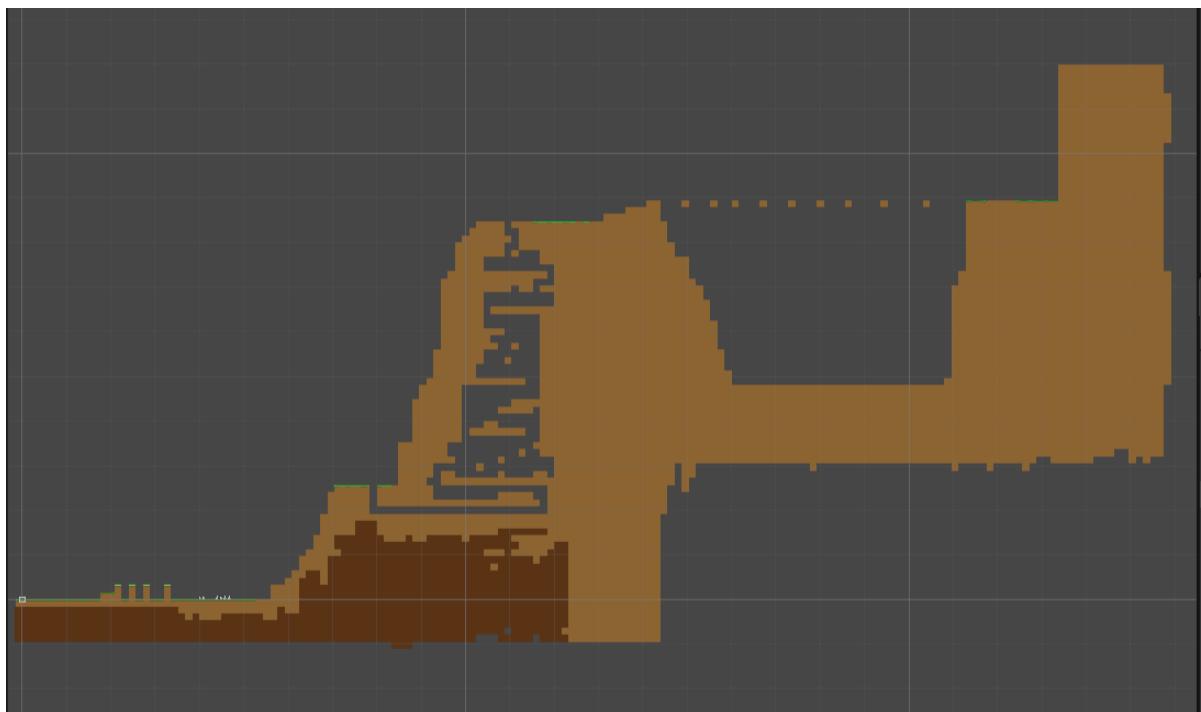
Maxime ELOPHE avait pour mission de réaliser à hauteur de 10 à 20 pourcents la conception des niveaux, ce qui équivaut au premier niveau. A la suite de notre décision de passer sur Unity au lieu de Godot, Maxime ELOPHE a décidé pendant les vacances de Noel de se familiariser avec Unity en regardant des playlists de vidéos sur Youtube de création de petit jeu complet sur Unity, c'est en reproduisant soi-même ce qu'il a vu dans les vidéos sur Unity qui a permis à Maxime ELOPHE d'apprendre les bases du fonctionnement d'Unity. C'est sur la partie de la création du niveau que Maxime ELOPHE s'est concentré car c'était en lien avec sa tâche. Cette partie lui a appris à créer et à manipuler une tile palette et une tilemap, et avec les assets données en lien sous ces vidéos, cela lui a permis de "dessiner" le niveau avec les mêmes éléments présentés dans les vidéos: en posant des blocs, en créant des obstacles, des pièges.

Après s'être familiarisé avec Unity, Maxime ELOPHE a décidé de commencer à créer un niveau. Pour cela il a décidé de regarder dans la boutique des assets de unity pour voir s'il y a des assets en 2D qui correspondent aux besoins du jeu Ascension, donc d'un environnement de montagne. Il a trouvé des assets qui sont toutes gratuites ,qui lui permettent de poser des blocs de terre, de glace, des pièges comme des pics mais aussi d'ajouter des éléments d'arrière fond comme des sapins enneigés, des panneaux enneigés, et même des objets comme des pièces. Avec

tous ces assets en main et avec toute son inspiration, il a commencé par créer une première scène qui compose le premier niveau. Au début le niveau présente quelques sauts simples, cela permet aux joueurs d'apprendre à sauter correctement, le joueur est ensuite testé avec un saut qu'il doit effectuer par-dessus des pics. Par la suite une section du niveau lui permet de perfectionner les sauts simples qu'il a appris avec une grande montée dans les grottes de la montagne. Enfin une section finale permettra au joueur après qu'on est implémentée les ennemis, de rencontrer des loups, ses premiers ennemis. Avec mon inspiration et avec le temps qui m'était permis j'ai décidé de tester de nouvelle chose avec la création d'une seconde scène ou cette fois ci les loups seraient mis au début du niveau en même temps que le joueur commence à apprendre à sauter, la section suivante est identique à celle de la première scène avec une grande montée dans une grotte de la montagne qui permet aux joueurs de perfectionner le saut précédemment appris. Enfin une section finale différente de la première scène apparaît ou au lieu d'avoir une zone où le joueur côtoie ses premiers ennemis, une nouvelle zone de sauts cette fois-ci en hauteur apparaît, ici si le joueur tombe il devra remonter en haut pour recommencer son parcours, cette section finale peut être considéré comme un test des sauts que le joueur accompli.



Premier prototype de niveau

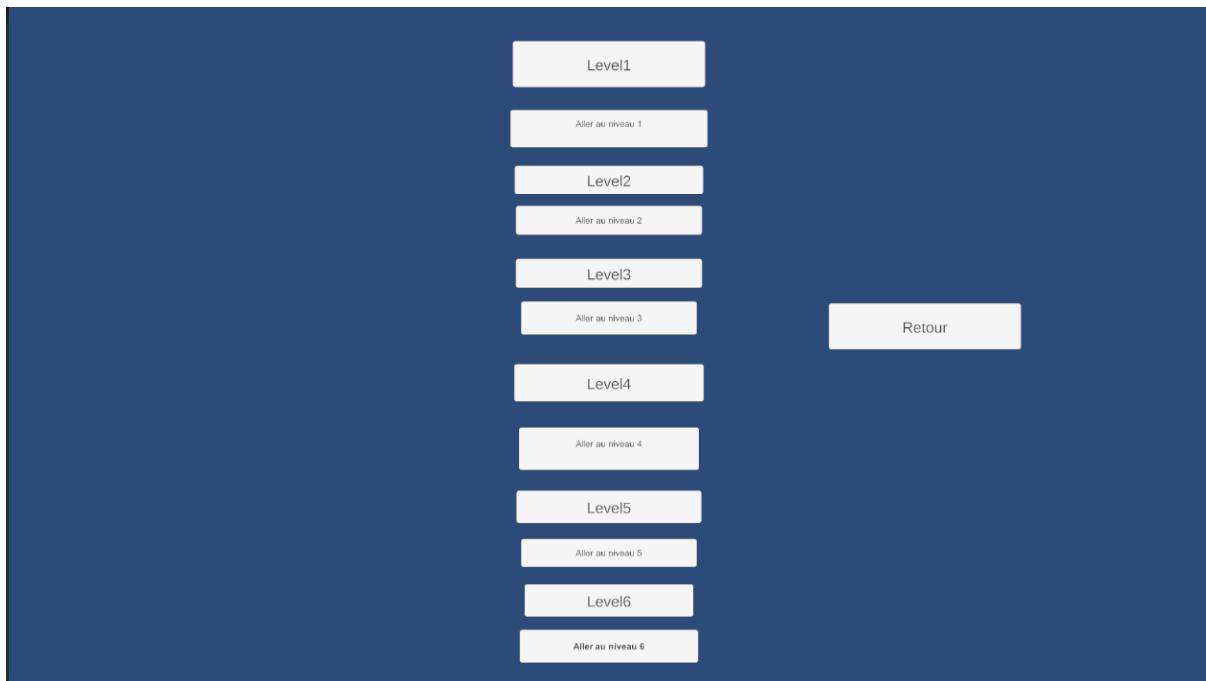


Deuxième prototype de niveau

Pour la deuxième soutenance, Maxime avait pour mission initialement de continuer la conception des niveaux, la gestion des effets des objets bonus et l'interface d'achats et d'utilisation des objets/bonus mais en raison d'une répartition des tâches mals équilibrés et en fonction de l'avancement effectuées sur nos tâches respectives, nous avons décidés de modifier la répartition des tâches. Avec cette nouvelle répartition Maxime avait pour tâche de faire un menu avec boutons: le bouton "Commencer" qui lance le niveau en cours, le bouton "Quitter" qui éteins le jeu, le bouton "Sélection des niveaux" qui amène à un autre menu où on peut sélectionner les niveaux déjà réussis, un bouton "Contrôles" amène à un menu listant les contrôles du jeu. Maxime s'est concentrée sur le côté fonctionnelle du menu et non sur le côté esthétique, d'où par exemple ,le fond bleu présent sur le menu qui sera changé d'ici la prochaine soutenance pour un fond reflétant le jeu. Le bouton "Contrôles" n'amène pour le moment à nulle part, car nous n'avons pas encore réfléchie aux touches du jeu. D'ici à la prochaine soutenance, Maxime doit donc faire la partie esthétique du menu.



Prototype du menu principal



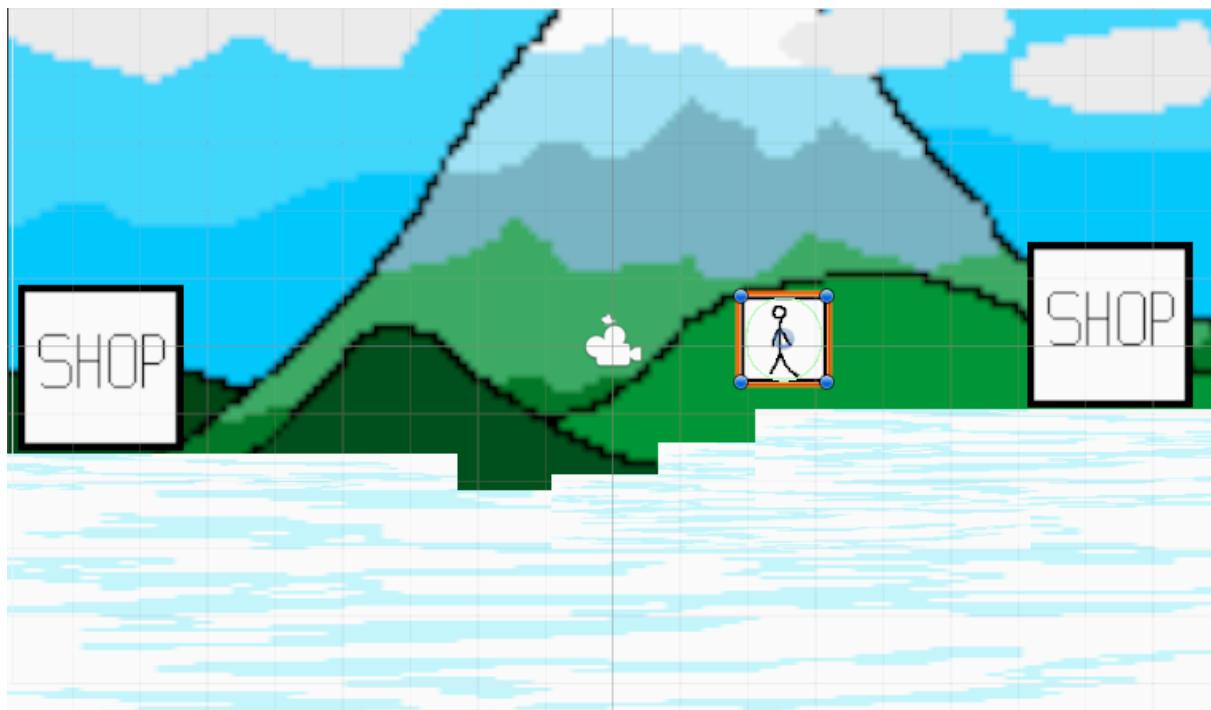
Prototype du menu “Sélection des niveaux”

Jean-Baptiste GUERIMAND :

Jean-Baptiste GUERIMAND avait quant à lui pour tâche de commencer la conception des camps de base et plus précisément d'en réaliser environ 40%. Pour cela, il a tout d'abord dû se familiariser avec Unity, le moteur de jeu qu'il devait utiliser. Afin de bien comprendre son fonctionnement et ses spécificités, il a visionné divers tutoriels et consulté plusieurs documentations sur le sujet. Ces premières étapes d'apprentissage lui ont permis d'acquérir une base solide et de commencer à expérimenter avec l'outil en créant de petits jeux suivant les instructions des tutoriels.

Une fois cette phase d'initiation passée, il a pu entrer dans le vif du sujet en réalisant une première scène servant d'exemple pour tous les camps de base à venir. Dans cette scène, il a tout d'abord placé quelques blocs qui serviraient de sol et a intégré un personnage jouable avec des déplacements basiques. Afin d'enrichir l'environnement, il a ensuite ajouté des échoppes sur lesquelles il a appliqué un script permettant au joueur d'accéder à une interface d'achat d'objets. Ce système fonctionnait de manière simple : lorsque le personnage entrait en collision avec l'échoppe, une interface d'achat devait s'ouvrir. Cependant, cette interface ne devait être réalisée que plus tard dans le projet, une fois que les autres aspects fondamentaux du jeu auraient été mis en place.

Bien que cela ne figurait pas initialement dans le tableau d'avancement, Jean-Baptiste a également dû s'atteler à la création d'un site web en HTML et CSS afin de répondre aux attentes de la soutenance. Ce site devait permettre de présenter le projet et l'équipe en reprenant des éléments du cahier des charges. Il a ainsi créé plusieurs pages interconnectées, dont une page dédiée au téléchargement du rapport et une autre qui, à terme, devait proposer le téléchargement du jeu finalisé. Le développement du site web s'est révélé être un défi supplémentaire, nécessitant de résoudre divers problèmes et bugs afin d'améliorer l'aspect visuel et l'ergonomie de la plateforme. Finalement, les seuls éléments manquants étaient les espaces de téléchargement des derniers rapports de soutenance ainsi que du jeu lui-même.



Scène du camp de base



Scène d'interface d'achat

Parallèlement à la finalisation du site web, Jean-Baptiste a grandement contribué à la conception des différents assets graphiques du jeu. Il a notamment travaillé sur la création de blocs en 32x32 pixels qui allaient constituer l'identité visuelle de l'univers du jeu. Grâce à cette approche, il a pu composer une palette de neuf blocs différents, permettant une variété d'environnements comprenant de la terre, de

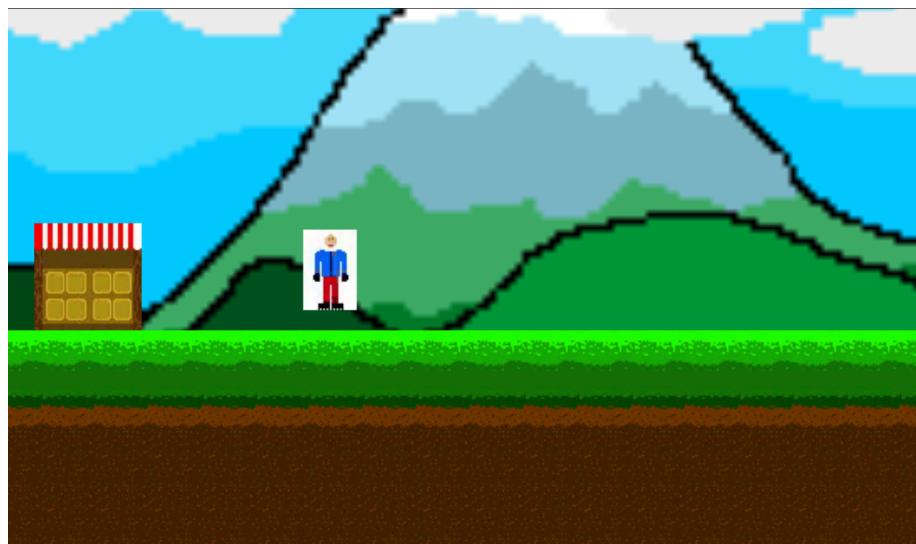
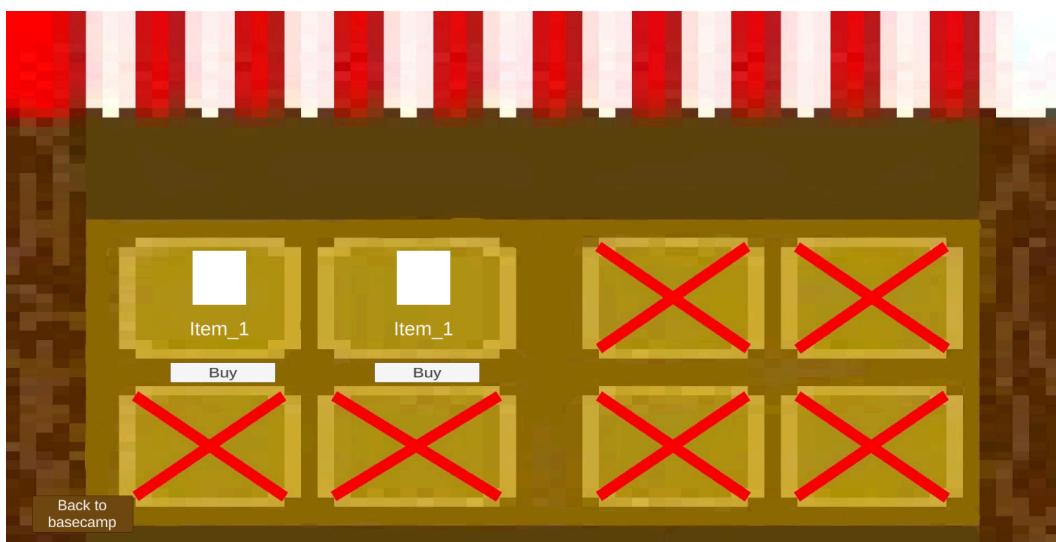
l'herbe, de la pierre, de la neige et de la glace. Ces éléments graphiques ont ensuite été utilisés pour reconstruire un camp de base plus esthétique et moins expérimental que la première version.

L'interface d'achat des objets a également bénéficié d'une amélioration visuelle. Le fond de cette interface a été modifié pour mieux correspondre à l'ambiance des échoppes du jeu. Ce travail sur l'interface visait à rendre l'expérience utilisateur plus immersive et agréable.

Enfin, Jean-Baptiste a progressé dans le développement du comportement des ennemis. Il a mis en place un script de déplacement simple permettant aux ennemis de toujours se diriger vers le joueur. Le tout en récupérant la valeur de la distance entre le joueur et l'ennemi ce qui permettra de faire des dégâts au joueur lorsque cette distance diminue. Cela permet de ne pas avoir besoin de gérer les collisions entre le joueur et l'ennemi et facilite en quelque sorte la tâche.

En conclusion, le travail accompli par Jean-Baptiste GUERIMAND dans le cadre du projet a été essentiel à plusieurs niveaux. Son investissement dans la conception des camps de base, le développement des mécaniques de jeu, la création des assets graphiques et l'élaboration du site web a permis d'avancer dans le projet. Grâce à ses efforts, les bases du jeu ont pu être posées de manière efficace et évolutive, offrant ainsi une plateforme propice aux futures améliorations et ajouts de contenu. Ce travail, réalisé en parallèle des autres membres de l'équipe, a contribué à la progression globale du projet et à la concrétisation d'une vision cohérente et aboutie.

Cependant, le projet a malgré tout pris un certain retard non négligeable, en effet, suite à certains imprévus et difficultés rencontrées par certains membres de l'équipe, des tâches ont dû être redistribuées et des membres, tels que Jean-Baptiste se sont retrouvés avec des tâches supplémentaires qui malheureusement n'ont pas pu être réalisées dans les temps. La gestion des effets et bonus et la tâche la plus touchée par ce problème, 50% aurait dû être atteint lors de la deuxième soutenance mais aucun avancement n'a été réalisé.

*Scène du camp de base**Scène d'interface d'achat*

3. AVANCEMENT DU PROJET

Tâche	Soutenance	Première : du 13 au 17 janvier 2025	Deuxième : du 10 au 14 mars 2025	Dernière : du 26 au 30 mai 2025
Programmation	10-20%	40-45%	100%	
Déplacement du joueur	50-75%	100%	100%	
Comportement des IA ennemis	0%	20%	100%	

Gestion et effets des objets/bonus	0%	50%	100%
Multijoueur/interactions entre joueurs	0%	0-10%	100%
Game Design	25-30%	75%	100%
Conception des niveaux	10-20%	70%	100%
Conception des camps de bases (entre les niveaux)	40%	100%	100%
Interfaces d'achats et d'utilisations des objets/bonus	0%	75%	100%
Scénario	100%	100%	100%
Animations des évènements	0%	50%	100%
Sound design	0%	50%	100%
Site Web	0%	50%	100%
html	0%	50%	100%
css	0%	50%	100%

3.1. Avances/ retards

Jusqu'à la première soutenance nous avons réussi à rester dans les temps malgré les différents problèmes que nous avons eu avec le moteur de jeu Godot. En effet, nous manquons de documentation sur ce moteur de jeu. C'est-à-dire qu'il n'existe que très peu de sites Web, possédant de la documentation Godot. Mais également très peu de vidéos dans lesquelles Godot est expliqué, analysé et détaillé. C'était la première fois que l'on utilisait ce genre de logiciel totalement nouveau pour nous. En effet, sachant que nous n'avions que très peu voir pas du tout d'expérience dans la création de jeu vidéo ou plus précisément dans l'utilisation de moteurs de jeu, nous avions prévu qu'il allait nous falloir du temps pour prendre en main ces nouveaux outils et avons donc rempli le tableau d'avancement en ne prévoyant de réaliser que

très peu de tâches lors de cette première période. Nous devions donc d'abord réaliser le scénario, avancer sur les déplacements du joueur car ils nous aideront à tester nos niveaux et avancer sur la conception du premier niveaux et le fonctionnement des camps de base. Nous avons donc pu réaliser ces quelques tâches dans les temps après avoir passé plusieurs semaines à tenter de régler un problème de moteur de jeu puis quelques semaines supplémentaires pour se familiariser avec le moteur de jeu Unity nous avons pu suivre notre tableau d'avancement.

Mais lors de cette deuxième période de travail sur le projet, les choses ne se sont malheureusement pas déroulées de la même façon et nous avons par conséquent accumulé du retard. En effet, nous avons rencontré des difficultés au sein du groupe dans différentes tâches que certains membres devaient réaliser, et à cause de contre-temps, nous avons pris la décision d'alléger le travail de certains pour le rajouter à d'autres. Mais ces derniers n'ont malheureusement pas pu assumer la charge de travail qui leur a été ajoutée et n'ont donc pas réussi à accomplir toutes leurs tâches dans les temps impartis.

Nous avons eu une mauvaise gestion des tâches à effectuer lors de cette deuxième période de travail mais cela nous a permis d'apprendre à ne pas reproduire ces erreurs. Nous sommes donc motivés à rattrapper notre retard pour pouvoir terminer notre jeu dans les temps.

4. TÂCHES À VENIR

Pour la troisième soutenance technique qui aura lieu la semaine du 26 au 30 mai 2025, la programmation devra être terminée. Les déplacements du joueurs devront être terminés, c'est-à-dire que le joueur pourra sauter, reculer et avancer, sans le moindre bug. Le responsable de cette tâche est Ugo BOLDRON avec Yazid ADGHOGHI comme suppléant. Le comportement des IA ennemis devra être fini, l'IA devra donc être un minimum maîtriser pour se familiariser avec cette intelligence, le responsable de cette tâche est donc Jean-Baptiste GUERIMAND, avec Maxime ELOPHE comme suppléant. La gestion des effets et objets devra être entièrement réalisé également. Les objets auront donc une programmation pour qu'ils puissent appliquer leur rôle. Le responsable de cette tâche est donc Maxime ELOPHE avec comme suppléant Ugo BOLDRON. Le jeu multijoueurs et les interactions entre joueurs devra être optimal. Le responsable de cette tâche est Yazid ADGHOGHI et le suppléant est Jean-Baptiste GUÉRIMAND.

Le game design quant à lui devrait être terminé, avec la conception des niveaux prévu à 100 % pour cette deuxième soutenance. Cinq niveaux devront donc être conçus avec tous les obstacles avant cette deuxième soutenance. Le responsable de cette tâche est donc Maxime ELOPHE avec comme suppléant Ugo BOLDRON. La conception des camps de base devra être faite dans son intégralité et l'interface d'achat et utilisation des objets devra être finie. Le scénario a déjà été complété dans son intégralité dès la première soutenance. C'est-à-dire que tous les camps de

base devront être terminés avec la boutique et les objets qui se débloquent. L'animation des évènements et le sound design devront être à eux deux terminés. C'est-à-dire que pour l'animation des évènements, dont Ugo BOLDRON est le responsable avec Maxime ELOPHE le suppléant, au moins deux des événements importants de l'ascension devront être réalisés. Pour le sound design les bruits de déplacement et la musique du menu devront être terminés. Yazid ADGHOGHI est donc le responsable et Jean-Baptiste GUERIMAND son suppléant pour cette tâche. Et pour finir les tâches à venir de cette deuxième soutenance le site Web devra être fini.

5. CONCLUSION

Le projet Ascension progresse de manière encourageante vers sa version finale malgré quelques retards. Grâce à une équipe impliquée et déterminée, chaque membre des Cœurs Brisés contribue activement à sa construction. Le développement a déjà atteint environ 40 % d'achèvement, reposant sur des bases solides, aussi bien sur le plan technique que créatif. Les efforts doivent désormais se concentrer sur l'aspect fonctionnel du jeu afin de garantir la robustesse des mécaniques du jeu. Les déplacements du joueur sont déjà partiellement en place, et la gestion des objets et les premiers niveaux de gameplay sont en bonne voie.

La prochaine étape cruciale sera la troisième soutenance technique, prévue entre le 26 et le 30 mai 2025. Cette échéance permettra d'évaluer plusieurs objectifs essentiels, notamment la parfaite fonctionnalité des déplacements du joueur, sous la responsabilité d'Ugo Boldron et Yazid Adghoghi, ainsi que la maîtrise totale du comportement des IA ennemis, dirigée par Jean-Baptiste Guerimand et Maxime Elophe. La gestion des objets devra être entièrement implémentée, un aspect pris en charge par Maxime Elophe et Ugo Boldron, tandis que le mode multijoueur nécessitera une optimisation des interactions, supervisée par Yazid Adghoghi et Jean-Baptiste Guerimand.

Parallèlement, le game design devra être achevé dans son intégralité, avec la conception de six niveaux complets, l'établissement des camps de base, la mise en place d'une interface d'achat et d'utilisation des objets, ainsi que l'animation de deux événements majeurs et la finalisation du sound design. Malgré certains défis techniques liés à la prise en main de Unity et aux exigences de programmation, l'équipe fait preuve d'une résilience remarquable et d'une grande capacité d'adaptation. La collaboration, l'organisation rigoureuse et le soutien mutuel demeurent des éléments clés pour surmonter ces obstacles.

Malgré les quelques retards, l'équipe assure une progression constante et s'engage à maintenir cette dynamique et à l'améliorer au fil des prochaines étapes du développement. Grâce à une planification minutieuse, une équipe soudée et une vision claire, les Cœurs Brisés poursuivent l'élaboration d'Ascension avec

enthousiasme. Convaincus du potentiel de leur projet, ils ambitionnent de créer une expérience ludique à la fois captivante et innovante.

Bien que le chemin soit encore long, chaque avancée rapproche un peu plus l'équipe de son objectif final. L'aventure Ascension continue, portée par la passion et la détermination de ses créateurs, avec la volonté de proposer un jeu riche en défis et en émotions.