

Rapport de projet

Super Platformer



Table des matières

1. Introduction	5
2. Présentation du projet.....	6
2.1. Origine du groupe et du projet	6
2.1.1. Origine du groupe	6
2.1.2. Origine du projet	6
2.2. Présentation des membres	7
2.2.1. CARLIER Heiko	7
2.2.2. ESCUDIER Cloé.....	7
2.2.3. PION Corentin.....	7
2.2.4. VISENTIN Aurélien	7
2.3. Etat de l'art.....	8
2.3.1. Jeu de plateforme	8
2.3.2. Super Platformer	8
3. Tâches effectuées par chaque membre	9
3.1. Entre le début du projet et la deuxième soutenance	9
3.2. Entre la deuxième soutenance et le rendu final	10
4. Les commandes	11
4.1. Lors de la première soutenance	11
4.2. Lors de la seconde soutenance	11
4.3. Etat final	12
5. Les mécaniques de jeu.....	13
5.1. Lors de la première soutenance	13
5.2. Lors de la seconde soutenance	13
5.3. Etat final	14
6. Les intelligences artificielles	15
6.1. Lors de la première soutenance	15
6.2. Lors de la seconde soutenance	15
6.3. Etat final	16
7. Le scénario	17
7.1. Dans le cahier des charges	17
7.2. Lors de la première soutenance	17
7.3. Lors de la seconde soutenance	18
7.4. Etat final	18

8. Les animations	19
8.1. Lors de la première soutenance	19
8.2. Lors de la seconde soutenance	19
8.3. Etat final	21
9. Les graphismes	22
9.1. Lors de la première soutenance	22
9.1.1. Les Objets	22
9.1.2. Les Personnages	23
9.1.3. Les fond d'écrans.....	23
9.2. Lors de la seconde soutenance	25
9.2.1. Les Objets	25
9.2.2. Les personnages	25
9.2.3. Les fond d'écrans.....	27
9.2.4. Les blocs de construction	27
9.3. Etat final	29
9.3.1. Objets	29
9.3.2. Les personnages	29
9.3.3. Les fonds d'écran.....	29
9.3.4. Le monde.....	30
10. Les bandes sonores.....	33
10.1. Lors de la première soutenance	33
10.2. Lors de la seconde soutenance	33
10.3. Etat final	34
11. L'interface	35
11.1. Lors de la première soutenance	35
11.2. Lors de la seconde soutenance	36
11.3. Etat final	37
12. Le site internet.....	39
12.1. Lors de la première soutenance	39
12.2. Lors de la seconde soutenance	40
12.3. Etat final	41
13. L'installateur	42
13.1. Lors des deux premières soutenances.....	42
13.2. Etat final	42

14. Répartition des tâches	43
14.1. Modifications lors de la première soutenance	43
14.2. Répartition des tâches lors des soutenances deux et trois	44
14.3. Déroulement des tâches	44
15. Avancement du projet	45
15.1. Lors de la première soutenance	45
15.2. Lors de la seconde soutenance	46
15.3. Avancement actuel	46
16. Déroulement du projet	47
16.1. Outils utilisés	47
16.2. Difficultés rencontrées	47
16.3. Apport du projet au groupe	48
16.3.1. Ressenti de CARLIER Heiko	48
16.3.2. Ressenti d'ESCUDIER Cloé	48
16.3.3. Ressenti de PION Corentin	48
16.3.4. Ressenti de VISENTIN Aurélien	49
17. Prochaines étapes : industrialisation et généralisation	50
17.1. Communauté d'utilisateurs	50
17.2. Industrialisation du projet	50
17.2.1. Organisation de projet Agile	50
17.2.2. Mise en place de tests automatiques	51
17.2.3. Automatisation de la construction des versions	51
17.2.4. Capitalisation des composants	51
17.3. Déploiement sur plusieurs plateformes	51
18. Conclusion	52

1. Introduction

Plus de quatre mois se sont écoulés depuis le rendu le cahier des charges pour notre jeu vidéo Super Platformer, à la mi-janvier. Quatre mois durant lesquels nous avons travaillé sur notre projet et que nous avons présenté à plusieurs reprises, lors des deux précédentes soutenances.

Nous sommes fiers de pouvoir enfin considérer notre jeu terminé. Notre travail d'équipe nous a permis de bien avancer. En effet, tout au long de la réalisation du projet, nous avons travaillé en groupe.

Les différentes classes nous permettaient de mettre de l'ordre et de regrouper nos différentes parties. Chaque script, variable, méthode était nommé rigoureusement pour que l'on comprenne ce que chacun avait souhaité faire.

A l'aide du principe de « collaboration » proposé par *Unity*, nous pouvions sauvegarder, partager et synchroniser nos différentes parties à tout moment et être ainsi toujours à jour avec les autres membres du groupe.

Vous trouverez dans ce rapport le détail de tous les aspects techniques du jeu, de leur réalisation, et la comparaison des éléments du cahier des charges avec le résultat obtenu.

2. Présentation du projet

Cette partie aura pour but de décrire le projet dans ses détails avec son origine et ses membres. Nous définirons également ce qu'est un jeu de plateforme et le but de notre jeu.

2.1. Origine du groupe et du projet

2.1.1. Origine du groupe

Avant que le groupe ne se forme, nous avions déjà sympathisé. Nous savions ainsi qu'il serait possible de travailler ensemble dans une bonne entente, dans un futur proche. La création du groupe de projet s'est donc faite rapidement et de manière naturelle.

Notre nom de groupe, ACHC, n'est autre que l'association de nos initiales en référence aux ACDC de l'école.

2.1.2. Origine du projet

Comment en sommes-nous arrivés à Super Platformer ? Tout d'abord, nous étions tous d'accord pour faire un jeu vidéo. Il était, pour nous, primordial de créer un jeu sur lequel tout le monde pourrait s'amuser facilement.

Après réflexion, nous nous sommes aperçus que nos premiers pas dans le monde vidéoludique étaient sur des jeux de plateforme avec des jeux comme *Super Mario Bros*. C'est donc avec une petite pointe de nostalgie que nous avons voulu rendre hommage à ce type de jeu, symbole de notre enfance.

2.2. Présentation des membres

2.2.1. CARLIER Heiko

Je suis Heiko Carlier, j'ai 18 ans, je viens d'une terminale S option SI et spécialité Mathématiques et je suis actuellement étudiant en première année du cycle préparatoire de l'EPITA, délégué et Chef du groupe sur le projet Super Platformer. J'ai toujours été attiré par l'informatique et le numérique en général, je trouve que c'est l'un des domaines dans lequel il y a le plus à apprendre.

2.2.2. ESCUDIER Cloé

Je m'appelle Cloé Escudier, j'ai 18 ans, et je viens d'une Terminale S spécialité ISN. Grâce à cette spécialité, j'ai eu l'occasion de programmer un jeu en groupe, mais cela était bien différent puisque ce n'était pas le même langage, et le niveau attendu était moindre.

La réalisation d'un projet collectif est pour moi très motivante, cela permet de nous entraider et de surmonter nos difficultés, en ayant un objectif concret en tête. D'ailleurs, la conception et réalisation de nombreux projets au cours de nos cinq ans à l'EPITA est une des raisons pour laquelle j'ai, personnellement, choisi cette école. En effet, cela nous permet de mieux appréhender le monde du travail et de découvrir de nombreux aspects du métier d'ingénieur en informatique.

2.2.3. PION Corentin

Bonjour, je suis Corentin Pion. Je viens d'une Terminale S-SI spécialité Mathématiques. Je suis passionné de jeu vidéo, et l'idée d'en réaliser un m'attire. Même si c'est ma première année de programmation, mon envie d'apprendre n'a pas de limites. Après tout, c'est en faisant qu'on apprend ! Le travail de groupe sous forme de projet est pour moi une opportunité d'apprendre des autres membres ainsi que de comprendre de nouvelles choses.

2.2.4. VISENTIN Aurélien

Je suis Aurélien Visentin, passionné par les nouvelles technologies depuis mon plus jeune âge, j'ai rapidement convergé vers l'informatique en suivant les traces de mon paternel. J'ai commencé la programmation par moi-même en troisième avec du Java, puis j'ai programmé presque exclusivement en python depuis la seconde, ayant choisi l'option ISN (Informatique et Sciences du Numérique). Je me suis donc naturellement orienté vers l'EPITA pour poursuivre mon parcours.

2.3. Etat de l'art

2.3.1. Jeu de plateforme

Le jeu de plateforme est un genre de jeux vidéo. Dans celui-ci, le joueur contrôle un personnage qui doit sauter sur des plateformes et éviter des obstacles.

L'univers du jeu peut être en deux ou en trois dimensions. Les environnements exigent de devoir sauter ou grimper pour pouvoir être parcourus et éviter les différents obstacles et dangers. Le joueur doit donc maîtriser le saut, celui-ci étant la mécanique de jeu la plus importante. Le gros point fort des jeux de plateforme est leur accessibilité à tout type de public : petits et grands, débutants et confirmés, tout le monde peut s'amuser sur ceux-ci.

Les jeux de plateforme existent depuis 1980, les premiers étant *Space Panic* et *Crazy Climber* développés respectivement par Universal et Nichibutsu. Cependant, le premier jeu dans lequel on pouvait sauter était *Donkey Kong*, sorti en 1981. Mais il est indéniable que la franchise phare reste celle de *Mario*, le premier fut sorti en 1983. Depuis, on ne décompte pas moins de 24 jeux de plateforme à son effigie. Cette franchise est toujours d'actualité aujourd'hui, puisqu'elle a été reprise par Nintendo.

Initialement, les jeux de plateforme étaient réputés pour leur difficulté et la rapidité qu'ils demandaient. Ils sont devenus plus simples au cours du temps tout en apportant leurs lots de nouveautés. On peut par exemple citer les capacités obtenues en collectant certains objets dans *Mario*, de nouvelles capacités apparaissant à chaque opus (boules de feu, de glace, capacité de planer ...).

2.3.2. Super Platformer

Super Platformer est un jeu regroupant de nombreux aspects des jeux de plateforme comme l'action et l'aventure. Il se destine à des joueurs de tous niveaux et se veut donc relativement accessible. Le jeu prendra place dans un univers en deux dimensions (2D), il possédera différents décors avec plusieurs types d'ennemis. L'élaboration de plusieurs stratégies sera donc requise afin de vaincre chaque adversaire.

3. Tâches effectuées par chaque membre

3.1. Entre le début du projet et la deuxième soutenance

Le tableau suivant synthétise l'ensemble des tâches effectuées par chaque membre du projet jusqu'à la deuxième soutenance.

Tâches	Sous-tâches	Aurélien	Cloé	Corentin	Heiko
Commandes	Déplacement		X		
	Saut		X		
	Accroupissement		X		
Mécaniques de jeu	Récupération d'objets	X			
	Changement de direction		X		
	Statistiques	X			
	Mort lors de la chute du joueur dans le vide		X		
	Déplacement de la caméra	X	X		
Scénario	Conception du scénario global			X	
	Retouches apportées		X		
	Premier niveau de jeu			X	X
	Dialogues		X	X	
Intelligences artificielles	IA suivant le joueur	X			
	Création d'un nouveau type d'IA	X			
	Classe IA	X			
Graphismes	Croquis personnages			X	
	Fond spatial				X
	Premier fond montagneux				X
	Second fond montagneux			X	
	Animations			X	X
	Sprites			X	X
Bandes sonores	Bruitages relatifs au jeu				X
	Choix des musiques				X
Interface	Menu principal	X	X		
	Menu des options	X	X		
Site internet	Page principale et téléchargements	X			
	Mise à jour de l'interface	X			
	Ajout de pages web	X			

3.2. Entre la deuxième soutenance et le rendu final

Ce tableau-ci répertorie les tâches effectuées par chaque membre du groupe depuis la deuxième soutenance. Nous avons construit notre plan à partir de celui-ci.

Tâches	Sous-tâches	Aurélien	Cloé	Corentin	Heiko
Commandes	Attaque		X		
Mécaniques de jeu	Classe EntityStats	X			
	Pause	X	X		
	Boîtes de dialogues		X		
Scénario	Conception des niveaux de jeu			X	
	Implémentation des niveaux de jeu			X	
Intelligences artificielles	Dégâts		X		
	I.A. lanceuse de boule de feu	X			
Graphismes	Sprites			X	
	Fond égyptien				X
	Animations		X	X	X
Bandes sonores	Bruitages				X
	Musique				X
Interface	Menu de pause	X			
	Amélioration des menus	X	X		
	Interface en jeu (HUD)	X	X		
Site internet	Modification de l'interface	X			
	Ajout des descriptions	X			
	Ajout des documents de soutenance et des médias	X			
Installateur	Création de l'installateur	X			

4. Les commandes

Les commandes représentent la base d'un jeu. Cette partie permet donc d'initialiser les touches de base, assurant le fonctionnement de Super Platformer. Les commandes peuvent être entièrement personnalisables, suivant les préférences de son utilisateur, et se doivent d'être intuitives afin de garantir une bonne expérience au joueur. Les commandes dont nous avons besoin pour notre jeu sont celles de déplacement, vers la gauche et vers la droite, le saut, l'accroupissement et le fait de blesser un ennemi avec l'épée.

4.1. Lors de la première soutenance

Les premières commandes que nous avons voulu implémenter étaient celles du déplacement et du saut. Celles-ci sont essentielles pour notre jeu mais aussi lors des tests de nos différents scripts. Il nous a donc fallu comprendre comment programmer ces actions sur *Unity* mais aussi nous familiariser avec le logiciel.

Pour le déplacement, il suffit simplement de regarder les entrées des flèches directionnelles de gauche et de droite et de regarder leurs valeurs (si le joueur appuie sur celle de gauche, la valeur vaut -1, s'il appuie sur celle de droite, la valeur vaut 1). Ainsi, nous savons lorsque le joueur veut se déplacer. Il suffit alors d'associer au personnage une vitesse de déplacement selon l'axe horizontal.

Le saut nous a posé un peu plus de problèmes. Dans un premier temps, il nous a fallu chercher la méthode d'*Unity* pour donner une force d'impulsion au personnage. Une fois trouvée, un problème majeur s'est imposé : la vérification que notre personnage est bien sur le sol pour qu'il puisse sauter. En effet, le joueur pouvait jusqu'alors sauter en permanence, même s'il était dans les airs. Or, ce n'est pas ce que nous voulions. Il a donc fallu créer un cercle, qui n'apparaît pas à l'écran, qui se trouve aux pieds de notre héros et dès qu'un point de ce cercle touche un objet avec un *Layer Ground*, soit les objets de type sol, une variable booléenne passe à vrai. Ainsi, l'action de sauter s'effectue seulement quand cette variable est vraie et que le joueur appuie sur la barre d'espace, ce qui résout notre problème.

4.2. Lors de la seconde soutenance

Lors de la seconde soutenance, la commande d'accroupissement avait été implémentée. Celle-ci, bien que fonctionnelle ne possédait pas encore d'animation et n'était donc pas encore visible par l'utilisateur. Afin d'implémenter l'accroupissement, nous utilisons des boîtes de collision, appelées « *Box Collider* » sur *Unity*. Nous en avons une qui faisait la taille du personnage lorsque celui-ci était debout et une qui faisait la taille du personnage lorsque celui-ci était accroupi. Lors de l'accroupissement, on activait la petite boîte et on désactivait la grande, et lorsque le personnage se relèvait, on activait la grande et on désactivait la petite. Pour activer cet accroupissement, il suffisait de détecter l'appui sur la touche de la flèche du bas à l'aide de la classe *Input* du logiciel, comme pour les autres commandes.

De plus, le personnage n'était pas censé se relever s'il avait un objet au-dessus de lui. Pour détecter cela, nous utilisâmes la même méthode que pour détecter si le héros touchait le sol au moment du saut : nous créâmes un cercle invisible au niveau de la tête du personnage, mais avec une marge plus grande, et si un point de ce cercle touchait un objet de type sol, alors une variable booléenne passait à vrai. Ainsi, si le joueur relâchait la flèche du bas mais que la variable était vraie, le héros restait accroupi jusqu'à que la variable passait à faux, soit jusqu'à qu'il n'ait plus d'objet au-dessus de lui et qu'il puisse donc se relever.

4.3. Etat final

Les commandes de l'accroupissement, du déplacement et du saut comportent désormais une animation chacune, celles-ci étant les derniers éléments manquants à leur réalisation.

La commande d'attaque est maintenant implémentée, celle-ci permettant d'infliger des dégâts voire de tuer les monstres se trouvant devant le joueur. Elle comporte également une animation. Pour enlever des dégâts aux ennemis, il nous fallait tout d'abord les détecter. Pour cela, lorsque le joueur appuie sur la touche d'attaque, un cercle invisible se dessine autour de l'épée et si un point de ce cercle entre en collision avec un objet de type « *Enemy*, » alors cet objet doit perdre des points de vie. Pour cela, on recherche sur cet objet en question le script « *EntityStats* » qui sera détaillé plus tard, et on inflige alors des dégâts avec la fonction « *TakeDamage* » de ce script. Ainsi, le monstre perd réellement de la vie.

De plus, nous ne voulions pas que notre joueur puisse simplement rester appuyé sur la touche d'attaque et attaquer infiniment, ce qui rendrait le jeu trop facile. Nous avons donc un compteur de temps entre les attaques et si le joueur appuie sur la touche avant que ce compteur n'ait atteint zéro, il ne se passe rien.

5. Les mécaniques de jeu

Les mécaniques de jeu représentent les différentes actions que le joueur peut effectuer mais également les contraintes imposées à ce dernier. Suivant les types de jeu, les mécaniques peuvent varier. Pour un jeu de plateforme, le saut, tout comme les déplacements sont les capacités les plus utilisées ainsi que les plus importantes. Les mécaniques de jeu comprennent également la récupération d'objets et la gestion de la vie.

5.1. Lors de la première soutenance

Dans un premier temps, le joueur était capable de récupérer des objets, des pièces par exemple. Un compteur était alors mis à jour. Afin de réaliser cette action, un script avait été créé. Il prenait en paramètre un joueur, et incrémentait le score lorsque l'objet sur lequel le script était appliqué entraînait en collision avec le joueur. L'objet était ensuite détruit. Mais bien que ce compteur fût présent, il n'était pas encore affiché sur l'interface du jeu.

Comme précisé dans la partie commandes, le joueur était capable de se déplacer dans le jeu.

Enfin, le personnage pouvait également se tourner dans la direction dans laquelle il se dirigeait. Cela permettait d'augmenter le réalisme du jeu.

5.2. Lors de la seconde soutenance

En préparation du deuxième compte rendu, nous avons implémenté une classe « Stats ». Celle-ci regroupait en un unique fichier l'ensemble des informations relatives à tout type de personnages, alliés ou ennemis. Parmi ces informations se trouvaient la vitesse de déplacement, la vie ou les dégâts de l'attaque.

Une autre classe regroupait l'ensemble des informations relatives au jeu, que ce soit le nombre de pièces récoltées ou le nombre de morts possibles avant de perdre. Bien que présente dans le jeu, elle n'était pas encore utilisée, le joueur n'était pas encore capable de mourir.

De plus, nous avons également modifié le fonctionnement de la caméra afin que celle-ci suive le joueur lors de ses déplacements. Cela permettait ainsi une meilleure fluidité et la possibilité de créer de plus grands niveaux. Une image de fond était également affichée sur la caméra, quelle que soit la position du joueur ou de la caméra.

5.3. Etat final

La classe « *Stats* » est renommée « *EntityStats* » afin de ne pas porter à confusion, en les séparant des statistiques globales du jeu. De plus, elle prend maintenant en compte la mort de l'entité. En effet, si la vie passe à zéro ou s'il y a une chute dans le vide, l'entité est détruite et, dans le cas du joueur, affiche un écran de « *Game Over* » puis relance le niveau. La chute dans le vide est également prise en compte, si le joueur passe en dessous d'une limite choisie dans l'axe vertical, il est tué comme décrit précédemment.

L'attaque des I.A. a maintenant été implémentée, nous le verrons par la suite dans la partie intelligence artificielle. Par exemple, ces I.A. sont capables d'être tuées par le joueur et peuvent aussi lui infliger des dégâts.

De même, le joueur peut infliger des dégâts aux ennemis et en recevoir au point de mourir.

Concernant l'antagoniste, celui-ci est maintenant capable d'envoyer des boules de feu.

De plus, nous avons implémenté la pause. Celle-ci, lorsqu'activée, arrête. Lorsqu'elle est activée, elle arrête l'ensemble des calculs physiques tout en laissant fonctionner les calculs graphiques. On peut donc toujours interagir avec l'interface du jeu mais les joueurs, les intelligences artificielles et les plateformes mouvantes sont gelés, aucun déplacement n'est possible.

Enfin, les boîtes de dialogues sont maintenant implémentées. Lors du passage du joueur dans une certaine zone, le jeu se met en pause et un rectangle noir avec du texte apparaît sur l'écran. Le joueur doit alors appuyer sur la touche « Entrée » du clavier pour passer sur la boîte suivante ou, s'il n'y en a plus, fermer la boîte de dialogue pour continuer le jeu.

6. Les intelligences artificielles

L'intelligence artificielle ou I.A. est un cadre important. Dans notre cas, elle se manifeste dans le comportement des différentes créatures, complexifiant la quête du joueur. Un jeu sans ennemis est monotone, ce qui n'encourage pas les utilisateurs à continuer. De plus, sans adversaires à vaincre, le jeu serait bien trop simple.

6.1. Lors de la première soutenance

Lors de la première soutenance, nous avons présenté un type d'intelligence artificielle suivant le joueur jusqu'à une distance donnée et ayant une vitesse de déplacement totalement paramétrable. Celle-ci était alors primitive mais avait pour but de se donner une base pour la création d'I.A.

Une classe, appelée « *GlobalAI* » avait également été créée afin de simplifier la création de futures intelligences mais aussi pour uniformiser l'utilisation de celles-ci dans le code. Cette classe avait par ailleurs été présentée dans les détails en décrivant son fonctionnement.

6.2. Lors de la seconde soutenance

Entre les deux premières soutenances, nous avons choisi de remodeler l'entièreté du code permettant de faire fonctionner les intelligences artificielles. Cela avait pour but de corriger de nombreuses erreurs présentes dans le code, d'augmenter la lisibilité et la clarté de celui-ci tout en facilitant l'implémentation de futures I.A. Par exemple, lors de la première soutenance, la classe « *GlobalAI* » comportait également le code de l'I.A. qui suivait le joueur. Les scripts furent donc séparés afin d'augmenter la clarté du code global. On avait donc une classe pour chaque type d'intelligence artificielle.

Dans les mécaniques de jeu, nous avons présenté une classe « *Stats* ». Comme son nom l'indique, elle retrouvait la totalité des caractéristiques relatives à un personnage. On peut, par exemple, citer la vitesse de déplacement, la vie, etc. Les I.A. furent alors modifiées pour prendre en compte cette classe. Chaque intelligence avait ainsi des caractéristiques propres comme la vie ou les dégâts d'attaque.

De plus, une seconde intelligence fut implémentée, celle-ci faisant « les cent pas ». Cette dernière se déplaçait de manière continue entre deux positions prédéfinies. Ce script pouvait également être utilisé pour effectuer des plateformes mouvantes sur lesquelles le joueur pouvait se déplacer.

Enfin, la vie des I.A. ayant été implémentée, nous avons alors créé une fonction appelée « *TakeDamage* » permettant de lui infliger des dégâts. Cette fonction n'était alors pas encore utilisée dans le jeu puisque l'attaque du joueur n'avait pas encore été implémentée.

6.3. Etat final

Le groupe étant en avance sur les I.A. lors de la deuxième soutenance, celles-ci n'ont reçu que peu d'améliorations. En effet, nous avons préféré nous concentrer sur le reste du projet puisque les intelligences artificielles déjà effectuées sont largement suffisantes pour le projet.

Les intelligences artificielles utilisent maintenant la totalité de la classe « *Stats* », renommée « *EntityStats* », celle-ci s'appliquant maintenant aussi bien sur les I.A. que sur le joueur. Par exemple la vitesse de déplacement de la classe est maintenant utilisée au lieu de la paramétrer directement sur le script faisant fonctionner l'intelligence artificielle.

Elles sont maintenant capables d'infliger des dégâts aux joueurs mais sont également capables d'en recevoir lorsqu'elles reçoivent un coup d'épée. La quantité de dégâts et de vie est facilement paramétrable au sein du logiciel. Ces améliorations étaient les dernières fonctionnalités requises pour procéder au bon déroulement du jeu.

Une nouvelle intelligence artificielle a été créée, capable de lancer des boules de feu. Elle sera utilisée pour notre antagoniste.

Enfin, lors de la seconde soutenance, la possibilité de création d'une I.A. hybride était mentionnée, elle devait osciller entre deux positions et se mettre à suivre le joueur lorsque celui-ci était dans son champ de vision. Nous avons choisi de ne pas l'implémenter afin de nous concentrer sur le reste du jeu.

7. Le scénario

7.1. Dans le cahier des charges

Dans le cahier des charges, nous avons créé une trame globale pour le jeu. Nous avons également précisé que l'histoire ne serait pas définitive et qu'elle serait modifiée tout au long de la réalisation du projet.

Super Platformer raconte l'histoire d'un héros dont le but est de trouver le pouvoir de la Triforce, qui réside dans une pierre sacrée. Ce pouvoir légendaire lui permettra de combattre Hervé le Destructeur qui menace d'anéantir l'humanité. Dans sa quête, il devra affronter plusieurs ennemis pour récupérer les fragments de portail, éparpillés dans différents mondes et époques.

« Êtes-vous prêts pour l'aventure qui vous attend ? Faites les bons choix, triomphez des ennemis, et traversez les mondes et les époques. Arriverez-vous à sauver l'humanité ? Venez le découvrir. »

7.2. Lors de la première soutenance

En préparation cette soutenance, plusieurs idées nous étaient venues à l'esprit. Nous nous étions alors concentrés sur une version du scénario qui satisfaisait l'ensemble du groupe. En voici l'introduction :

« Il y a plus de cent ans, les forces du mal se déchaînaient sur la Terre. A leur tête se trouvait Hervé le Destructeur, un jeune garçon aux pouvoirs maléfiques, dont la force et la dangerosité sont sans égales. Sa sorcellerie avait entraîné la crainte de tous les habitants de la planète. Il n'y avait plus aucun espoir... Arriva alors un héros, au cœur pur et innocent, qui tenta de vaincre Hervé à l'aide de la Triforce, un artefact aux propriétés magiques, capable de décupler la force de son utilisateur. Le combat entre ces deux êtres fut épique, malheureusement Hervé triompha. Dans un dernier souffle, le héros scella les fragments de la Triforce dans différentes époques, pour qu'elle ne tombe pas aux mains des forces du mal. Vous, descendant du héros vaincu, êtes le seul capable de rassembler ces fragments pour pouvoir enfin débarrasser la Terre de ce fléau. Mais attention, au cours de votre quête, Hervé le Destructeur et ses sbires feront tout leur possible pour vous faire échouer. Pour atteindre votre objectif, vous voyagerez à travers les époques en utilisant des portails temporels pour récupérer les fragments de la Triforce. Ainsi, vous pourrez, à votre tour, combattre Hervé le Destructeur et libérer le monde de ce danger. »

Pour cette première présentation, nous avons donc créé la trame principale du jeu avec une meilleure idée de son déroulement. Trois époques, passé, présent et futur seraient présentées au travers de trois niveaux différents.

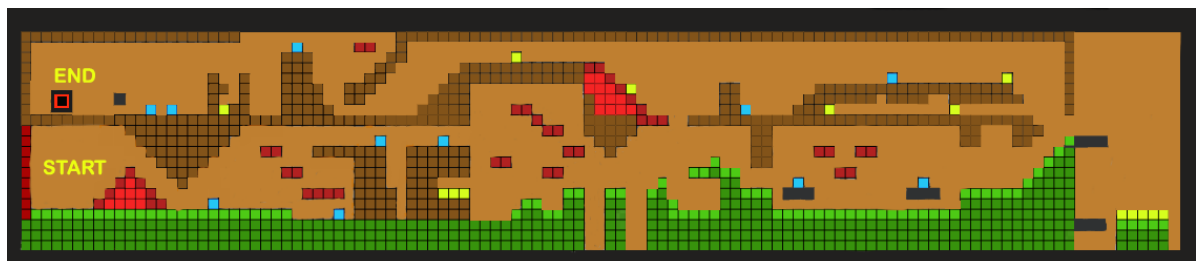
7.3. Lors de la seconde soutenance

Pour la deuxième soutenance, nous nous étions fixés comme objectif de créer les différents dialogues ou monologues des personnages. Les interactions entre Hervé le destructeur et notre protagoniste et les dialogues d'introduction avaient donc été notre priorité.

Nous avons également rédigé des phrases qui serviraient en tant que didacticiel, celles-ci expliqueraient au joueur les différentes touches du jeu ainsi que les actions possibles. Par exemple, « Appuyez sur les flèches directionnelles pour vous déplacer ».

Les dialogues créés ne faisaient pas encore partie du jeu puisque nous n'avions pas implémenté l'interface qui permettaient leur affichage.

En outre, le premier niveau du jeu fut imaginé, il ne restait plus que son implémentation. En voici une version schématisée.



Rouge :
Brique

Marron :
Pierre

Vert :
Terre et herbe

Bleu :
Monstres

Jaune :
Pièces

7.4. Etat final

Pour la dernière soutenance nous avons implanté les différentes scènes de dialogues dans les niveaux. Les dialogues permettent aussi au joueur d'apprendre les différentes fonctionnalités de notre jeu. Nous avons aussi réalisé les trois niveaux de notre jeu ainsi que la salle du boss final. Les 3 niveaux offrent chacun une expérience différente. Par exemple, le deuxième niveau est basé plus sur le saut ainsi que les plateformes alors que le troisième est basé sur le combat de monstres.

8. Les animations

Les animations permettent de dynamiser le jeu, il s'agit de l'affichage de mouvements qui dépendent souvent des actions du joueur, des I.A. et des mécaniques de jeu. La création d'une animation s'effectue en mettant bout à bout des images dont une partie bouge plus les images avancent.

L'ensemble des images utilisées pour les animations ont été faites à la main, l'une après l'autre à l'aide du logiciel « *Gimp* ». Cela produit de très bons résultats mais consomme une quantité de temps non négligeable.

8.1. Lors de la première soutenance

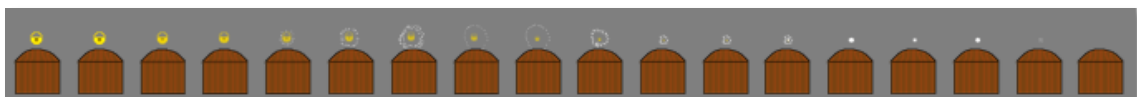
Lors de la première soutenance nous avons uniquement effectué un petit nombre d'animations pour nous familiariser avec le logiciel. Nous avons alors animé un coffre dont le cadenas se déplace de haut en bas.

Voici les différentes images d'un coffre lorsqu'il est verrouillé et en attente d'être ouvert. Bien que cela ne soit pas évident lorsque les images sont côte à côte, la distance entre le cadenas et le coffre évolue au cours du temps.



| Animation du coffre (mouvement du cadenas)

Le déverrouillage de ce coffre avait également été animé.

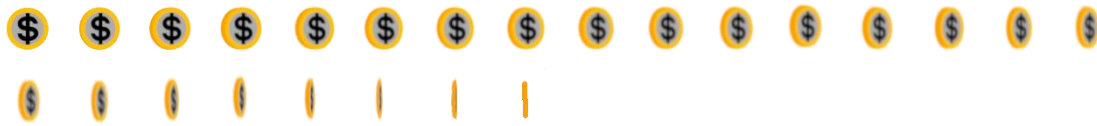


| Déverrouillage du coffre

8.2. Lors de la seconde soutenance

Pour la deuxième soutenance, nous nous concentrâmes sur l'animation du personnage principal. Nous créons ainsi différentes animations tel que le saut, l'attaque ou encore le déplacement. Pour les différentes actions du joueur, nous dûmes réaliser plusieurs prototypes afin de choisir l'ensemble d'images ayant le meilleur résultat. Par exemple, le premier ensemble d'images concernant l'attaque n'était pas très différent du mouvement final, donc nous optâmes pour un ensemble différent.

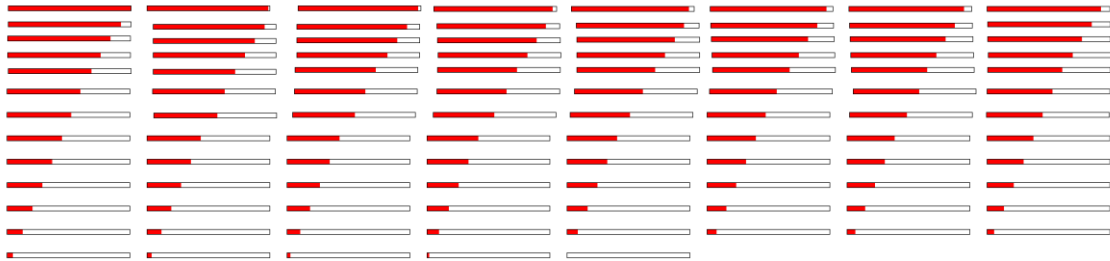
Nous avançâmes également sur les autres personnages, avec le déplacement du monstre ou encore l'attaque du monstre. Pour les objets, nous animâmes les pièces et autres objets récupérables.



Les images composant l'animation « aller » d'une pièce

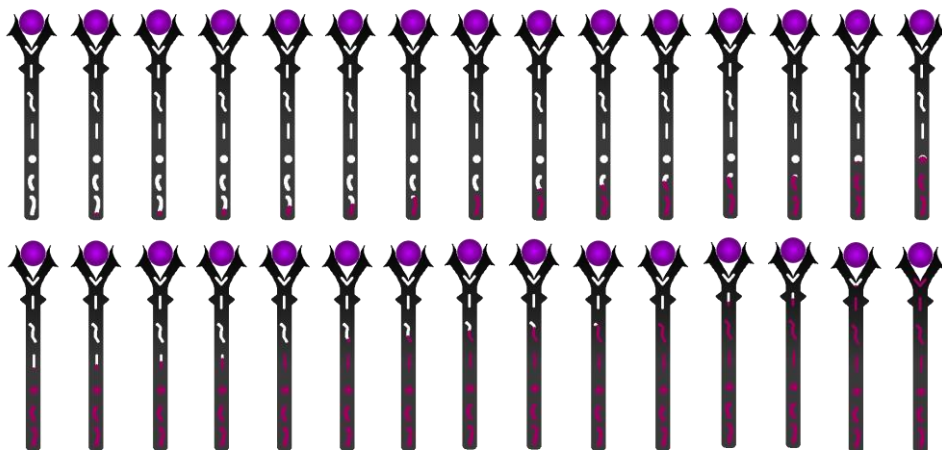
Nous avons également créé une animation pour la barre de vie. Celle-ci ne sera malheureusement pas utilisée dans le projet, une méthode plus simple ayant été trouvée et mise en place.

Nous pouvons néanmoins décrire comment celles-ci auraient été utilisées. Tout d'abord, les images ci-dessous devaient être dynamiquement jouées lorsque le joueur se faisait toucher par un ennemi. Cela devait résulter en une animation fluide et agréable pour l'utilisateur final. Par exemple, si le joueur possédait 100 points de vie et prenait 20 points de dégât, chacune des images allant de 100 à 80 seront jouées à la suite. L'inverse aurait été effectué si le joueur regagnait de la vie.



Les 101 images composant les animations de perte de vie

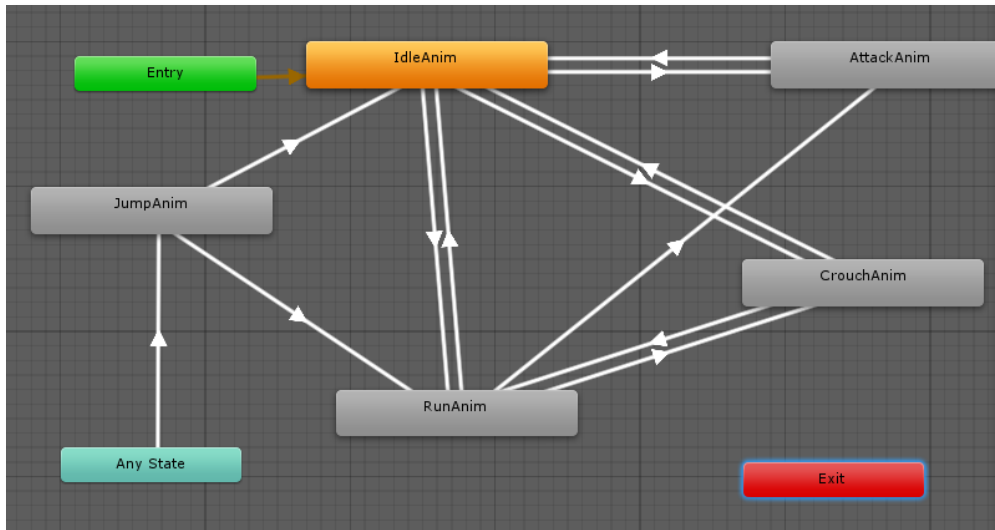
La dernière animation effectuée était celle du sceptre, celui-ci représentait un temps d'attente avant la prochaine attaque d'Hervé le destructeur. En effet, lorsque le sceptre était entièrement rempli, l'attaque de notre antagoniste aurait été lancée.



Images servant à l'animation du sceptre

8.3. Etat final

Pour cette dernière soutenance, il nous a fallu terminer de dessiner les animations qu'il nous manquait et il nous a surtout fallu les implémenter à l'aide d'*Unity*. Pour cela, il fallait tout d'abord créer une animation grâce à la partie *Animation*. Il fallait mettre toutes les images d'un « *Sprite* » à la suite pour obtenir un mouvement. Une fois que les différentes animations étaient créées, nous devons alors utiliser la fonctionnalité *Animator* d'*Unity*. Elle nous ont permis de créer un schéma fonctionnel pour déclencher l'animation au moment voulu.



La fonctionnalité Animator d'Unity

Chaque flèche blanche est une transition, et sur chaque transition, il y a des conditions. Par exemple pour passer de l'animation de saut à l'animation de course, il faut vérifier que le héros soit retombé sur le sol et que le joueur ait appuyé sur la touche pour se déplacer.

9. Les graphismes

Les graphismes sont une partie requise à la réalisation d'un jeu. En effet, ils permettent une dynamisation de l'environnement tout en rendant l'interaction avec celui-ci plus plaisante. Nous nous étions fixés comme objectif de réaliser la totalité de notre jeu, c'est-à-dire sans utiliser d'éléments que nous n'avons pas créés. Ainsi, en termes de graphismes, nous avons dû dessiner tous les décors, personnages, objets, etc.

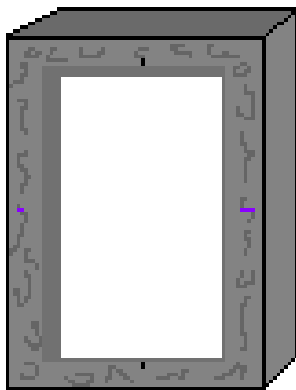
Afin de réaliser ceux-ci, nous avons choisi d'utiliser un logiciel gratuit et performant, *Gimp*. *Photoshop* a également été utilisé mais, ce dernier présente l'inconvénient d'être payant.

Afin de simplifier la lecture, nous allons diviser chacune de leur présentation en différentes sous-parties.

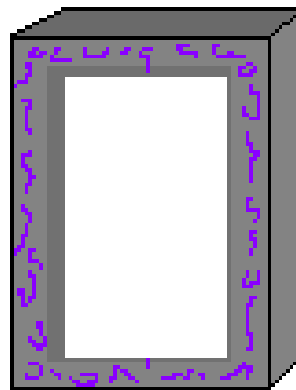
9.1. Lors de la première soutenance

9.1.1. Les Objets

Nous dessinâmes quelques objets, mais ces dessins servaient principalement à l'animation, il fallut donc dessiner entre une dizaine et une centaine de dessins différents pour le même objet afin de permettre une animation. Pour la première soutenance nous réalisâmes les dessins destinés à l'animation du coffre et du portail.



Première image de l'animation du portail



Dernière image de l'animation du portail

9.1.2. Les Personnages

Pour la première soutenance, notre objectif était de s'imaginer les différentes apparences de personnages, leurs nombres, leurs attributs, etc. Nous commençâmes par dessiner des croquis de nos personnages. En voici un exemple :

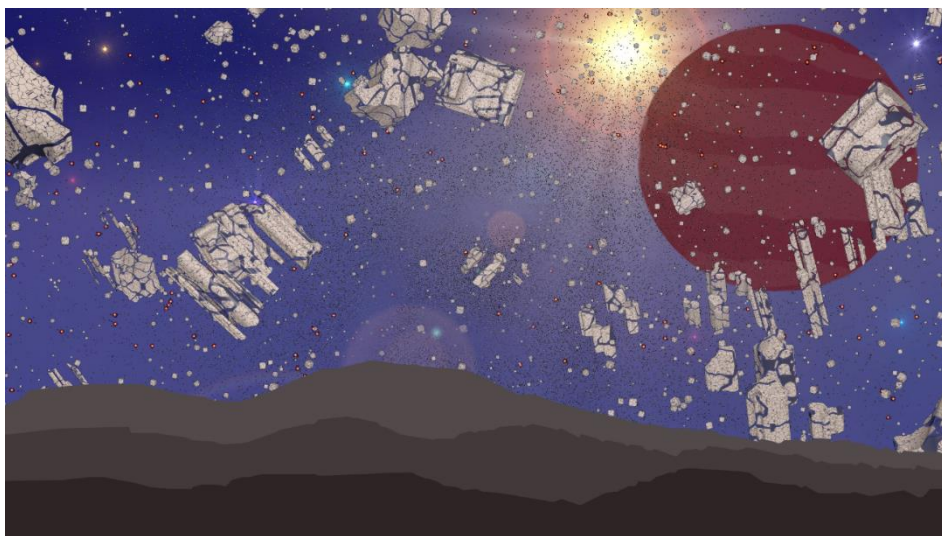


| Numérisation de l'antagoniste

Les croquis nous permettaient une certaine maniabilité pour les attributs des personnages. Ils furent notre base pour la numérisation informatique. Il faut savoir que ces dessins comportaient beaucoup de caractéristiques qui furent supprimées lors de la reconstruction sur ordinateur (le niveau de détails étant trop important).

9.1.3. Les fond d'écrans

Nous créâmes aussi deux fonds d'écran à afficher lors des menus et des niveaux pour rajouter à l'ambiance du jeu : un fond d'écran pour le futur, dans l'espace, et un fond d'écran dans le présent avec un paysage vallonné bordé de montagnes.

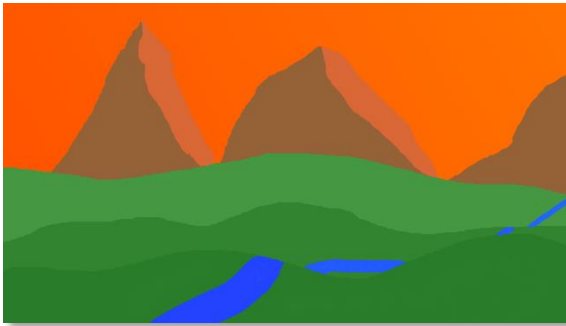


| Paysage futuriste

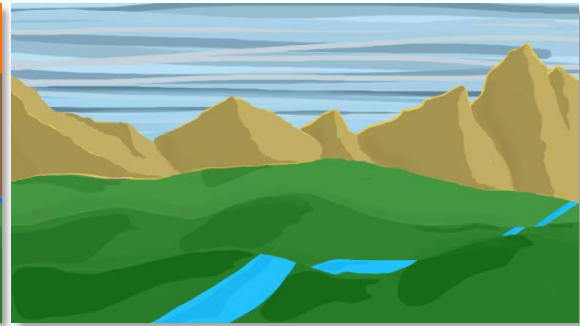
Pour le futur, nous fûmes partis sur une base grise/noire pour le sol. Cela nous rappelait une atmosphère lunaire. Nous avons aussi un côté post-apocalyptique avec tous les débris de stations spatiales qui furent détruites par Hervé le Destructeur. Nous rajoutâmes la planète pour donner un effet de distance et de lumière avec du jaune (soleil) en fond à du noir (premier plan).

Pour le paysage représentant l'époque du présent nous voulions créer un fort contraste avec celui du futur qui devait posséder une ambiance sombre. Nous avons alors décidé de créer un paysage verdoyant montrant différents éléments de la nature.

On peut remarquer que la première ébauche comportait des couleurs mates, sans aucun détail. Nous avons remédié à ce problème avec un dégradé de couleurs plus ou moins fort afin donner une impression de profondeur. Le fond a également été changé pour donner un aspect plus réaliste tout en gardant quelques nuages sombres pour signaler un danger proche.



| *Première ébauche du paysage*



| *Dessin final du paysage*

9.2. Lors de la seconde soutenance

9.2.1. Les Objets

Pour la seconde soutenance nous incorporâmes de nombreux objets comme les pièces, les fragments de portail ou encore une boule de feu pour l'attaque du boss.



| Pièce



| Fragment de portail



| Boule de feu

Certaines de ces images furent prévues pour l'animation comme la pièce, tandis que d'autres ne seraient pas animées, par exemple la boule de feu (pour des raisons de complexité).

Chaque objet a une importance dans le jeu, ce ne sont pas des éléments de décor. Par exemple, les fragments de portail servent à activer le portail et ainsi pouvoir finir le niveau.

9.2.2. Les personnages

Concernant les différents changements apportés aux personnages par rapport à la première soutenance, nous modifiâmes le nombre de personnages présents dans notre jeu. Nous ne pouvions pas réaliser différents types d'ennemi sans réduire la qualité de chaque dessin qui entraînerait une certaine perte sur le plan visuel. Nous passâmes donc de onze personnages à quatre personnages.

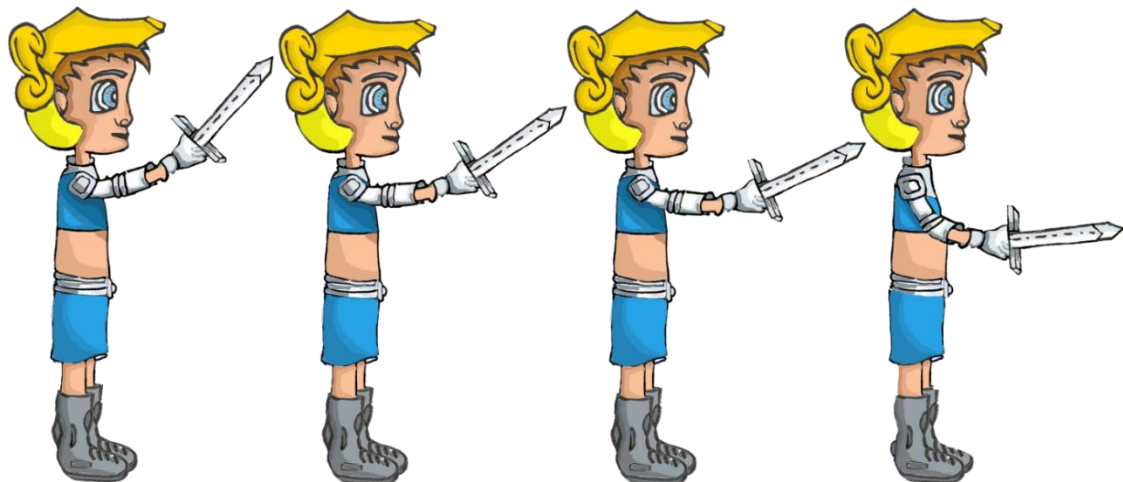
Nous modifiâmes également l'apparence de l'antagoniste principal pour répondre à différents critères émis dans le scénario : « un jeune garçon aux pouvoirs maléfiques, dont la force et la dangerosité sont sans égales. Sa sorcellerie avait entraîné la crainte de tous les habitants de la planète ».

Pour ne pas perdre les premières ébauches de l'antagoniste, nous les réutilisâmes afin de façonner notre héros. Dans l'exemple ci-dessous, nous remarquons bien que l'apparence de l'antagoniste a changé et que le croquis a servi à représenter le joueur.

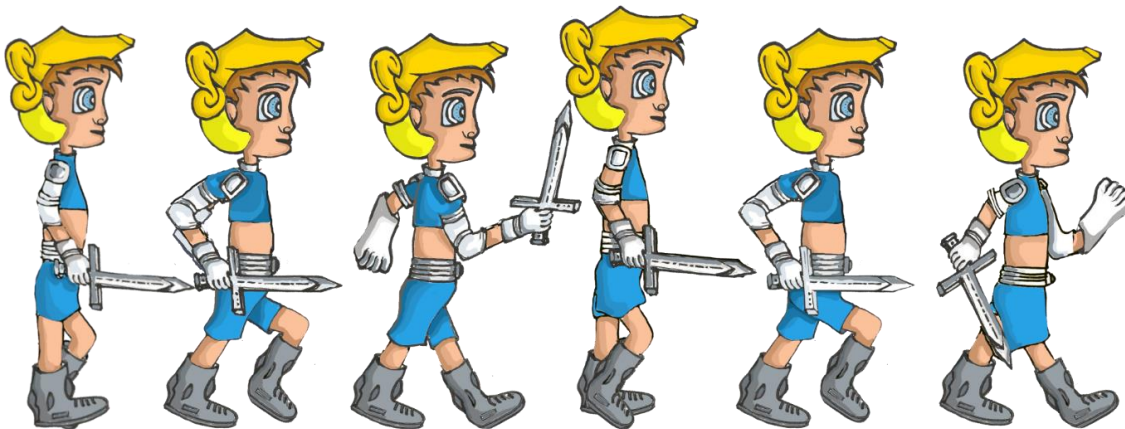


| *Changement de l'apparence de l'antagoniste*

Nous avons tout de même réalisé différents dessins notamment le déplacement ainsi que l'attaque du héros. Voici les 4 images nécessaires à l'attaque du joueur.



Ci-dessous, les 6 dessins servent à représenter le déplacement horizontal du personnage. Ceux-ci, comme précisé dans la partie animations, sont affichés les uns à la suite des autres, ainsi, pour le joueur, une impression de mouvement se crée.



| Déplacement du joueur principal

9.2.3. Les fond d'écrans

Lors de la seconde soutenance, un fond de perte du jeu fut créé. Celui-ci s'affichait lorsque le joueur n'avait plus de vie. Ce fond est souvent appelé écran de « *Game Over* » dans le domaine vidéoludique.



| Ecran de « *Game Over* »

9.2.4. Les blocs de construction

Les blocs de construction sont des éléments indispensables pour la création de nos niveaux. Ceux-ci s'assemblent entre eux et forment alors des plateformes sur lesquelles le joueur peut se déplacer.

Lors de la seconde soutenance, nous créâmes quatre blocs différents. Ces blocs étaient des blocs de terre, d'herbe, de pierre et de brique. Ci-dessous se trouvent les différentes textures :



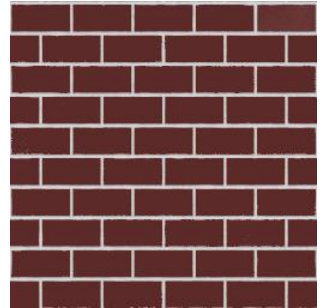
| *Bloc de terre*



| *Bloc d'herbe*



| *Bloc de pierre*



| *Bloc de brique*

9.3. Etat final

9.3.1. Objets

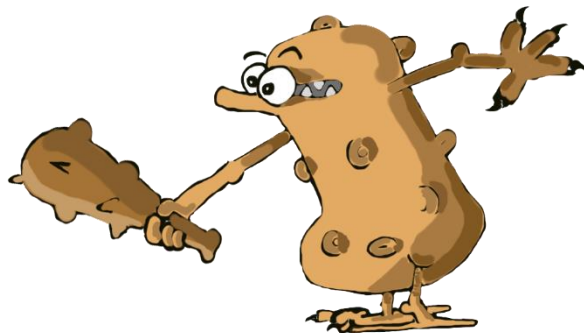
Pour la troisième soutenance nous finîmes de dessiner les différents objets notamment les potions qui permettaient de redonner de la vie au joueur.



Potion de vie

9.3.2. Les personnages

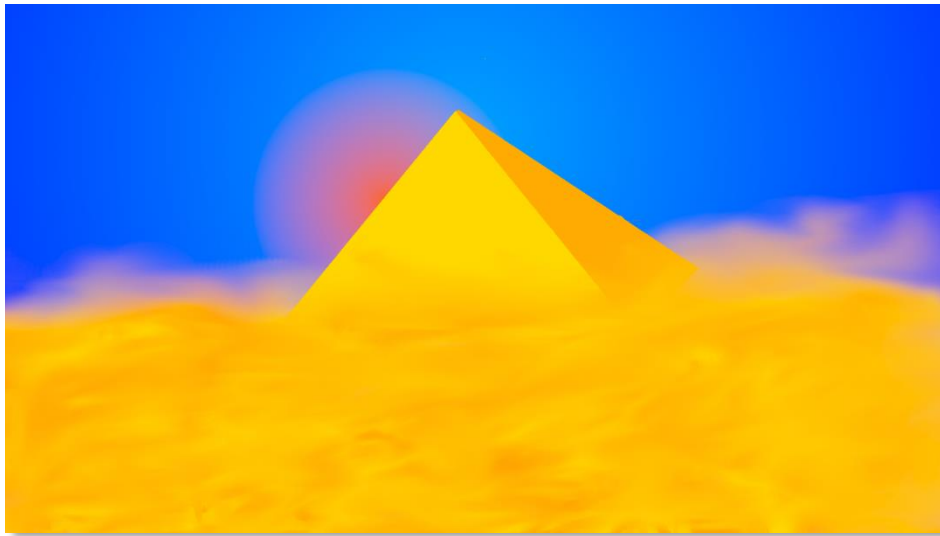
Concernant les personnages, nous finîmes les différentes images des personnages nécessaires pour l'animation. Nous dessinâmes également les monstres de base. Pour leurs apparences, nous nous inspirâmes de la bande dessinée « Game OVER » recouvrant différents monstres.



Monstre inspiré de « Game OVER »

9.3.3. Les fonds d'écran

Lors de la seconde soutenance, le seul fond d'écran manquant était celui du passé. Nous le dessinâmes alors en choisissant de représenter une pyramide égyptienne. En effet, celle-ci était facilement associable à l'Egypte antique, et donc au passé, par l'utilisateur du jeu, même s'il possédait peu de connaissances historiques.



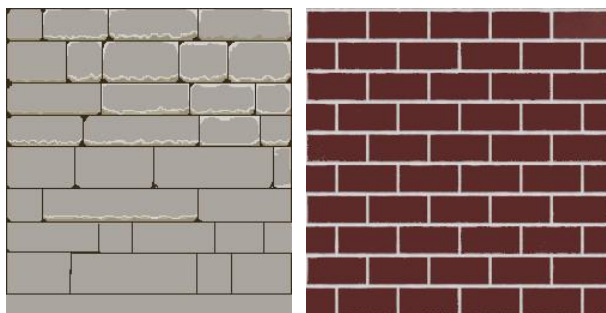
| *Fond du troisième niveau, représentant le passé*

Pour la création du troisième niveau nous choisîmes donc un thème sur l’Egypte. Nous dûmes dessiner de nouveaux blocs pour mieux rentrer dans cet univers. Nous partîmes donc sur une base jaunâtre rappelant des constructions en grès suivant la même apparence que le bloc de briques. Enfin, pour le dernier rempart, nous fîmes un mélange des différents mondes. Le combat final se composait en grande majorité sur de l’esquive de projectiles comme les boules de feu.

9.3.4. Le monde

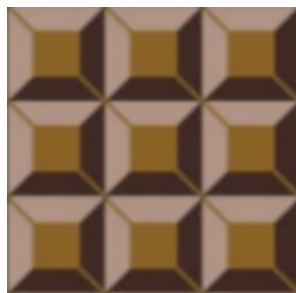
Pour la gestion des différents niveaux nous dûmes créer différents blocs qui représenteraient le monde. Pour notre jeu, nous partîmes sur un ensemble de blocs, comme des blocs de pierres, de briques, de terre et divers autres blocs.

Par rapport à la deuxième soutenance, nous modifiâmes certains blocs dans un souci d’apparence. En effet, dû à leur petite taille, les effets sur les blocs donnaient un aperçu qui ne nous convenait pas. Ci-dessous, on peut voir les différents changements apportés à notre univers. Les premières images correspondent aux blocs de la deuxième soutenance.



| *Bloc de pierre*

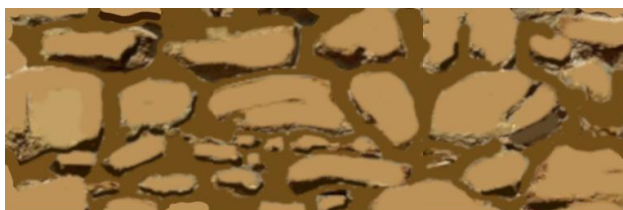
| *Bloc de brique*



| Bloc de brique

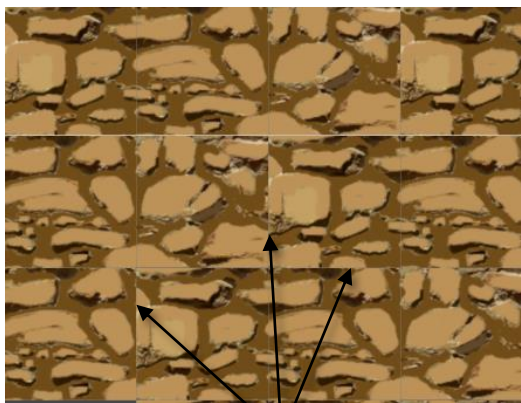


| Bloc de brique n°2



| Ensemble pierre

Un défi à réaliser, était de détruire une forme de redondance. En effet, lorsqu'on remplit un grand espace avec une sorte de bloc, on peut créer plusieurs anomalies. Un exemple d'anomalie qui était présent sur les blocs de pierre est montré ci-dessous.



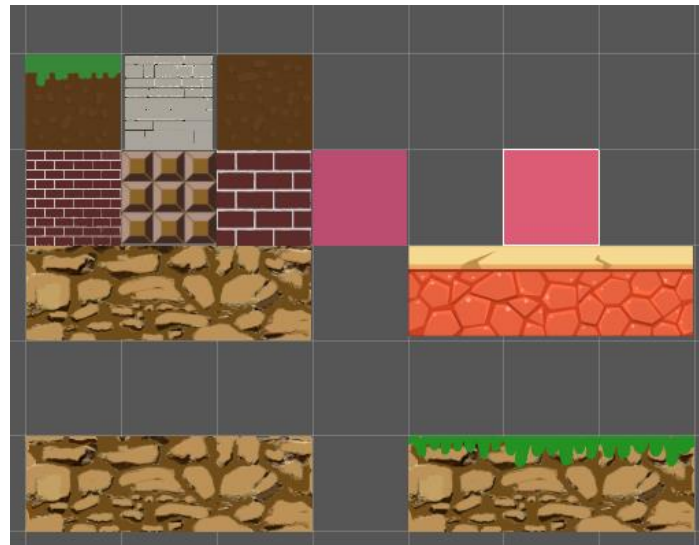
| Anomalie entre les blocs de pierre



| Blocs de pierre corrigés

A noter que sur l'image de droite, on peut remarquer quelques défauts. Il ne faut pas oublier que les images sont très agrandies et que par conséquent certains « défauts » ne sont en réalité pas visibles sur le jeu.

A l'aide d'Unity nous pouvons créer une « *Tile Palette* ». Celle-ci regroupe l'ensemble des textures utiles pour la création d'un niveau sous forme de cubes qu'il suffit d'assembler dans une « *Tilemap* ».



| La « *Tile Palette* » du projet

Les « *Tilemaps* » sont une fonctionnalité d'*Unity* qui nous permet de placer des blocs, créés à partir de différentes images, dans une grille 2D. Les blocs ont donc un aspect similaire à ceux du jeu « *Super Mario Bros* ». Comme précisé précédemment, ce jeu est une des sources d'inspiration principale du projet.

La fonctionnalité « *Tilemap* » est très pratique car l'ensemble des blocs restent toujours alignés entre eux et n'ont pas de risque de décalage.

10. Les bandes sonores

Les bandes son représentent l'âme d'un jeu. Les différentes musiques ou effets sonores nous permettent de créer une ambiance spécifique permettant une immersion du joueur dans celui-ci. Un son blanc ou une bande sonore répétitive risque également d'ennuyer, voire d'agacer le joueur. Il en résulte une expérience de jeu dégradée.

Le format papier du document ne nous permet malheureusement pas de donner un aperçu des sons. Ceux-ci sont cependant disponibles sur le site web du projet à l'adresse internet « anvstin.github.io/SuperPlatformer » dans la section médias.

10.1. Lors de la première soutenance

Dans notre cahier des charges, les sons ne devaient avoir aucun avancement. Nous avons néanmoins pris un peu d'avance en enregistrant deux sons afin de se familiariser avec *Audacity*, notre logiciel de traitement audio.

Les sons jouent un rôle important dans le projet car ils permettent de sublimer le jeu mais, comme les sons sont calqués sur les actions et non l'inverse, nous devions attendre d'avoir une idée de la plupart des états physiques et des mouvements du personnage. Nous avons donc commencé à imaginer quels sons pourraient être utilisés et allions réaliser tous les sons nous-même en nous servant d'objets, de bruits de la vie quotidienne ainsi que de nos cordes vocales.

10.2. Lors de la seconde soutenance

Lors de cette soutenance, afin de gagner du temps, nous avons choisi d'utiliser des musiques externes libres de droit. Cela signifie que les compositeurs de ces musiques devaient seulement être crédités et l'usage ne devait pas être commercial. Cette restriction d'usage correspond à notre projet, celui-ci étant présenté uniquement dans le cadre de nos études à l'EPITA et n'étant pas destiné à être vendu.

Nous avons alors lancé la recherche de musiques pour l'ambiance sonore globale du jeu en fonction du niveau et de la situation du joueur, comme le passé, le présent ou le futur.

Une dizaine de bruitages furent par ailleurs enregistrés afin de permettre leur implémentation dans le jeu. Parmi ces enregistrements, se trouvaient les bruits de pas sur les différentes textures comme l'herbe ou la pierre.

Ces différents bruitages furent enregistré à l'aide d'un téléphone en effectuant plusieurs actions comme marcher dans l'herbe sur de la pierre, sur du gravier, ou encore en faisant s'entrechoquer différents matériaux.

Pour les musiques du jeu, trois artistes furent choisis. Chacun de ces compositeurs possédait au moins une chaîne YouTube, celle-ci regroupant la grande majorité de leurs créations. Les deux premiers sont les frères Fiechter qui possèdent, par ailleurs, trois chaînes YouTube. En ce qui concerne le dernier artiste, celui-ci possède la chaîne YouTube *Approaching Nirvana*, ses musiques étant couramment utilisées dans le domaine vidéoludique.

10.3. Etat final

Afin de finir le projet, nous enregistrâmes les sons manquant pour notre jeu à l'aide d'*Audacity*. Parmi ces sons, se trouvaient les bruits des monstres ainsi que ceux d'objets divers. Le saut du joueur émet maintenant un son, celui-ci ayant été enregistré.

Par la suite, la totalité des sons utiles au projet furent ajoutés et implémentés dans l'éditeur du jeu, *Unity*.

Nous ajoutâmes également dans le jeu les différentes musiques présentées lors de la seconde soutenance. Celles-ci diffèrent en fonction de l'époque dans laquelle se situe le joueur, c'est-à-dire en fonction du niveau.

11. L'interface

L'interface du jeu se doit d'être simple à utiliser tout en fournissant les informations essentielles au joueur. Elle doit par exemple afficher des statistiques telles que la vie, le nombre de pièces récoltées, etc.

11.1. Lors de la première soutenance

Lors de cette soutenance, nous avons effectué le menu principal accueillant les joueurs lors du lancement du jeu. Celui-ci proposait plusieurs options telles que lancer une partie, accéder aux options et quitter le jeu.

Les boutons « *Play* » et « *Exit* » étaient alors fonctionnels. Le premier lançait notre niveau de test tandis que le second était utilisé pour quitter l'application.

Les options n'étaient pas encore implémentées, lors du clic sur le bouton, aucune action n'était effectuée.

Enfin, les textures de chacun des boutons n'étaient pas implémentées, celles-ci étaient les textures par défaut du logiciel *Unity* et n'étaient que provisoires.



| Le menu principal du jeu lors de la première soutenance

11.2. Lors de la seconde soutenance

Nous décidâmes alors d'améliorer notre menu principal en y apportant quelques modifications. Nous voulions notamment nous détacher des textures par défauts des boutons *Unity* qui n'étaient pas vraiment à notre goût. Nous changeâmes alors le fond blanc des trois boutons, « *Play* », « *Options* » et « *Exit* » pour un fond désormais invisible, mais qui se noircissait lorsque la souris passait dessus, pour que le joueur visualise bien le bouton sur lequel il s'apprêtait à appuyer, et par souci d'esthétique. Les textes des boutons étaient désormais en rouge et en lettres capitales pour les rendre plus visibles.

La couleur du titre du jeu, « *Super Platformer* », fut également modifiée afin d'être en accord avec la couleur de nos boutons et afin d'améliorer la lisibilité pour l'utilisateur final.



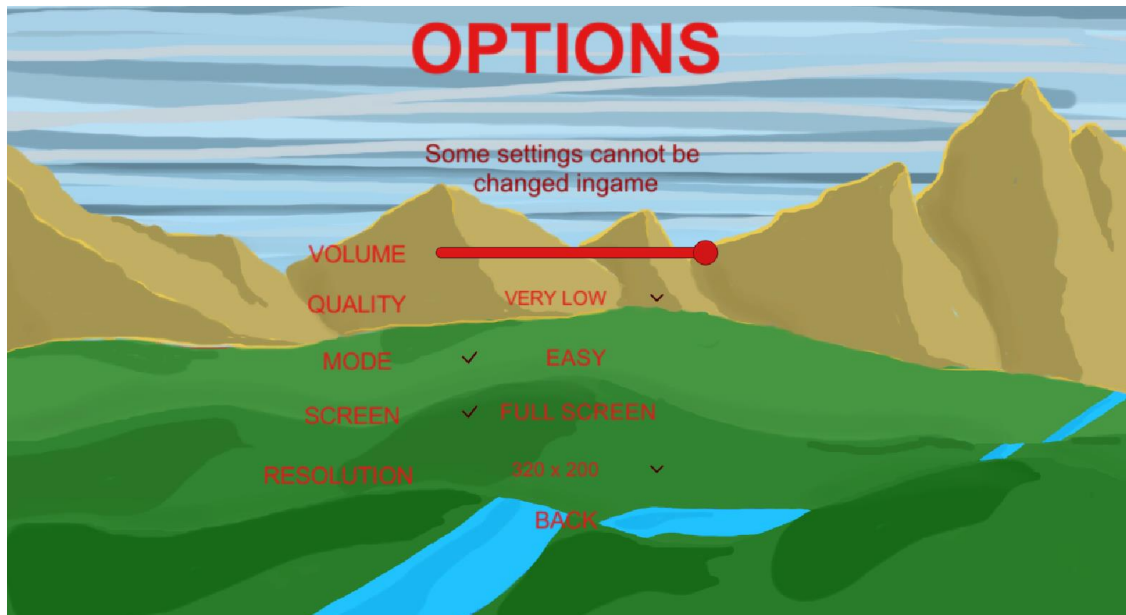
Le menu principal du jeu lors de la deuxième soutenance

L'ensemble des boutons implémentés lors de la soutenance précédente restaient fonctionnels tandis que ceux permettant d'accéder aux options du jeu renvoyait dorénavant sur une nouvelle interface : celle des options.

Concernant ce menu, il était possible de régler le volume ainsi que la qualité des graphismes, d'activer ou désactiver le plein écran et le « *mode easy* » qui permettait de recevoir moins de dégâts, et enfin de choisir sa résolution. De plus, un bouton Back permettait de revenir en arrière, au menu principal.

Nous avons, de plus, appliqué l'ensemble des changements esthétiques opérés sur le menu principal dans celui des options, ce qui rendait le tout plus homogène.

Tous les boutons furent implémentés et changeaient ainsi réellement les options du jeu.



| *Les options du jeu, entièrement personnalisables par l'utilisateur*

Un des défauts que présentaient les options était qu'elles affichaient les paramètres par défaut de l'utilisateur et non pas les options actives. L'utilisateur n'avait alors aucun moyen de savoir ses paramètres sans réappliquer la totalité des paramètres.

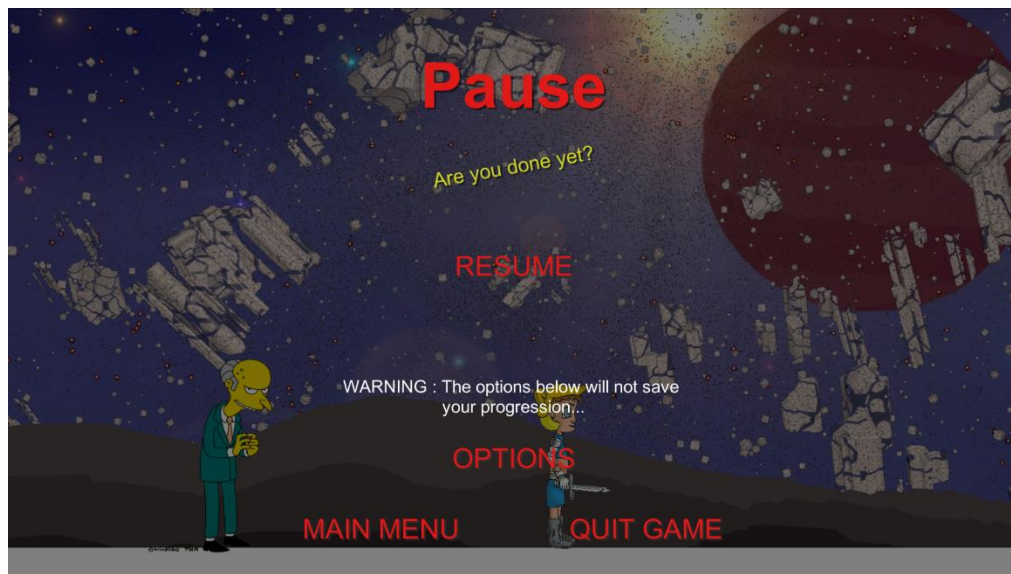
11.3. Etat final

Le menu principal n'a subi qu'une unique modification, celui-ci lance maintenant le premier niveau au lieu du niveau utilisé pour les tests.

Les options, bien que fonctionnelles, n'affichait pas les paramètres en vigueur concernant la résolution d'écran. Le menu affichait la plus petite résolution prise en charge par le jeu au lieu de la résolution actuelle. Ce défaut a maintenant été corrigé.

Un problème que possédait notre jeu est pour les écrans à forte résolution et haute densité de pixel : les textes et boutons des menus étaient pixellisés et s'adaptaient mal à ces types d'écrans. L'expérience utilisateur était alors dégradée. Ce problème a maintenant été réglé.

Un menu de pause a été ajouté, celui-ci permettant à l'utilisateur de quitter le programme, revenir au menu principal ou encore changer les options de jeu. Afin d'accéder à ce menu, l'utilisateur doit simplement appuyer sur la touche « Echap », celle-ci étant communément utilisée dans le domaine vidéoludique.



| *Le menu de pause*

De plus, lorsque l'utilisateur est en jeu, celui-ci a maintenant des indications sur la vie restante et sur le nombre de pièces récoltées. En effet, le jeu comporte une barre d'affichage en haut à gauche pour la vie et un logo de pièce suivi d'un numéro pour le nombre de pièces en haut à droite.

Enfin, le choix de mettre le texte en rouge nous avait posé plusieurs problèmes de lisibilité, notamment avec le contraste par rapport au fond vert. Des ombres ont donc été rajoutées sur l'ensemble des textes afin de mieux les discerner de l'image de fond. Cela donne également une impression de profondeur à l'utilisateur final, rendant le tout plus agréable à lire.



| *Aperçu des ombres sur les textes des menus*

12. Le site internet

Le site internet est une partie essentielle de la présentation d'un projet. C'est le support de référence qui permet de donner un rendu plus professionnel pour un acheteur potentiel. Il permet également de donner différentes informations à propos du produit, par exemple une description du jeu, des images, des extraits ou bien une actualité sur le développement de celui-ci.

Le site du projet est disponible à l'adresse « anvstin.github.io/SuperPlatformer ».

L'adresse suivante « github.com/anvstin/SuperPlatformer » permet de visualiser le code source du site, celui-ci étant en libre accès.

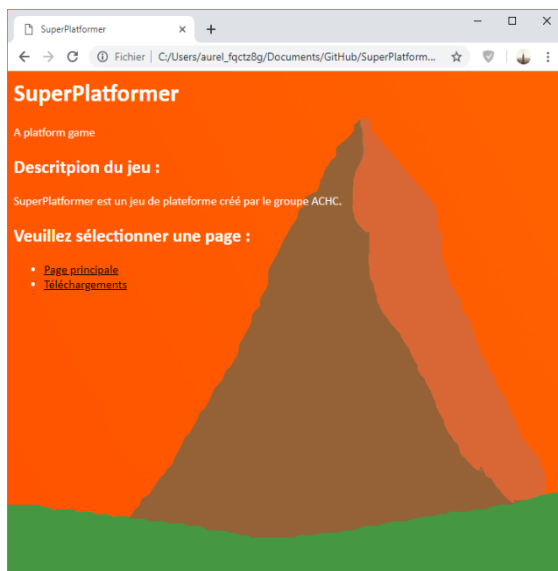
12.1. Lors de la première soutenance

Lors de la première présentation du projet, le site internet avait déjà été commencé. Nous avons alors choisi d'utiliser GitHub afin d'héberger le site web, celui-ci étant gratuit et simple à mettre en place.

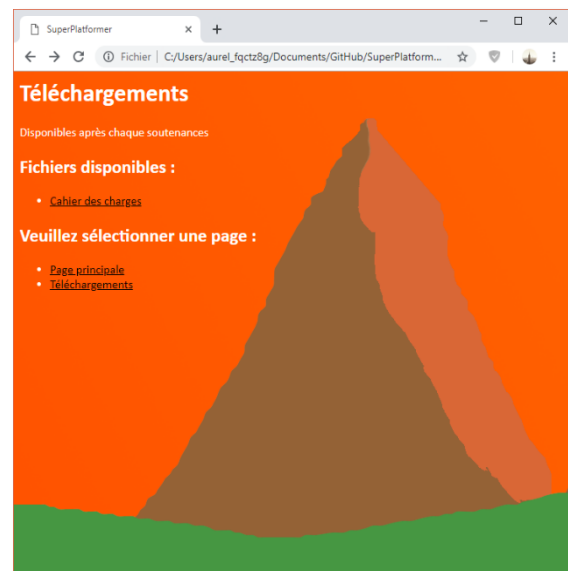
En outre, le site comportait une page d'accueil ainsi qu'une page de téléchargements. Ces dernières étaient peu complètes en termes de contenu et étaient peu attractives pour l'utilisateur final.

Enfin, une section dédiée à la navigation avait été mise en place. Celle-ci avait la particularité de s'appliquer sur l'ensemble des pages web à l'aide du langage de programmation JavaScript.

Voici le site internet lors de la première soutenance :



La page principale



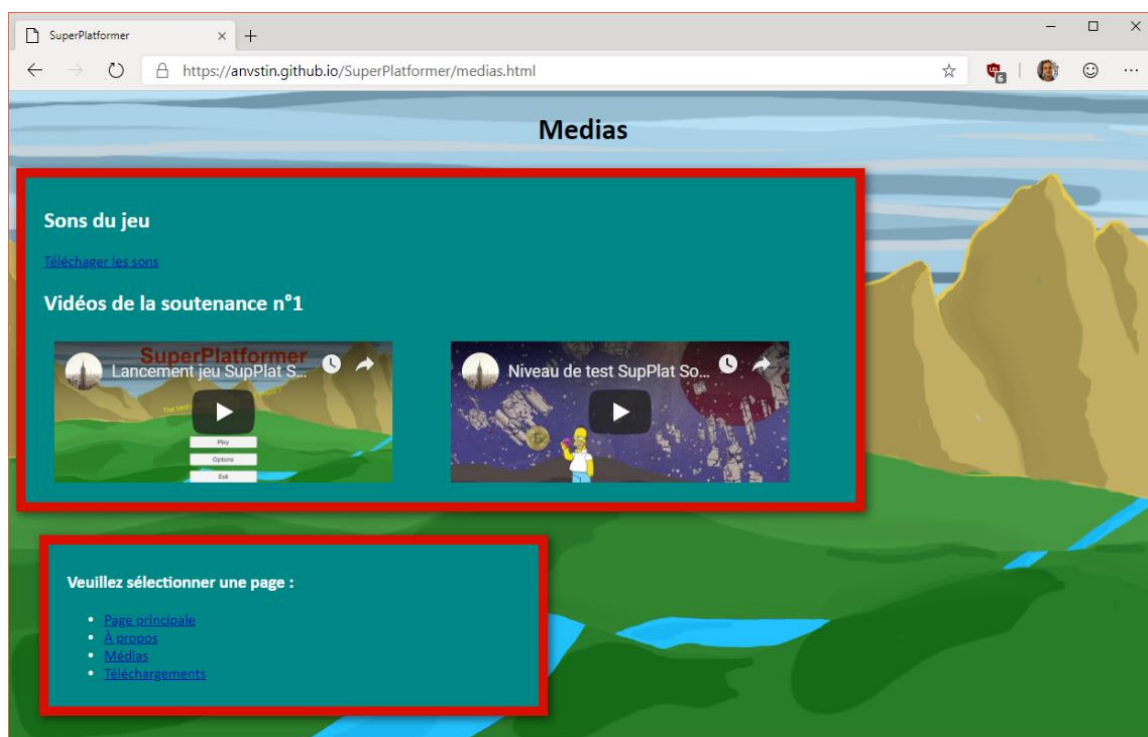
La page de téléchargement

12.2. Lors de la seconde soutenance

Ensuite, lors de notre deuxième présentation du projet, le site avait subi une refonte totale, aussi bien au niveau du code qu'au niveau de l'interface. Cela avait pour but d'améliorer l'expérience utilisateur mais aussi de simplifier la modification et l'ajout de contenu au site.

Les téléchargements avaient également été mis dans un tableau afin d'aiguiller plus facilement l'utilisateur final. Chaque document était alors séparé en fonction de chaque soutenance. En outre, les dates de création de chaque document avaient également été ajoutées, toujours pour augmenter la lisibilité.

Enfin, une page regroupant l'ensemble des médias avait été ajoutée. Cette dernière regroupe l'ensemble des médias du jeu, comme les sons, mais aussi les vidéos présentées durant la première soutenance.



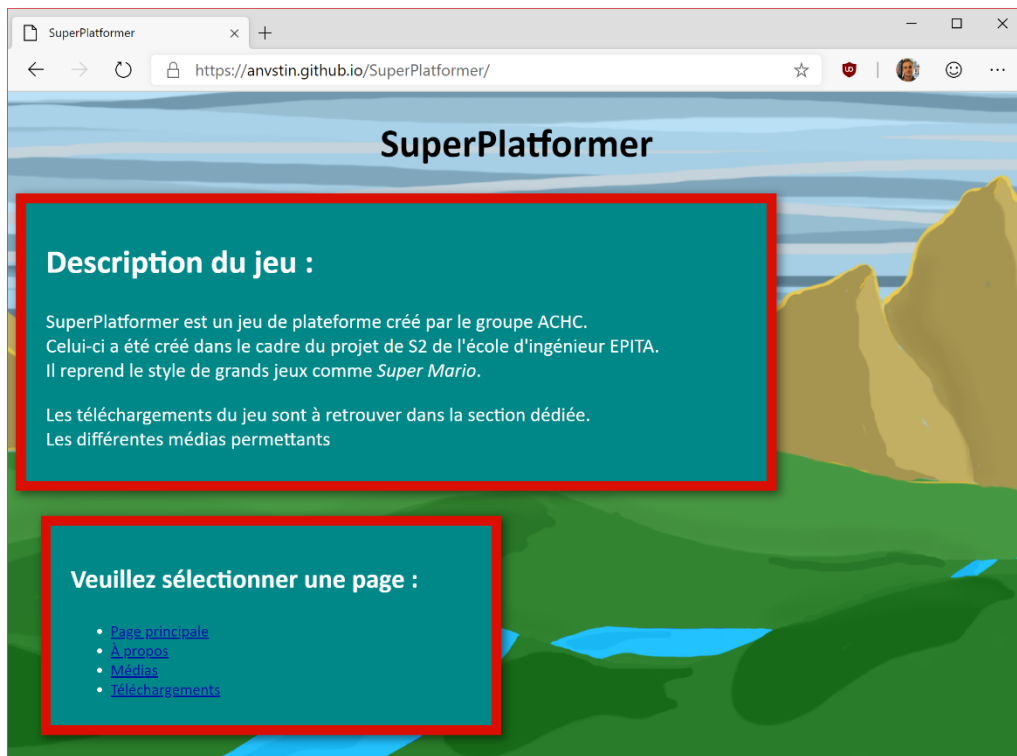
La nouvelle interface et la page média

12.3. Etat final

Précédemment, l'image de fond du site ne s'adaptait pas à la hauteur de l'écran de l'utilisateur. Cela entraînait l'apparition d'une zone verte peu esthétique en dessous de l'image. Ce problème a maintenant été réglé, l'image prenant la taille du texte affiché sur la page web.

Afin de rendre l'interface plus lisible pour l'utilisateur, les polices ont été en outre grossies.

Bien que le site internet n'ai subi que peu de modifications au niveau de l'interface, celui-ci comprend maintenant des pages bien plus complètes. Par exemple, les descriptions des membres du projet ont été ajoutées et la page principale comporte un message d'accueil ainsi qu'une brève description du projet. Les descriptions servent d'éléments informatifs pour l'ensemble des visiteurs du site.



| La page d'accueil du site

De plus, les documents relatifs à la deuxième soutenance ont été ajoutés sur la page de téléchargements. En outre, une brève procédure de téléchargement et d'installation du jeu est affichée sur le site afin de faciliter la prise en main du programme par les utilisateurs.

13. L'installateur

Afin de créer un jeu facile à utiliser, un installateur est nécessaire. Celui-ci permet, en quelques clics de souris, d'installer le jeu ainsi que l'ensemble des ressources nécessaires à son bon fonctionnement. Une fois le jeu installé, l'utilisateur doit être également capable de désinstaller le jeu s'il le désire.

L'installateur d'un programme se doit d'être simple à utiliser, tout en lui fournissant les informations relatives au programme de manière claire. Par exemple, l'utilisateur doit pouvoir choisir où installer le jeu.

13.1. Lors des deux premières soutenances

Comme précisé dans le tableau d'avancement, l'installateur serait mis en place uniquement lors du rendu final du projet. En effet, il est inutile de créer un programme d'installation pour un jeu non fini et dont le contenu est destiné à changer : cela nous imposerait alors la re-crédation de l'installateur à chaque modification apportée au jeu.

13.2. Etat final

Pour l'installation du projet, nous avons décidé d'utiliser un logiciel dont le code est en libre accès : « *Inno Setup* ». Nous n'avons donc pas besoin de payer des droits d'utilisation. Celui-ci nous permet de créer un installateur facilement tout en restant personnalisable. Il fournit également un outil pour désinstaller le jeu.

La facilité de création de l'installateur du jeu est importante, notamment si nous devons créer une nouvelle version du jeu, par exemple lors de correction de bugs.

14. Répartition des tâches

14.1. Modifications lors de la première soutenance

Lors de cette soutenance, nous avons modifié l'appartenance de certaines tâches suivant les préférences ou les facilités des membres du groupe.

Voici deux tableaux récapitulatifs des différents changements.

Répartition des tâches (Cahier des charges)

Tâches	Aurélien	Cloé	Corentin	Heiko
Commandes	S	R		
Mécaniques de jeu			R	S
Animations des modèles		S	R	
Scénario		R	S	
Intelligences artificielles	R	S		
Graphismes	S			R
Bandes sonores		S		R
Interface			S	R
Site Internet	R			S
Installateur	R		S	

Répartition des tâches (Première soutenance)

Tâches	Aurélien	Cloé	Corentin	Heiko
Commandes	S	R		
Mécaniques de jeu	S	R		
Animations des modèles			R	S
Scénario		R	S	
Intelligences artificielles	R	S		
Graphismes			R	S
Bandes sonores		S		R
Interface			S	R
Site Internet	R			S
Installateur	R		S	

R : Responsable

S : Suppléant

14.2. Répartition des tâches lors des soutenances deux et trois

Nous avons décidé de ne modifier aucune partie de la répartition des tâches que ce soit lors de la seconde soutenance ou lors de la soutenance finale. Notre volonté de répartir les tâches de manière équitable, notamment en termes de temps, s'est révélée payante. En effet, de manière générale, l'ensemble du groupe a effectué dans les temps le travail lui ayant été assigné. De plus, personne n'avait à faire des tâches qui lui déplaisaient, cela a grandement facilité le travail.

14.3. Déroulement des tâches

Dès le début nous avons convenu de deux grosses parties, la partie technique et la partie créative dont les membres n'allaient pas changer. Nous avons effectué ce choix car nous avons décidé de faire nous-mêmes tous nos graphismes et surtout qu'il est plus pénible et consommateur de temps de travailler sur le croquis ou le code de quelqu'un d'autre.

Nous avons tout de même souvent pris part à des tâches qui ne nous étaient pas forcément assignées, parfois pour faire avancer le projet lorsque nous avions du retard, mais également parfois par simple curiosité. Ainsi, aucun membre du groupe ne s'est retrouvé à faire des tâches qui ne lui convenaient pas, étant donné que les tâches étaient réparties selon les préférences de chacun. Enfin, aucun membre n'a eu le sentiment de faire la même chose durant tout le projet.

De manière générale, les tâches étaient bien réparties entre les membres du groupe bien que la partie des graphismes nous ait pris une quantité non négligeable de temps.

15. Avancement du projet

Lors du développement de notre projet, nous avons choisi de modifier notre tableau d'avancement. D'une manière générale, cela peut s'expliquer par une mauvaise prise en compte du temps requis pour l'accomplissement de certaines parties.

Voici un récapitulatif de notre avancement lors de chaque soutenance ainsi que les modifications apportées au tableau entre chacune de celles-ci.

	Soutenance 1		Soutenance 2			Soutenance finale
Avancement	Sur le cahier des charges	Lors de la soutenance n°1	Sur le cahier des charges	Sur le rapport de la soutenance n°1	Lors de la soutenance n°2	Sur la totalité des documents
Commandes	80%	60%	100%	80%	80%	100%
Mécaniques de jeu	40%	40%	75%	75%	75%	100%
Animations des modèles	0%	1%	50%	50%	50%	100%
Scénario	50%	50%	90%	90%	80%	100%
Intelligences artificielles	20%	20%	40%	40%	50%	100%
Graphismes	20%	25%	50%	50%	50%	100%
Sons	0%	5%	40%	40%	40%	100%
Site internet	10%	10%	35%	35%	50%	100%
Installateur	0%	0%	0%	0%	0%	100%

Dans les temps

En avance

En retard

Modification apportée

15.1. Lors de la première soutenance

Lors de notre première présentation du projet, nous avons pris un léger retard sur la réalisation des commandes. Lors de l'écriture du cahier des charges, nous avons mal pris en compte la nécessité de l'implémentation de certaines fonctionnalités du jeu pour pouvoir finaliser cette partie. En effet, sans les animations d'attaque, d'accroupissement et de déplacement les commandes ne sont pas totalement fonctionnelles.

Nous avons par ailleurs pris de l'avance sur trois parties : les animations, les graphismes et les sons du jeu. Cela nous a par la suite permis de compenser divers problèmes qui auraient pu nous mettre en retard.

De manière plus générale, le groupe était en accord avec les prévisions.

15.2. Lors de la seconde soutenance

Sur le tableau présenté précédemment, nous pouvons constater un léger retard sur la partie du scénario lors de la seconde soutenance. Ce retard a eu différentes raisons. Tout d'abord nous avons réparti notre temps sur d'autres parties, celles-ci nous semblaient plus urgentes. D'autre part, nous n'avons pas pris en compte la conception des niveaux dans les pourcentages de la partie scénario, au départ. Ceux-ci représentaient donc les 20% qu'il nous restait à réaliser dans la partie du scénario.

Nous avons cependant pris de l'avance dans les parties des intelligences artificielles et du site internet, et étions dans les temps pour la majorité des tâches.

15.3. Avancement actuel

Tâches	Première soutenance	Seconde soutenance	Avancement final
Commandes	60%	80%	100%
Mécaniques du jeu	40%	75%	100%
Animations des modèles	1%	50%	100%
Scénario	50%	80%	100%
Intelligences artificielles	20%	50%	100%
Graphismes	25%	50%	100%
Sons	5%	40%	100%
Site internet	10%	50%	100%
Installateur	0%	0%	100%
	Dans les temps	En avance	En retard

Pour ce projet, nous sommes dans les temps sur l'ensemble des catégories. Notre projet est totalement fini, bien que nous aurions aimé ajouter des fonctionnalités supplémentaires.

16. Déroulement du projet

16.1. Outils utilisés

Lors de ce projet, nous avons bien évidemment utilisé *Unity*, la plateforme de développement du jeu. Celui-ci, contrairement aux apparences, s'est révélé être relativement simple à manipuler, en faisant un bon outil de travail.

De plus, nous avons utilisé des logiciels comme *Gimp* et *Photoshop* pour la création des différentes textures du jeu. Les textures ont également été réalisées à l'aide d'une tablette graphique, cela permet ainsi une meilleure numérisation des textures tout en donnant un aspect plus professionnel.

Afin de créer les différents sons du jeu, le logiciel d'enregistrement et de traitement audio *Audacity* a été utilisé, en plus de l'enregistreur audio de nos téléphones.

Pour la création des différents documents de soutenance, que ce soient les rapports ou les diaporamas, nous avons utilisé la suite *Office* de *Microsoft*. Les outils collaboratifs qui y sont présents nous ont permis la rédaction et l'enregistrement d'un document en temps réel. Ces actions sont possibles même si plusieurs membres du groupe sont présents sur le document en cours d'édition.

Enfin, afin de maximiser l'échange d'information et la productivité du groupe, nous avons créé un serveur sur le logiciel *Discord*. Celui-ci nous permet de communiquer au travers de canaux de discussion dont l'ensemble des membres possèdent l'accès. Ceux-ci peuvent être vocaux comme textuels.

16.2. Difficultés rencontrées

Le choix d'effectuer l'ensemble des graphismes du jeu par nous-même s'est avéré représenter un réel défi, ceux-ci étant extrêmement longs à créer et avec des résultats parfois peu convaincants. Cela nous a alors obligé à créer de nombreux graphismes qui ne seront finalement pas implémentés dans le jeu. Afin de pallier à cela, nous avons décidé de réduire la quantité de graphismes afin d'économiser du temps tout en garantissant la qualité de ceux-ci.

Nous avons choisi d'effectuer la bande sonore par nous-même, mais pour une meilleure expérience utilisateur nous avons préféré incorporer des musiques libres de droits à notre projet, en plus des bruitages déjà créés par nos soins.

En outre, un autre défi à relever fut la conception d'un site web. En effet, aucun membre du groupe n'avait les bases requises à sa création. Afin de réaliser celui-ci, nous avons alors dû apprendre à la fois le langage HTML ainsi que le CSS, ces deux langages étant requis.

Enfin, le dernier défi que nous ayons eu à relever fut la gestion du temps. Respecter notre tableau d'avancement s'avéra, non seulement dur à réaliser pour les deux premières soutenances mais le fut encore plus pour la soutenance finale. En effet, nous avions mal pris en compte le temps restant entre la deuxième et la troisième soutenance. En effet, nous pensions posséder deux mois avant la présentation, comme pour les deux premières soutenances, mais nous n'en possédions en réalité qu'un unique. Nous avons alors dû mettre les bouchées doubles pour finir le projet.

16.3. Apport du projet au groupe

16.3.1. Ressenti de CARLIER Heiko

Ayant été en section Science de l'Ingénieur, j'avais déjà effectué des projets dans des conditions similaires. Ceux-ci m'avaient beaucoup apporté, autant en termes de connaissances, qu'en travail d'équipe. Cela a encore été le cas avec ce projet qui était plus concret. En effet, en Sciences de l'Ingénieur, nous faisons majoritairement des expériences sur un sujet précis tandis que le projet de ce second semestre était surtout orienté dans la création d'un produit.

De manière générale, j'ai trouvé les autres membres du groupe très professionnels et toujours enclins à travailler. Cela a participé à la bonne ambiance du groupe.

16.3.2. Ressenti d'ESCUDIER Cloé

Ce projet a été une expérience très enrichissante pour moi, et qui me conforte dans le choix de ma formation. Chercher des solutions à des problèmes qui paraissent impossibles à résoudre, et ce à chaque avancée du jeu, était très difficile, mais lorsque l'on trouvait la solution, la fierté surpassait la frustration. De plus, chercher soi-même et découvrir le langage ainsi que la plateforme *Unity* était vraiment très intéressant.

Enfin, le travail d'équipe a été très important, et je suis contente d'avoir été dans un groupe avec une bonne ambiance de travail et d'entraide, de la bonne humeur, et du sérieux dans les moments importants.

16.3.3. Ressenti de PION Corentin

Venant de la filière S.S.I, j'avais déjà fait quelques projets mais plus sur une partie réalisation (Dessin 3D / Construction). Ce fut mon premier projet invoquant le thème de l'informatique. Les débuts furent difficiles, il fallait apprendre à utiliser les différents logiciels notamment *Unity*. De manière générale, il nous a appris une certaine forme d'autonomie et de gestion de groupe. L'ambiance de travail étant présente, elle a permis à notre groupe d'être plus efficace ainsi que de passer des bons moments.

16.3.4. Ressenti de VISENTIN Aurélien

Le projet a été un défi important à relever, celui-ci différant de manière significative de ceux faits les années précédentes au lycée, que ce soit pour les Sciences de l'Ingénieur (S.I.) ou pour les Travaux Pratiques Encadrés (T.P.E.).

Dans mon cas, le temps a été un ennemi redoutable, les soutenances étant toujours proches des partiels. Ce projet m'a donc permis de mieux m'organiser, de me concentrer sur les tâches essentielles du projet ainsi que sur les points importants de mes révisions, sans entrer dans des détails superflus.

Travailler sur un projet nous tenant à cœur pendant plusieurs mois m'a été extrêmement enrichissant, que ce soit en termes d'entente de groupe ou bien en maîtrise de la conception et du développement d'algorithmes complexes.

Enfin, prendre en main la plateforme de développement Unity m'a permis de me rendre compte du processus de développement d'un jeu, celui-ci étant souvent bien différent des travaux pratiques effectués chaque semaine à l'EPITA.

17. Prochaines étapes : industrialisation et généralisation

En imaginant une industrialisation du jeu Super Platformer, dans une prochaine étape afin d'en faire un projet utilisable en production, une organisation plus structurée pour le développement du projet devrait être mise en œuvre.

Des fonctionnalités orientées vers un grand nombre d'utilisateurs devraient également être considérées, avec un plus large déploiement vers une communauté de joueurs.

17.1. Communauté d'utilisateurs

La mise en place d'une communauté d'utilisateurs du jeu, via une plateforme de contribution sur internet, permettrait de récupérer les avis des utilisateurs et d'alimenter le projet. Les joueurs pourraient apporter leurs commentaires, et des propositions d'améliorations. En fonction de leur nombre, ces dernières alimenteraient la liste des évolutions de notre projet en l'orientant en fonction des besoins du plus grand nombre.

En outre, cette communauté pourrait contribuer à tester les nouvelles versions du jeu et remonter des avis et anomalies avant leur mise en production. Un mécanisme de récompense pourrait être mis en place pour les joueurs contribuant le plus à la communauté et proposant les meilleures suggestions, ou ayant remonté des anomalies, par exemple sous forme d'accès gratuit aux nouvelles versions en bêta, ou à des fonctionnalités étendues.

Enfin, l'animation de cette communauté, à laquelle notre projet contribuerait, constituerait un excellent canal de communication pour rendre notre jeu populaire et lui faire de la publicité.

Des plateformes communautaires comme Steam facilitent aujourd'hui grandement la publication de jeux vidéo et leur adoption par une très large population de joueurs et l'accès à leurs contributions : store.steampowered.com/reviews/.

17.2. Industrialisation du projet

17.2.1. Organisation de projet Agile

Afin d'intégrer au fur et à mesure de nouveaux développements dans le jeu, tout en conservant un projet qui soit fonctionnel en production, une organisation projet de type Agile pourrait être mise en place. Cette organisation intégrerait des représentants des joueurs dans le développement, s'appuyant sur les retours de la communauté, afin de prendre en compte au fur et à mesure les nouvelles fonctions prioritaires, et d'avoir ainsi à tout moment le meilleur projet possible pour les utilisateurs.

17.2.2. Mise en place de tests automatiques

La communauté permettrait de nous aider à tester les futures versions beta du jeu avant leur mise en production. Mais auparavant, les développements correspondant aux versions alpha, au niveau de notre équipe, devraient être testés en profondeur, par la mise en place de scénarios de tests qui seraient exécutés régulièrement, et de manière automatique. Il existe un ensemble d'outils pour la plateforme *Unity* dédiés aux tests, par exemple en suivant le lien suivant : unity3d.com/unity/qa/test-tools

17.2.3. Automatisation de la construction des versions

Cette automatisation des tests serait couplée avec la constitution automatique des versions packagées du jeu, sur ce que l'on appelle une plateforme d'intégration continue. Une solution Open Source (dont le code est en libre accès) comme *Jenkins* pourrait être utilisée, comme décrit sur le lien suivant :

blogs.itmattersgames.com/continues-integration-with-unity3d-and-jenkins-part-1

Couplée à une solution de gestion de configuration, cette construction automatique permettrait un déploiement fiable et très régulier de nouvelles versions, ce qui permettrait d'avoir des retours encore plus fréquents des joueurs et de mieux répondre à leurs attentes. C'est de cette manière que sont développés les jeux et plus généralement les logiciels ciblant le grand public, avec de nombreux utilisateurs.

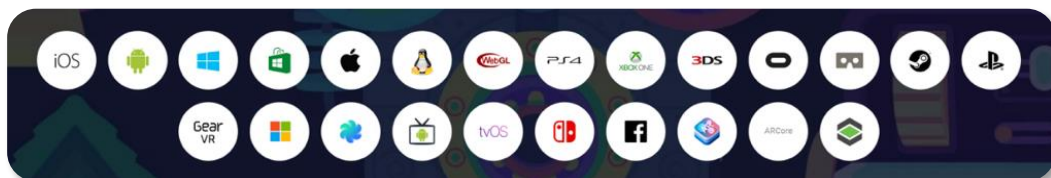
17.2.4. Capitalisation des composants

Certaines fonctionnalités de notre jeu, et potentiellement réutilisables pour d'autres projets futurs, devraient être capitalisées. Les différents éléments du projet ou *Assets* pourraient par exemple être exportés d'un projet à l'autre sous forme de paquet personnalisé : docs.unity3d.com/Manual/HOWTO-exportpackage.html

17.3. Déploiement sur plusieurs plateformes

Enfin, *Unity* permet de construire des jeux pour plus de 25 plateformes : Web, Desktop (Windows, Mac ou Linux), Mobile (Android et iOS), réalité augmentée (Hololens), réalité virtuelle (Oculus rift ou HTC Vive), console (Xbox et PlayStation) :

unity3d.com/unity/features/multiplatform



Notre jeu devrait prendre en compte cette capacité cross-plateforme de *Unity*, même si elle pourrait nécessiter de développer des versions spécifiques, car leur construction en est grandement facilitée. Cela permettrait d'envisager de déployer notre jeu de manière très large, adresser un plus grand nombre d'utilisateurs avec de nouvelles expériences de jeu, et nous adapter ainsi aux nouvelles tendances, le domaine du jeu vidéo étant l'un des plus riches en termes d'innovation : cet aspect a beaucoup contribué au plaisir que nous avons pris à développer notre projet.

18. Conclusion

Notre groupe a su trouver une grande motivation à travers ce projet, et nous l'avons gardé précieusement jusqu'à la fin de l'année. En effet, la création de ce jeu vidéo nous a permis de découvrir ce qu'était un travail de groupe, en situation, à longue durée, avec ses réussites et ses échecs.

D'autre part, certains problèmes ont mis nos nerfs à rude épreuve mais le résultat final prévaut et prouve que nous avons su les résoudre.

Enfin, nous sommes tous d'accord pour dire que ce fut une riche expérience sur le plan décisionnel. Il a fallu apprendre à s'écouter, à exposer ses points de vue mais aussi à accepter la critique et en tirer le meilleur possible.

Ainsi s'achève ce rapport de projet et par la même occasion, notre première année à l'EPITA. L'équipe ACHC a été très heureuse de vous faire partager le travail qu'elle a fourni tout au long de ce semestre.