



PROJET GALAXY TRAVELLER

by Cygnus

Rapport de soutenance 1

Mars 2023

BELPERIN Noé, BLANC Johan,
DEFONTAINE JOLIVET Emilien, THOME Aubin

Table des matières

I	Introduction	4
II	Présentation globale du projet	5
1	Origine	5
2	Nature	5
3	Équipe	6
3.1	BELPERIN Noé	6
3.2	BLANC Johan	6
3.3	DEFONTAINE JOLIVET Emilien	7
3.4	THOMÉ Aubin	7
III	Répartition des tâches	8
IV	Avancement des tâches et objectifs	9
4	Global	9
5	Par tâches	9
5.1	Joueur	9
5.2	Armes et collectables	10
5.3	Combats	11
5.4	Ennemis	12
5.5	Parcours	12
5.5.1	Plateformes	12
5.5.2	Terrain de spawn	13
5.5.3	Surface de respawn	14
5.6	Menus	14
5.7	Réseau et multijoueur	15
5.8	Graphismes et animations	16
5.9	Site web	18
V	Conclusion	19

Première partie

Introduction

Ces pages constituent le rapport de la première soutenance technique du groupe Cygnus pour leur projet Galaxy Traveller.

Ce rapport de soutenance technique a pour but de faire un état des lieux des avancées techniques de notre projet de jeu vidéo à compter du lancement du projet, en Septembre 2023 jusqu'à cette première soutenance technique du 20 mars 2024.

Pour rappel, Galaxy Traveller est un jeu en 3 dimensions qui se veut être à la jonction du “Die and Retry” et du jeu de plateforme, qui peut se jouer en solo ou à deux joueurs.

Dans un premier temps, nous vous présenterons le projet et l'équipe. Ensuite, nous aborderons la répartition des tâches et leur avancement. Puis, dans un troisième temps, nous aborderons les objectifs pour la prochaine soutenance technique de Juin et enfin, nous conclurons ce rapport de soutenance.

Deuxième partie

Présentation globale du projet

1 Origine

Ce travail a comme origine le projet de première année de l'*EPITA* qui consiste à développer un jeu vidéo en 3D, jouable sur la plateforme *Windows* et en mode coopération (donc en multijoueurs), le tout développé en C#.

Dès l'annonce du projet, nous avons immédiatement constitué notre équipe et s'est rapidement mis en place une excellente synergie de groupe. Très vite, nous nous sommes tournés vers le jeu de plateforme que chacun affectionne et qui se présente comme un intemporel dans le monde du jeu vidéo. Nous avons tous joué à ce type de jeu, en solo ou en multijoueurs et, créer un nouveau monde et de nouveaux mécanismes pour un jeu de plateforme nous a tout de suite motivés et fédérés autour de ce projet.

Après de nombreuses délibérations, nous avons décidé de concevoir un jeu vidéo avec comme imaginaire l'espace et au vu des caractéristiques de celui-ci, nous lui avons rapidement trouvé un nom : "*Galaxy Traveller*"!

2 Nature

Aujourd'hui, les jeux sur mobile, PC ou encore console sont multiples et continuent de se réinventer. Cependant, chacun possède son style, sa mécanique et ses caractéristiques. Bien qu'étant parmi les premiers jeux ayant vus le jour, le *jeu de plateforme* est toujours populaire, continue de perdurer et à se décliner en différentes versions, qui chacune, apporte son lot de nouvelles fonctionnalités.

Dans cette dynamique de toujours remettre au goût du jour le jeu de plateforme, nous avons imaginé notre jeu, un *plateforme 3D de coopération*. A chaque niveau, les joueurs doivent survivre à un parcours d'obstacles et récupérer différentes capacités (armes, armures, potions diverses) avant d'atteindre une planète et d'en affronter le boss pour pouvoir ensuite accéder au parcours et au boss suivant.

Dans ce jeu, votre objectif est de passer d'une planète à l'autre en surmontant une série d'épreuves de plus en plus complexes sur votre chemin vers le boss final. Les planètes et leurs atmosphères forment des mondes distincts. Chaque monde est différent

des autres et il faut affronter et éliminer un boss contrôlé par une intelligence artificielle pour quitter chaque planète en sachant que la difficulté est graduelle au fil de l'avancée dans le jeu. Pour passer à la planète suivante, vous devez emprunter un chemin de plateformes complexe, parsemé d'obstacles et de dangers. Vous devez naviguer sur des plateformes amovibles, faire très attention aux plateformes qui disparaissent sous vos pieds, et même faire face à de mystérieux maléfices, tels que des visions altérées et des hallucinations. Ces éléments perturbateurs ont pour but de déstabiliser le joueur, le forçant à s'adapter et à développer de nouvelles stratégies pour progresser.

"*Galaxy Traveller*" offre une expérience de jeu unique, où chaque planète est une aventure en soi, et où la persévérance et la maîtrise des mécanismes de jeu sont essentielles pour explorer l'univers. Vous vous lancez dans un voyage spatial palpitant, où seuls les plus habiles et stratèges pourront affronter les spécificités de chaque planète et triompher des boss qui se dressent sur leur chemin.

Notre jeu intègre un mode multijoueurs qui demande à nos joueurs de faire preuve de stratégie tout au long des niveaux pour se compléter et arriver à vaincre le boss de fin. Ainsi des stratégies pourraient être mises en place entre les joueurs, l'un se concentrant par exemple sur la régénération et l'autre sur l'attaque.

3 Équipe

Notre équipe est nouvellement composée de quatre membres à la suite d'un départ.

3.1 BELPERIN Noé

Co-fondateur de l'entreprise Cygnus, j'ai comme missions principales la gestion de l'intelligence artificielle qui se caractérise par les combats de boss, j'ai de plus comme responsabilités la partie graphisme ainsi que le son et la musique de notre jeu. J'assiste Johan Blanc dans la réalisation de l'interface utilisateur et également dans la mise en place des armes et collectables.

3.2 BLANC Johan

Dans le cadre de notre projet, j'ai été nommé chef de groupe. J'essaie de motiver tout le groupe pour mener à bien la réalisation du projet dans les temps impartis. Je suis, de plus, responsable de la partie "Armes et Collectables" ainsi que de la partie "Interface Utilisateur". Pour terminer, je suis suppléant de Noé pour l'implémentation de l'intelligence artificielle ainsi que des sons et musiques.

3.3 DEFONTAINE JOLIVET Emilien

Je fais partie du comité d'administration de Cygnus. Je suis le responsable du site internet, et tout ce qu'il induit, ainsi que toute la partie réseau du jeu. J'ai également contribué au développement du parcours, notamment des plateformes et des scripts qui leurs sont propres, c'est-à-dire, les déplacements de celles-ci ou encore leurs propriétés. De plus, tout comme mes collègues, je contribue à la création des niveaux du jeu.

3.4 THOMÉ Aubin

Je suis actuellement responsable de la programmation fonctionnelle au sein de notre projet. Mon rôle est d'harmoniser l'ensemble des scripts de projets et m'assurer que ceux-ci soient capables de communiquer de manière efficace. Je me suis, dans un premier temps, penché sur l'utilisation de l'héritage et de la composition au sein de notre projet. En parallèle, je suis aussi responsable de la création des différents parcours de chaque niveau.

Troisième partie

Répartition des tâches

L'un de nos membres a quitté le projet, c'est pourquoi nous avons établi une nouvelle répartition des tâches de manière à ce que chacun ait la même quantité de travail. Nous en avons profité pour modifier les anciennes responsabilités après s'être rendu compte du travail à fournir dans certains domaines, nous avons donc décidé de fonctionner par paire de deux étudiants en fonction des aptitudes de travail de chacun, les tâches sont donc divisées en deux groupes ce qui nous permet de progresser plus rapidement. De plus, lors de l'avancement du projet nous avons fait émerger de nouvelles responsabilités que nous n'avions pas imaginées lors du cahier des charges. Nous avons opté pour une répartition des tâches en fonction des préférences et aptitudes de chaque étudiant du groupe.

Responsabilité	Responsable	Suppléant
Programmation fonctionnelle	Aubin	Emilien
Graphisme	Noé	Aubin
IA	Noé	Johan
Git	Emilien	Aubin
Site Web	Emilien	Aubin
Réseau et multijoueur	Emilien	Johan
Administratif	Johan	Noé
Rendus	Aubin	Emilien
Parcours	Aubin	Emilien
Armes et collectables	Johan	Noé
Sons et musiques	Noé	Johan
UI	Johan	Noé

TABLE 1 – Nouvelle répartition des tâches

Quatrième partie

Avancement des tâches et objectifs

4 Global

Ce tableau résume les avancements au global des tâches que l'on détaillera dans la section suivante.

Tâche	Soutenance 1	Soutenance 2	Soutenance 3
Graphisme	0%	20%	100%
IA	30%	55%	100%
Site Web	100%	100%	100%
Réseau et multijoueur	25%	50%	100%
Parcours	10%	60%	100%
Armes et collectables	30%	80%	100%
Sons et musiques	0%	0%	100%
UI	15%	75%	100%

TABLE 2 – Avancement des tâches

5 Par tâches

5.1 Joueur

Notre joueur prend l'apparence d'un robot. Il est référencé sous forme d'un objet appelé "GlobalPlayer". Cet objet est équipé d'une caméra qui le suit dans ses déplacements. De plus, nous avons doté l'objet d'une interface utilisateur, permettant à ce dernier de voir la vie, l'oxygène et l'inventaire. Celui-ci est donc aussi équipé d'un objet inventaire, dans lequel il pourra stocker armes et collectables. L'objet est aussi équipé d'un objet "Player", l'apparence du joueur, implémentant les scripts suivants :

- **Script mouvement** : Ce script est celui qui permet au joueur de se déplacer dans son environnement : il peut ainsi sauter, rouler, courir et sprinter.
- **Script caméra** : Ce script permet simplement à la caméra de suivre les déplacements du joueur de manière à offrir un gameplay agréable.
- **Script de visée** : Ce script permet d'avoir accès au mode visée ainsi que d'adapter la caméra à ce mode tout en continuant de suivre les déplacements du joueur.

- **Script de gestion des propriétés du joueur** : Ce script gère l'oxygène et la vie du joueur : l'oxygène descend linéairement en fonction du temps. La vie descend selon les interactions avec les ennemis. Les deux propriétés peuvent remonter grâce aux consommables trouvés par le joueur.

Le "GlobalPlayer" est l'objet qui est généré et utilisé dans le mode multijoueur. L'objectif pour la prochaine soutenance concernant le joueur sera de réussir à implémenter l'attaque à distance.



FIGURE 1 – Prefab du Player

5.2 Armes et collectables

C'est, pour le moment, la partie sur laquelle nous sommes les plus avancés en dehors du site web. La collecte d'armes et d'autres objets est essentielle au jeu. En effet, c'est avec ces items que nous pourrions terminer les niveaux. Pour l'instant, il existe un unique type d'arme et 3 potions différentes :

- **L'épée** : disponible sous plusieurs coloris caractéristiques des capacités qui l'accompagnent, elle est la première arme collectée par le joueur et lui permettra de se battre en combat rapproché.



FIGURE 2 – Prefab de l'épée

- **La potion de gain de vie** : comme son nom l'indique, cette potion permettra au joueur qui l'a bue de gagner quelques points de vie, et ainsi de pouvoir progresser plus loin dans le niveau. L'effet de cette potion est instantané.
- **La potion d'oxygène** : Ici aussi, le nom est explicite. Le joueur qui boit cette potion pourra recharger sa jauge d'oxygène de quelques unités. L'effet de cette potion est instantané.
- **La potion d'invincibilité** : La dernière potion laisse, elle aussi, présumer ses effets. Une fois consommée, le joueur ayant bu cette potion est invincible pendant une faible durée. C'est alors le moment idéal pour s'attaquer au boss. L'effet de cette potion est actif sur une plage de temps définie.



(a) Potion de vie



(b) Potion d'oxygène



(c) Potion d'invincibilité

FIGURE 3 – Ensemble des potions

Tous ces items sont référencés en tant que Scriptable Objects dans notre projet, de manière à ne pas créer trop de scripts qui auraient beaucoup de points communs et à ne pas avoir àinstancier à chaque fois des objets items. L'objectif pour la prochaine soutenance est l'implémentation de l'arme à distance ainsi que celle des projectiles.

5.3 Combats

Dans notre jeu, le premier type de combat disponible est le combat rapproché. Le joueur peut équiper une épée et déclencher une animation de combat rapproché lorsqu'il souhaite attaquer. Pendant cette animation, les dommages de l'épée actuelle sont appliqués aux ennemis touchés. Pour cela nous allons utiliser le Raycast, on lui rentre en paramètre la position du joueur et la direction dans laquelle il regarde, on pourra ainsi tester si l'ennemi est dans son champ de vision et si il est assez proche du joueur pour pouvoir l'attaquer.

L'objectif pour la prochaine soutenance sera de mettre en place l'attaque à distance, ce qui comprend les dommages que l'arme met à son ennemi mais aussi les animations de cette attaque.

5.4 Ennemis

Les ennemis adopteront une apparence différente en fonction de l'environnement dans lequel ils se trouvent. De plus, il y aura diverses attaques qui seront actives à partir d'un moment spécifique du niveau ou d'un état d'une des entités. Les ennemis pourront uniquement nous attaquer en mêlée au début de notre jeu et au fur et à mesure du temps pourrons développer d'autres attaques, comme par exemple, celle à distance. Chaque ennemi combat sur une plateforme qui est délimitée par son Nav-Mesh Surface pour éviter qu'il ne puisse sortir de l'arène et qu'il puisse trouver le meilleur chemin entre lui et le joueur en contournant de potentiels obstacles.

Nous définissons dans une classe Enemy les points de vie et le rayon à partir duquel il nous attaque, cela va varier en fonction des types d'ennemis. En effet les boss intermédiaires ne seront pas très difficile à combattre comparé aux boss de chaque fin de niveaux. Chaque ennemi va prendre en argument une liste contenant chaque les caractéristiques de chaque attaque telles que leurs noms, leurs animations et leurs points de dommages.

Un ennemi IA va implémenter à la fois la classe Enemy et la classe AiMovement contrairement à un ennemi statique qui va seulement implémenter la class Enemy. La principale difficulté lors de la création de ces ennemis était d'organiser correctement la hiérarchisation de manière à adapter les ennemis à nos besoins de manière efficace.

L'objectif pour la prochaine soutenance concernant l'IA sera de mettre en place le combat à distance de manière à ce que l'animation de tir soit en cohérence avec les mouvements de l'IA, ainsi qu'adapter les spécificités de l'IA à tous les autres boss des niveaux suivants.

5.5 Parcours

5.5.1 Plateformes

Nous avons créé des prefabs de plateformes qui nous permettent de les réutiliser à différents endroits. Nous avons aussi développé des scripts pour assigner différents comportements à celles-ci. Par exemple, nous avons des plateformes qui se déplacent

de haut en bas et/ou de gauche à droite en suivant des “waypoints” (le script peut suivre autant de waypoints que nécessaire).

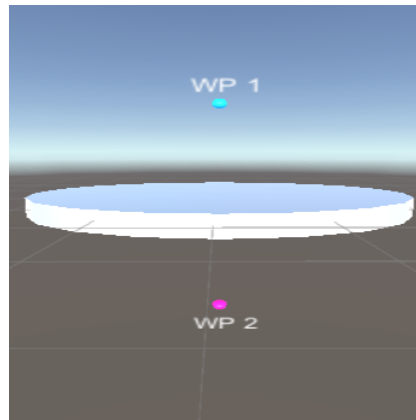


FIGURE 4 – Plateforme avec deux waypoints haut/bas

Nous avons aussi fait en sorte que les joueurs restent “collés” aux plateformes lorsqu’elles se déplacent. Nous avons également élaboré des plateformes qui disparaissent lorsqu’un joueur reste dessus pendant plus de n secondes et qui réapparaissent ensuite.

Nous sommes actuellement en train de développer une plateforme qui permet de rebondir dessus, une sorte de “trampoline” mais nous rencontrons au moment de l’écriture du rapport des difficultés pour adapter la physique de cette plateforme avec celle de l’asset de notre joueur.

Étant donné que nous projetons d’avoir plusieurs niveaux avec plusieurs biomes différents et donc différents environnements spécifiques, nous prévoyons d’ici la prochaine soutenance de développer encore plus de comportements de plateformes adaptés à ces environnements comme par exemple une plateforme glissante pour le biome glace.

5.5.2 Terrain de spawn

Le terrain de spawn est l’endroit où les joueurs arrivent pour la première fois. Il est en quelque sorte la plateforme de départ. Contrairement aux autres plateformes qui sont des objets 3D classiques, ce terrain est un objet spécifique lui permettant d’être sculpté pour donner un terrain non régulier réalisable depuis Unity et en temps réel. Il peut être peint de plusieurs textures mais pour le premier niveau la texture est uniforme. Nous avons ainsi utilisé les outils disponibles de ce type pour réaliser le volcan de cette première plateforme.

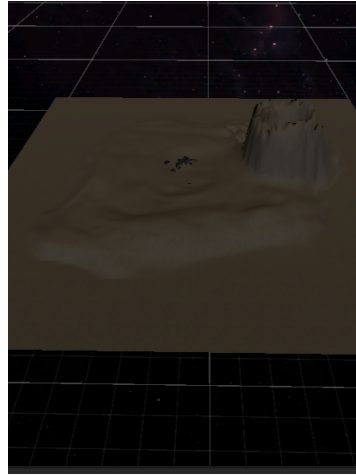


FIGURE 5 – Terrain du premier niveau

5.5.3 Surface de respawn

Ces surfaces sont présentes à différents endroits du jeu et font mourir le joueur s'il entre en contact avec elles. Pour le premier niveau elles abondent un matériel de lave.



FIGURE 6 – Terrain du premier niveau avec le lac de lave

5.6 Menus

La partie menus se divise en deux catégories : les menus principaux et le menu de pause. Pour ces menus nous avons utilisé UGUI, l'ancienne manière de créer les UI dans Unity. Ce choix s'explique par un manque de documentation et d'explications sur le nouvel outil nommé UIToolkit.

Le menu pause n'étant pas essentiel dans le développement actuel du jeu, nous

avons donc priorisé les menus principaux dont une grande partie est fonctionnelle. Ils nous permettent de lancer une partie solo, une partie multijoueur, d'accéder à une page des crédits et de quitter le jeu. Le menu des réglages, déjà accessible n'est pas encore fonctionnel et sera l'un des prochains objectifs à réaliser.



FIGURE 7 – Main Menu

5.7 Réseau et multijoueur

Pour le réseau, notre objectif est de rendre le jeu jouable en multijoueur, en coopération à deux comme mentionné précédemment. Pour ce faire, nous avons dû utiliser la librairie Photon. En plus de pouvoir connecter deux joueurs sur le même niveau, il faut que les boss, les caractéristiques spécifiques telles que les “easter eggs” soient synchronisés entre les deux joueurs. Par exemple, si un raccourci n'est prenable qu'une seule fois et qu'il a déjà été emprunté par le premier joueur alors le second ne peut pas y accéder. De même pour la vie des boss.

De plus, notre jeu implique des mécaniques de combat rapproché mais également à distance qui implique la gestion des projectiles en multijoueur, il faudra attendre l'intégration des armes à distance dans le projet pour gérer cette partie ci.

De manière générale, notre avancée sur ce point est plus que satisfaisante. Néanmoins, nous aurions dû nous pencher plus tôt dans le développement du réseau car nous avons été amené à restructurer une partie des objets liés au joueur.

Pour la prochaine soutenance notre objectif est donc de gérer ces combats. En bonus, nous aimerions implémenter le fait de pouvoir créer des parties privées pour

pouvoir jouer entre amis et non pas avec un joueur inconnu.

5.8 Graphismes et animations

Pour la partie **Graphismes**, nous avons posé les bases des visuels de notre jeu pour nous permettre d'avancer au mieux sur les autres tâches :

- **Joueur** : Nous avons les graphismes du personnage qu'incarne le joueur, un robot.
- **Premier niveau** : Une partie des graphismes du premier niveau, dont le thème tourne autour de la lave, sont déjà en voie de développement.
- **Armes et collectables** : Nous avons aussi les graphismes des armes de mêlée pour chacun des niveaux même si seule celle du premier niveau est pour le moment utilisée ainsi que certains collectables comme les potions.

L'ensemble de ces graphismes nous proviennent du Unity Asset Store.

Pour la partie **Animations**, nous avons réussi à mettre en place différentes animations :

- **Joueur** : toutes les animations qui concernent les mouvements :
 - **Le saut**, qui est accessible depuis les états de course et d'arrêt.
 - **La chute**, qui est provoquée après un saut ou quand le joueur tombe d'une plateforme.
 - **L'arrêt**, qui est l'état de base. Il permet d'accéder à tous les autres états du joueur.
 - **La course**, qui est accessible depuis l'état d'arrêt, la chute et après une roulade.
 - **La roulade**, qui est accessible depuis l'état d'arrêt et celui de course.
 - **L'attaque mêlée**, qui est accessible depuis n'importe quel état et renvoie à cet état initial.
 - **La mort du joueur**, qui est elle aussi accessible depuis n'importe quel état et renvoie à l'état arrêté au respawn du joueur.

Les animations ont toutes été trouvées et importées avec le Unity Asset Store. Les liaisons entre les différentes animations ainsi qu'entre les animations et leur script ont été les plus compliqués à implanter dans cette partie du projet.

- **UI** : des animations réalisées à la main ont permis d'animer l'arrière-plan de nos menus principaux.

- **Lave** : la lave est animée pour la rendre plus réaliste.

L'objectif pour la prochaine soutenance est d'implémenter toutes les animations liées à l'attaque à distance, mais aussi de régler tous les petits défauts de liaisons entre les différentes animations. Pour les graphismes, nous devons trouver les apparences pour les différentes IA ainsi que les designs des objets et de l'environnement des prochains niveaux.

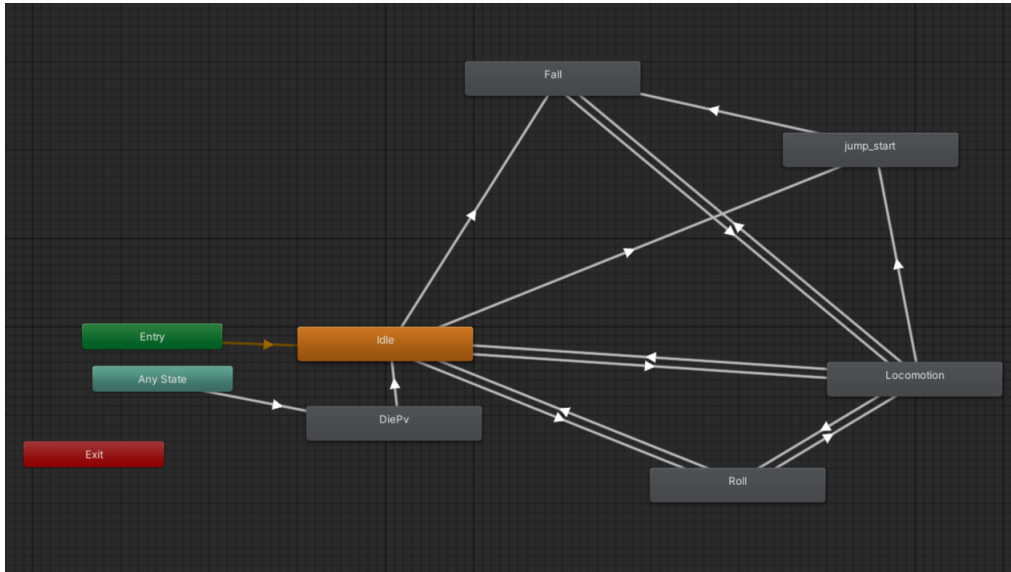


FIGURE 8 – Animateur principal du joueur

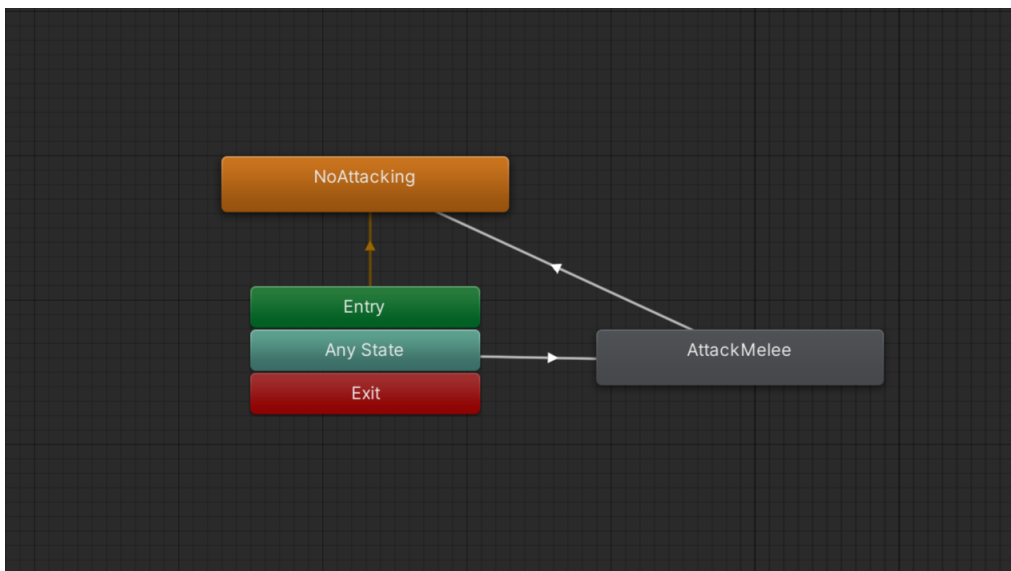


FIGURE 9 – Animateur du joueur pour les attaques

5.9 Site web

Le site Internet a été entièrement développé par nos soins. Nous l'avons développé en ReactJS. Le site est accessible à l'adresse suivante : <https://galaxytraveller.fr/> depuis Janvier 2023.

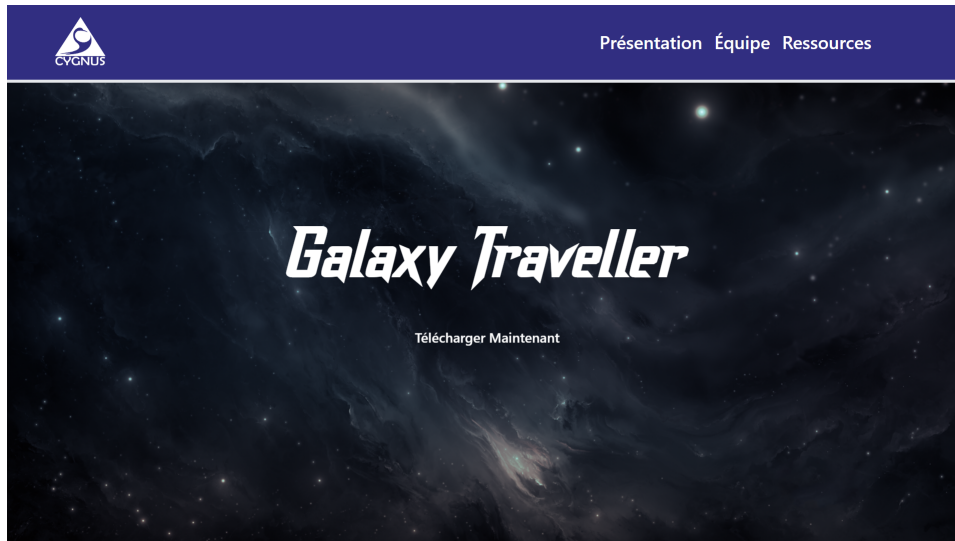


FIGURE 10 – Site du jeu

Sur ce site, vous pouvez actuellement trouver une rapide présentation de l'équipe *Cygnus* et du projet *Galaxy Traveller*, télécharger des documents liés à notre projet et trouver les sources qui nous ont été utiles. Le site est responsive sur téléphone. Nous n'avons pas connu de soucis pour développer le site internet car nos membres en avaient déjà réalisés par le passé.

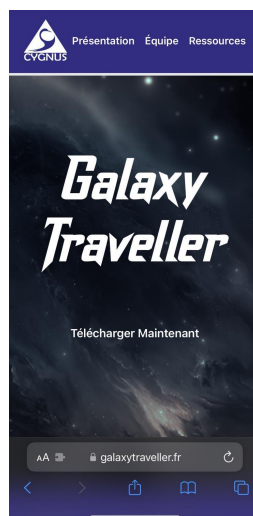


FIGURE 11 – Responsive du site

Cinquième partie

Conclusion

Pour conclure, notre entreprise est dans les temps concernant l'avancée de notre jeu *Galaxy Traveller*. Celui-ci verra le jour dans les temps impartis. Chacun avance la partie dont il est responsable tout en aidant les autres pour une avancée homogène. Pour la prochaine soutenance, il nous reste, dans les grandes lignes, à finir l'implémentation de l'IA, à créer les niveaux 2 et 3, et mettre en place le système de combat à distance. Autrement dit, nous entrons dans la dernière phase de la conception de notre jeu.

L'équipe *Cygnus*.