

|  |  |
| --- | --- |
|  | **Édition 2022**  **DOSSIER DE CANDIDATURE**  **PRÉSENTATION DU PROJET** |

Ce document est l'un des livrables à fournir lors du dépôt de votre projet : 4 pages maximum (hors documentation).

Pour accéder à la liste complète des éléments à fournir, consultez la page [**Préparer votre participation**](https://trophees-nsi.fr/preparer-votre-participation).

Vous avez des questions sur le concours ? Vous souhaitez des informations complémentaires pour déposer un projet ? Contactez-nous à [**info@trophees-nsi.fr**](mailto:info@trophees-nsi.fr).

**NOM DU PROJET : SpaceShooter**

# > PRÉSENTATION GÉNÉRALE :

## Présentation

SpaceShooter est un jeu de survie en vue 2D, le joueur doit y vaincre des monstres tous plus nombreux aux files des vagues. Pour cela deux modes lui sont mis a disposition, un mode Campagne où il devra traverser plusieurs mondes, il y rencontrera des boss et des monstres en tous genres. Ainsi qu’un mode infini dans lequel, il fera face à des vagues toujours plus difficiles, dans un but d’atteindre des vagues toujours plus imposantes.

Alors ces parties le joueur accumule des pièces d’or lui permettant d’acquérir des armes toujours plus fort, mais aussi de nouveaux personnages. Ces personnages pourront eux aussi être améliorés à l'aide de ces pièces. Elle permet aussi l’achat d’équipement qui vise à faciliter la progression du joueur, comme un bouclier, des grenades, ou des potions.

## Idée et objectifs

Le projet a été débutez dans un but de progression en code, plus particulièrement en python, plus particulièrement en programmation orienté objet. Le projet devait donc contenir de nombreuses class qui interagissent en elle-même.

Un jeu était parfaitement adapté à cela, de plus le jeu rajouter une difficulté, le nombre d'interactions qui est fortement élevé. L’idée était aussi de trouvée un projet qui nous donne envie, le développement d’un jeu répondait donc parfaitement à nos attentes.

## Intérêts et origine du projet

Comme dit ci-dessus l’intérêt du projet était de progresser en python, découvrir de nouvelles choses comme la gestion des fichiers, la gestion des événements, la sauvegarde de données dans d’autres fichiers. Tout en prenant du plaisir à développer.

Dans une recherche de type de jeux, nous avons fait le choix de rendre un ancien petit projet jeu texte, en graphique. Tout était donc prêt il ne restait plus qu’a commencé.

# > ORGANISATION DU TRAVAIL :

## Présentation de l’équipe

Heilmann Jonathan : Programmation du projet, crée les différente fonctionnalité gère le bon fonctionnement du jeu, intégré les éléments apporter par le reste de l’équipe.

Schmitt Nicolas : Graphismes et équilibrage du projet, crée et recherche les décor du jeu, comme les personnages, les armes, les équipements… Equilibre le jeu, en définançant le nombre de dégâts du joueur, la vie du monstre, le nombres de monstres par vague…

## Organisation du travail

L’équipe travaille sur le projet quotidiennement, la communication se fait en partie au lycée. Mais aussi à travers certains logicielle, tel que :

* Discord : Pour communiquer rapidement et envoyer des petits fichier
* Trello : Pour répartir les tache. Noté les idée et voir l’avancée du projet. Notre Trello : <https://trello.com/b/uS65OQof/id%C3%A9e-a-faire>
* Git et GitHub : Partager le code, suivre l’avancée, GitHub nous a permis de garder une structure commune pour tous, et de rapidement partager ou récupéré l’avancement.

GitHub du Projet : <https://github.com/Miniquoinquoin/SpaceShooter>

* Photo pea : Création des modèles graphique

# > LES ÉTAPES DU PROJET :

## Planification

Structuration et recherches des idées, création d’une structure des fichiers/ documents. Test de première interaction entre Object python.

## Développement du cœur du jeu

Création de la répétition du fond, des déplacements et animation du personnage. Ajout des monstres, création d’effet avec les sorciers, ajout des projectiles avec l’armes du joueur et les monstres qui tire.

## Ajout des Menus

Implémentation de nouvelle fonctionnalité tel que de nouveaux mondes, de nouveau personnages, de nouveau mode. Création de Menu pour simplifier et améliorer l’expérience de Jeu.

## Sauvegarde de l’avancement

Sauvegarde des informations (pièces d’or, personnages, son, avancer dans la campagne…) dans différent fichier texte et csv.

## Fonctionnalité tierce

La possibilité d’améliorer son personnage, ses armes, d’acquérir et d’améliorer des équipement (boucliez, grenade, potion). L’ajout d’une carte pour rendre le mode campagne, de son et de musique pour améliorer l’immersion du joueur.

Les étapes détailler de l’avancement du projet son disponible sur GitHub !

# > FONCTIONNEMENT ET OPÉRATIONNALITÉ :

## Avancement et approches du Projet

Le jeu est jouable, on peut y faire beaucoup de choses, on y retrouve de nombreux aspects qui permettent d’améliorer la diversité du jeu.

Le jeu est actuellement en phases de test, par plusieurs personnes l’objectif est de détecter et corriger un maximaux de bug.

Le jeu est à première vue terminer, il ne reste rien d’élémentaire à ajouter. L’ajout de nouvelle fonctionnalité que nous verrons plus tard pourraient être envisagés.

## Difficultés rencontrées et solution apportées

### Faire marcher le Joueur :

On n’y pense pas forcément tout de suite, mais simplement déplacer le personnage ne suffit pas. Il faut aussi lui données un effet de marche, pour cela nous nous sommes inspirés d’autre jeu vidéo, et avons utilisé plusieurs images qui affichent de manière successive font marcher le joueur.

### Saut est trajectoire de lancer :

La modélisation d’un saut, est particulièrement complexe, en effet la vitesse y du joueur doit varier en fonction de l’avancer du saut met aussi de la gravité. Une simple formule de mathématique avait permis de résoudre le problème du saut. Mais celle-ci ne pouvait pas être utilisée pour les lancées qui devaient aussi dépendre de x, nous avons donc modélisé le lancer par un ovale et utilisé le cercle trigonométrique.

### Sauvegarder des informations à long terme :

Sauvegarder des informations dans python c’est facile grâce aux variables, mais comment faire pour ne pas tout perdre lorsqu’on stop le programme. Pour régler ce problème nous avons choisit d’utiliser des fichiers csv, qui peuvent stoker une grande quantité de donnée séparer par un caractère définit comme le point-virgule et forme ainsi un tableau.

### Création des animations :

Crée une animation avec EZ, c’est plus ou moins facile tant qu’on utilise uniquement des formes directement traçables avec une fonction EZ. Mais quand l’animation demande des formes plus complexes telles que nos coins rouges pour illustrer un coup. L’utilisation d’outiller mathématique devient alors nécessaire, dans le cas de notre animation de dégâts nous avons utilisé le cercle trigonométrique pour calculer l’ensemble des points présents sur notre ovale dont la taille varie à chaque instant.

# > OUVERTURE :

## Idées d’amélioration

### Crée des bordures variées en fonction des mondes :

Les bordures sont représentées par des caisses empiler que vous soyez sur Terre, sur Mars ou sur Forestia. L’idée serait de créer une bordure propre à chaque monde, afin de la rendre plus esthétique.

### Crée des effets dans les mondes :

Vous avez probablement remarqué que la principale différence entre les mondes hormis le décor, ce sont les monstres. L’idée serait d’accentuer ces différences en créant des particularités à chaque monde, cela peut être la gravité, du vent (pousse aléatoirement le joueur dans une direction), des zones empoisonner …

Plus d’idées d’amélioration sont disponible sur notre Trello.

## Stratégie de diffusion

Nous n’avions pas prévu de rendre notre jeu célèbre, étant donné que le but était de progresser en python, ainsi quand logicielle dessin.

Si nous avions voulu diffuser notre jeu, nous aurions probablement organisé un concours sur les réseaux sociaux, ou le joueur qui atteindrait la plus haute vague emporterait cash price. Mais pour cela nous devrions crypter, ou délocalisez-les donner pour éviter la modification de certains éléments comme les nombres d'or.

## Analyse critique du résultat

Nous trouvons le résultat convainquant, mais si c’était refaire :

Heilmann Jonathan : J’organiserait mieux les class fondamentale des menus, ainsi que les déplacements du joueur différemment pour permettre un mode multi-joueur local.

Schmitt Nicolas : J’essaierais de crée plus de model moi-même, plutôt que de partir de model libre sur google.

**DOCUMENTATION**

*• Spécifications fonctionnelles (guide d’utilisation, déroulé des étapes d’exécution, description des fonctionnalités et des paramètres)*

*• Spécifications techniques (architecture, langages et bibliothèques utilisés, matériel, choix techniques, format de stockage des données, etc)*

*• Illustrations, captures d’écran, etc*

La documentation se trouve dans **doc.html** ou **doc.md**