



nom de votre projet :	Spaceship Attack
membres de l'équipe :	Diego – APARICIO
membres de l'équipe :	Thomas - BARSEGHIAN
membres de l'équipe :	Valentin - ZOU
membres de l'équipe :	Esthébane - BOILLEAU
Niveau d'étude :	Terminale

établissement scolaire :	Lycée Kleber
enseignante/enseignant de NSI :	Christophe - CAIRE

> PRÉSENTATION GÉNÉRALE :

Pouvez-vous présenter en quelques mots votre projet ?

Notre projet est inspiré du jeu d'arcade Space Invader. Dans notre jeu, nous incarnons un personnage perdu au fin fond de l'espace dont le but est de retrouver son chemin jusqu'à la Terre. Pour ce faire nous devons vaincre les vagues d'ennemis aliens qui tentent de nous capturer, survivre aux pluies d'astéroïdes et pourfendre les boss aliens dans des combats spatiaux certes en deux dimensions mais acharnés tout de même. Au fur et à mesure que le joueur progresse dans le jeu, il a la possibilité d'améliorer certains paramètres de son vaisseau tels que les dégâts, la vie, la cadence de tir, mais également la probabilité de faire apparaître du soin.

Comment est né ce projet ? Quelle était la problématique de départ ?

Au départ, nous nous sommes mis d'accord pour que notre projet soit un jeu. Puis, nous avons eu l'idée de s'inspirer du jeu emblématique Space Invader. Dès lors, le problème était de coder le jeu car seul un d'entre nous était familier avec Pygame.

Quels sont les objectifs ? À quels besoins répondez-vous ?

Nos objectifs étaient de créer un jeu divertissant rappelant les vieux jeux d'arcades mais se renouvelant dans la façon de jouer. Mais aussi d'apprendre et d'approfondir nos connaissances en python et de constater puis dépasser nos limites.

> ORGANISATION DU TRAVAIL :

Pouvez-vous présenter chaque membre de l'équipe et préciser son rôle dans ce projet ?

Diego (spécialité NSI et maths) s'est occupé de la structure du jeu, de son fonctionnement, de la définition des classes principales, des ennemis, des boss, des attaques, des sons, et du remplissage des formalités de participation au concours.

Valentin (spécialité NSI et maths) s'est occupé du menu principal, du menu de pause, du menu d'améliorations, du menu de développeur, du menu de fin, des boutons et du texte.

Thomas (spécialité NSI et maths) s'est occupé des vaisseaux du joueur, des objets, de la pluie de météorites, et de l'équilibrage du jeu.

Esthébane (spécialité NSI et SES) s'est occupé de l'affichage tête haute, de tous les sprites et de toutes les animations.

Comment avez-vous réparti les tâches et pourquoi ?

Diego a pris en charge la partie la plus compliquée car il était celui qui était le plus familier avec Pygame.

Valentin s'est occupé des menus, des interactions et de la présentation de l'interface car c'est ce qu'il trouvait le plus intéressant et aime s'occuper du côté visuel du jeu.

Thomas s'est beaucoup occupé du gameplay car selon lui c'est la partie la plus amusante.

Esthébane a pris en charge l'implémentation dans le code de tout le côté graphique et design car il est passionné de dessin et de pixel art.

Combien de temps avez-vous passé sur le projet ? Avez-vous travaillé en dehors de l'établissement scolaire ?

Nous avons démarré le projet le 14 février et nous y avons passé plus de 180 heures. Près de 90% du développement du projet a été fait en dehors de l'établissement scolaire car nos emplois du temps ne le permettaient pas.

Quels sont les outils et/ou les logiciels utilisés pour la communication et le partage du code ?

Nous avons mis en place un répertoire GitHub pour le partage du code et avons créé un groupe sur Discord pour pouvoir communiquer et mieux travailler ensemble à distance. Par ailleurs, nous avons utilisé Photoshop, Pixilart, et Paint3D pour créer les images des sprites (sauf pour certaines images que nous avons récupéré d'une librairie gratuite libre de droit en ligne).

Vous veillerez au bon équilibre des différentes tâches dans le groupe. Chaque membre de l'équipe doit impérativement réaliser un aspect technique du projet (hors design, gestion de projet).

LES ÉTAPES DU PROJET :

Présenter les différentes étapes du projet (de l'idée jusqu'à la finalisation du projet)

- Décision de créer un jeu inspiré de Space Invader.
- Pose des bases du jeu Pygame.
- Création d'un joueur, d'un ennemi et de projectiles.
- Ajout du déplacement du joueur, des ennemis et des projectiles, et ajout de la collision.
- Création un système de score et un écran de défaite.
- Ajout de la possibilité de rejouer sans relancer le code.
- Création un système de niveaux et d'expérience.
- Ajout d'une image de fond qui défile.

- Création de vagues aléatoires et sauvegarde du meilleur score (tant que l'on ne quitte pas).
- Création d'objets qui apparaissent aléatoirement à la mort des ennemis.
- Création d'un menu d'améliorations et d'un système d'améliorations.
- Ajout d'animations.
- Ajout d'attaques supplémentaires.
- Ajout de plus d'ennemis.
- Création d'un système de progressions de vagues de plus en plus compliquées.
- Ajout de sons.
- Création de l'ATH (affichage tête haute).
- Ajout d'un boss.
- Ajout du changement de mode de tir lors du combat de boss.
- Ajout d'une pluie de météorites entre chaque vague.
- Création d'un menu de pause.
- Changement de collision rectangulaire à collision de masks (au pixel près).
- Création du menu de démarrage et du menu de développeur (pour tester le jeu plus facilement).
- Ajout d'un deuxième vaisseau pour le joueur et d'autres objets.
- Amélioration de la pluie de météorites.
- Amélioration des boss
- Ajout d'animations pour le vaisseau du joueur.
- Décision du nom final du jeu
- Ajout de la fin du jeu
- Correction des derniers bugs

> FONCTIONNEMENT ET OPÉRATIONNALITÉ :

Pouvez-vous présenter l'état d'avancement du projet au moment du dépôt ? (ce qui est terminé, en cours de réalisation, reste à faire)

Actuellement, le jeu comporte 8 ennemis, 2 boss, 6 objets, 2 vaisseaux de joueur, 12 attaques, 6 menus, 112 images d'animations, 60 images, 17 sons et des pluies de météorites. Ce qui fait que le joueur doit compléter 10 vagues avant de gagner. Nous sommes toujours en train d'ajouter plus d'ennemis, de sons et de corriger certains problèmes. Par ailleurs nous essayons toujours de régler le problème principal de notre jeu : l'alignement des projectiles en fonction de l'angle de rotation du joueur par rapport à la souris. De plus, nous comptons implémenter un mode multijoueur dans le futur.

Quelles approches avez-vous mis en œuvre pour vérifier l'absence de bugs et garantir une facilité d'utilisation de votre projet ?

Afin de trouver et corriger les bugs de manière plus efficace, nous avons fait tester notre jeu par plusieurs camarades qui n'ont pas la spécialité NSI. Pour ce faire nous avons dû apprendre à compiler le code python avec les dépendances en fichier exécutable (.exe) grâce au module auto-py-to-exe. Évidemment, nous avons testé notre propre jeu pour vérifier l'absence de bugs notamment grâce au

menu développeur qui nous permettait de tester beaucoup plus vite tous les aspects du jeu. Cependant, le fait d'avoir des avis extérieurs nous a permis de trouver plus de bugs plus facilement et d'avoir plus d'idées à ajouter.

Quelles sont les difficultés rencontrées et les solutions apportées ?

Les premières difficultés que nous avons rencontrées ont été la distribution du code car malgré le fait que le code et les dépendances étaient publiés dans un répertoire GitHub accessible par tous les membres de notre groupe, il fallait que les fichiers soient situés exactement au même endroit pour que le code fonctionne. C'est le module `os` et ses méthodes `chdir()` et `getcwd()` qui nous ont permis de résoudre ce problème.

Le second problème était l'ajout de sons dans le jeu. Grâce à la documentation Pygame, nous avons vite trouvé la classe `Sound()` et sa méthode `play()`. Or il nous a fallu un long moment pour comprendre pourquoi un seul son pouvait être produit à la fois. Il se trouve que Pygame utilise des canaux sur lesquels le son est produit et nous avons dû utiliser la classe `Channel()` de Pygame et ses méthodes `find_channel()`, `play()`, mais aussi la fonction `set_num_channels()` pour définir le nombre de canaux et trouver des canaux libres pour y produire le son.

Le troisième problème était la rotation des sprites car même si la fonction `rotate()` de Pygame est simple d'utilisation. Il se trouve que lorsqu'on tourne un sprite, sa zone rectangulaire change et puisqu'on affiche le sprite à partir du centre de son rectangle le sprite « bouge ». Ce problème nous a demandé beaucoup de temps et de recherche pour comprendre son origine. Or, la solution est très simple, il suffisait de sauvegarder l'ancien rectangle, puis de remplacer le rectangle actuel par le nouveau et enfin, affecter le centre du nouveau rectangle à celui de l'ancien.

Mis à part les problèmes techniques une des principales difficultés a été le fait que 3 d'entre nous sont partis de zéro en termes de connaissance du module Pygame. Mais en s'entraidant nous avons pu pallier ce problème.

> OUVERTURE :

Quelles sont les nouvelles fonctionnalités à moyen terme ? Avez-vous des idées d'amélioration de votre projet ?

Comme dit précédemment, nous comptons ajouter plus d'ennemis, de sons, un mode multijoueur, etc., et corriger certains problèmes.

Pourriez-vous apporter une analyse critique de votre projet ? Si c'était à refaire, que changeriez-vous dans votre organisation, les fonctionnalités du projet et les choix techniques ?

Nous trouvons que le jeu manque de diversité (notamment au niveau de la musique) et que l'expérience de jeu aurait été mieux si la fenêtre de jeu était plus grande. Si nous devions refaire notre projet nous séparerions le code en plusieurs fichiers dès le départ car nous n'y avons pensé que très tard et au

stade où on en était, diviser le code était trop compliqué. De plus, nous essayerions de mieux nous répartir les tâches.

Quelles compétences/appétences/connaissances avez-vous développé grâce à ce concours ?

Ce concours nous a permis de beaucoup progresser en programmation, d'avoir une idée plus précise de comment fonctionne les jeux et de ce que pourrait ressembler le métier de développeur ainsi que le travail en équipe. Ce projet a été le plus grand que nous ayons fait jusqu'ici. Nous avons tous pris beaucoup de plaisir dans la création de ce projet.

En quoi votre projet favorise-t-il l'inclusion ?

Malheureusement, la démographie de notre classe ne nous a pas permis de favoriser l'inclusion.

Ce document est l'un des livrables à fournir obligatoirement lors du dépôt de votre projet : 4 pages maximum. Le non-respect du modèle fourni peut impacter la notation.

La documentation technique complète est à intégrer dans le dossier technique, dans un répertoire nommé doc.

Pour accéder à la liste complète des éléments à fournir, consultez la page [Comment participer ?](#).

Vous avez des questions sur le concours ? Vous souhaitez des informations complémentaires pour déposer un projet ?

Contactez-nous à info@trophees-nsi.fr ou consultez la page [Foire aux questions](#).