

# **Rapport : PROJET ASTEROÏDE**

**MAVEAU Lucas**

**NIZAMUDDIN Hanif**

**L1 SDN**

**Mr WeinBerg**

## **Introduction**

Au cours de cette première année en SDN, nous avons appris à utiliser le langage de programmation Python. Avec un premier semestre pour apprendre les bases, nous avons amélioré nos capacités dans ce langage au deuxième semestre. Nous devons donc réaliser un projet qui consiste à jouer un jeu dans lequel nous contrôlons un vaisseau qui doit détruire des astéroïdes. Pour cela, on doit faire appel à toutes nos connaissances sur Python pour réussir ce projet.

Avant de commencer le travail, nous avons récapitulé les tâches puis nous les avons divisés entre nous. Nous avons tout de suite remarqué que nous serons obligés de passer par des *classes* pour réaliser le jeu. Effectivement, on a vu qu'il y avait pas mal d'éléments qui avaient leurs propres contraintes tels que la gestion du vaisseau, les missiles envoyés et la trajectoire des astéroïdes.

Alors, nous avons tout d'abord commencé par chercher des images pour représenter les différentes composantes du jeu. Ensuite, une fois que cela a été fait, nous nous sommes chacun occupé d'un problème. Puis, dès que cette tâche était finie, on s'attaquait à un autre problème.

Tout d'abord, nous allons parler de la division du problème en plusieurs sous-problèmes et comment nous nous sommes répartis les tâches. Après, nous montrerons comment ces sous-problèmes ont été résolus et les décisions qu'on a dû prendre pour aboutir au résultat final. Enfin, nous concluons ce rapport en exprimant ce que la réalisation du projet nous a apporté.

## **I) Diviser le problème : un bon moyen de comprendre les tâches à réaliser**

### **a) Pourquoi diviser le problème ?**

Ici, on est confronté à un problème assez vaste sans trop de consignes à suivre. On est donc obligé créer nous même un sujet permettant de réaliser ce projet. Il nous fallait donc tout d'abord imaginer quels sont les principales missions à réussir pour l'aboutissement du projet. Nous nous sommes donc appuyés sur le principe même de l'algorithmie qui consiste à décortiquer un grand problème en plusieurs sous problèmes pour nous faciliter le chemin menant à la réalisation du travail.

Cependant, avant de diviser le problème, nous avons d'abord lister les points sur lesquels nous devons travailler. On a donc décider de travailler principalement sur le vaisseau, les astéroïdes et les missiles lancés par notre vaisseau. Tout à coup, nous avons remarqué qu'il était plus simple pour nous de gérer les différentes problématiques du projet. Pour le vaisseau on a vu que le déplacement de celui-ci comportait plusieurs contraintes comme l'accélération ou la rotation. En ce qui concerne les astéroïdes, on a remarqué que leur mouvement devait être aléatoire sur l'écran et que lors d'une attaque subie, ils devaient se diviser et suivre par la suite une trajectoire aléatoire. Puis, pour les missiles, Lucas a constaté que ces derniers suivaient la trajectoire sur laquelle le vaisseau était dirigé. On en a donc déduit que le code allait avoir des similarités entre le code des missiles et le code du déplacement du vaisseau.

Du coup, diviser le problème nous a permis de mieux comprendre ce que le prof attend réellement de notre part. Par la suite, nous avons donc pu commencer à nous répartir les différentes tâches pour réaliser le projet.

## ***b) Répartition des tâches***

Ainsi, on est passé à la répartition des tâches. Tout d'abord, nous les avons listés :

- Choisir des images pour les éléments du projet(vaisseau, astéroïde, ...)
- Définir les différentes actions réalisés par les prototypes pour mieux gérer les évènements lors de la création des classes
- Créer les classes : Vaisseau, astéroïde et missiles
- Créer les fonctions réalisant les actions listées dans chaque classe

Une fois notre listing terminé nous avons réparti les tâches :

- **Hanif :**
  - Redimensionner les images pour qu'elles puissent entrer dans l'écran
  - Travailler sur les fonction de déplacement du vaisseau
  - Créer les fonctions dans la classe astéroïde
  - Lister une partie des fonctions de la classe missiles
  - Recherche sur des fonctions pour la collision
  - Rédaction du rapport
- **Lucas :**
  - Finalisation et amélioration de la classe vaisseau

- Amélioration de la classe astéroïde
- Création de la classe missiles
- Correction des erreurs du programme
- Test des programmes
- Validation des programmes (avec Hanif)
- Réalisation des collisions

Premièrement, Hanif s'est chargé de rogner les images qui représentent notre vaisseau, les missiles et les astéroïdes. Ensuite, nous avons décidé de travailler à deux sur la gestion des événements du vaisseau. Ensemble, on a travaillé sur les déplacements et les orientations du vaisseau. Puis, Lucas s'est chargé de finir la classe vaisseau en travaillant sur la vitesse du vaisseau et sur l'amélioration de son déplacement. Pendant ce temps, Hanif a travaillé sur la classe astéroïde en créant les fonctions qui permettraient aux astéroïdes de se déplacer sur l'écran. Lucas a amélioré le programme mais ne l'a pas fini en raison d'un problème d'affichage. Une fois qu'Hanif a listé les fonctions à coder pour les missiles, Lucas a programmé la classe missile. Enfin, après des pistes données par le prof, Hanif a fait des recherches sur la collision et après Lucas a codé pour les réaliser.

## ***II) Résolution du problème***

## **a) Comment on a résolu les sous-problèmes**

Après avoir cibler les différents sous-problèmes, nous avons commencé à les travailler. Voilà comment nous avons procéder.

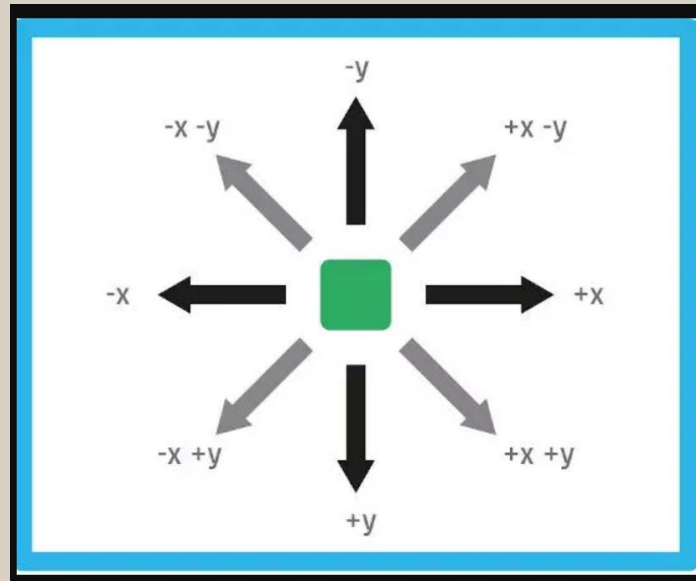
- **Classe vaisseau :**

Nous avons tout d'abord commencé par travailler la rotation du vaisseau. Pour cela, nous avons utilisé les fonctions du TP sur Pygame qui consistaient à faire bouger un carré. Cependant, nous avons eu un problème, nous n'arrivions pas à faire tourner notre vaisseau sur place et on n'avait pas réussi à effectuer une rotation en continu tant que la touche correspondante est appuyée. Pour régler cela, nous avons revu notre code au niveau des conditions KEYUP et KEYDOWN et nous avons pu trouver le problème.

Une fois que cela a été fait, nous avons travailler sur le déplacement dans la direction vers lequel se trouvait le vaisseau. Ici, Lucas a eu l'idée de passer par le cosinus et le sinus après qu'on ait eu la réflexion du triangle rectangle et du coup de Pythagore et de la trigonométrie.

Ensuite, nous avons travaillé la vitesse du vaisseau. On a écrit dans la classe que lorsque le vaisseau avançait en ligne droite, il accélérerait et lorsqu'il prenait un virage pour changer de direction, on a écrit un code pour que le vaisseau décélère afin que le joueur puisse contrôler ses mouvements.

Enfin, on a travaillé sur le moment où le vaisseau sortait de l'écran. Notre but était de le faire apparaître de l'autre côté. Avec l'aide du prof, on a trouvé un schéma avec des coordonnées qui nous a aidés à faire apparaître le vaisseau du côté opposé par lequel il est sorti.



- **Classe astéroïde**

Pour la classe astéroïde, nous avons procédé étape par étape.

Effectivement, on a remarqué que les astéroïdes suivaient une trajectoire aléatoire et apparaissaient n'importe où sur l'écran

Premièrement, nous avons choisi des astéroïdes de grande, moyenne et petite taille. Ce choix a été fait pour répondre à la consigne qui disait que quand un missile touchait un astéroïde, ce dernier se décomposer en 2 autres astéroïdes.

Ensuite, nous voulions les faire apparaître sur l'écran à un certain intervalle de temps. Pour cela, Lucas avait d'abord utilisé la méthode « Timer », cependant, le résultat n'était pas satisfaisant. C'est là que le prof nous a conseillé d'utiliser la méthode « clock » qui permettait de faire apparaître un astéroïde chaque 5 secondes.

Enfin, nous avons également mis le système de passage d'un bout de l'écran à l'autre comme pour le vaisseau pour ne pas qu'il y ait de bug dans le programme au cas où l'astéroïde s'éloignerait trop.

- **Classe missile :**

En ce qui concerne les missiles, c'était la classe qu'on a gardé à fait à la fin car elle prenait quelques contraintes de la classe vaisseau et n'était pas aussi chargé en action comme les deux autres classes.

Premièrement, on a travaillé sur la direction qu'ils suivaient lors de l'appui d'une touche. Une fois la touche appuyée, le missile partait du centre du vaisseau en suivant une trajectoire rectiligne. Pour la trajectoire des missiles, nous avons utilisé le même principe de la trigonométrie qu'avec le vaisseau mais cette fois-ci, c'était plus simple car le missile gardait la même trajectoire.

Par contre, nous ne lui avons pas attribué la faculté de réapparaître à l'autre bout de la fenêtre une fois qu'il l'aurait passé car on a supposé que cela serait un peu trop répétitif en termes de code et n'aurait pas trop servi dans le jeu étant donné qu'on a un nombre de lancer infini.

## ***b) Les tâches non accomplies et les idées abandonnées***

Malgré le fait qu'on ait réussi à programmer la base du jeu, nous avons échoué à le terminer.

L'obstacle qui nous a empêché de finir le jeu était la collision. Nous n'avons pas réussi à écrire un code optimal qui prenait en compte la collision. Le premier script qu'on a écrit pour la collision avait de sérieux problèmes : lenteur d'arrivée des astéroïdes, collision entre missiles et astéroïdes qui ne se produit pas, vaisseau qui passe en dessous des astéroïdes et surtout un arrêt du programme quelques fois.

Nous avons donc opté pour la collision à l'aide des rectangles d'images mais le résultat n'était pas fameux et pas intéressant pour le



projet. Le prof nous avait donc parlé également de la méthode des masques, une méthode que nous avons essayée mais qui demandait trop d'opérations à réaliser. La collision aura donc été la contrainte la plus importante qui nous aura bloqué durant ce projet ce qui nous a aussi empêché de programmer l'évènement de division des astéroïdes lors d'un impact.

Il y a aussi quelques petits détails qu'on n'a pas su réglé, notamment la direction des missiles qui avaient certes la trajectoire dans laquelle ils étaient lancés mais qui étaient dirigés de la même manière à chaque lancer. Pour résoudre ce problème, on a essayé remplacer les missiles par une boule de feu qui avait une même forme peu importe sa trajectoire mais le rognage sur celle-ci n'a pas fonctionné.

En raison du temps perdu pour la réalisation des collisions qu'on n'a malheureusement pas réussi, on n'a pas pu aller au bout de nos idées. Effectivement, dans le listing des tâches, nous avions prévu de mettre un fond de l'espace, un compteur d'astéroïdes détruits par le vaisseau et même un bruitage au moment de l'impact mais la plupart de nos idées bonus étaient liées aux collisions qui se produisent très mal voire pas du tout.

### **c) *Avantages et inconvénients de nos choix***

Les avantages de notre programme sont la bonne structure de notre code. Grâce au listing qu'on a fait au début, on a pu bien séparer les différentes tâches et ne pas se mêler les pinceaux lors de la réalisation des programmes. Aussi, nous avons pu diviser le travail pour ne pas trop surcharger l'un ou l'autre.

L'inconvénient est qu'on a malheureusement essayé de faire trop compliqué. Nous avons essayé de passer par des chemins assez

complexe ce qui a mené à une perte de temps dans la réalisation d'autres tâches et qui a pour conséquence la non finalisation du jeu.

## **Conclusion**

Finalement, ce projet nous aura donc permis de comprendre comment Pygame permettait la création de jeux. On a trouvé que ce travail était un moyen de mettre en œuvre nos capacités sur Python mais aussi de nous tester dans le travail d'équipe.

En ce qui concerne le projet, nous n'avons pas rempli la tâche à 100%. Au niveau des attentes, nous avons seulement pu réaliser la base du projet qui ressemblait totalement à ce qu'on avait imaginé. Malgré certaines attentes remplies, nous n'avons pas réussi à produire la même satisfaction pour chaque mission. Même si nous avons pu produire un résultat à chaque demande, ces derniers n'étaient pas intéressants et ne devaient pas figurer dans le projet.

Au-delà du projet, cette activité aura surtout été une bonne manière de tester notre investissement dans une équipe et surtout notre gestion des tâches. Nous nous sommes divisés les tâches en fonction des capacités de l'un et de l'autre par exemple, Lucas qui est plus performant en code s'est chargé de valider les classes et de les améliorer et Hanif quant à lui a produit les idées, démarrer des classes pour faciliter le travail de Lucas. Du coup, c'était vraiment un projet où les deux membres devaient compléter les faiblesses de l'autre.

## Annexe : trace écrite lors du projet

TACHES A FAIRE

- Classe astéroïde:
  - Déplacement aléatoire
  - Apparition aléatoire
  - Apparition d'un écran à l'autre
  - Dispersion en 2 astéroïdes
- Classe vaisseau:
  - Déplacement : orientation
  - Vitesse de déplacement
  - Passage d'un écran à l'autre
- Classe missiles:
  - Trajectoire de lancer direct
  - Pas de changement d'écran
  - Système de détection que celui du vaisseau Hrc.
- Collision à faire:
  - 3 types d'astéroïdes: Grand, Moyen, Petit.
  - Lors de toucher d'un missile:
    - 0  $\rightarrow$  20  $\rightarrow$  détruit
    - 10  $\rightarrow$  20  $\rightarrow$  détruit
    - 2  $\rightarrow$  détruit

Bonus

- Fond
- Bruitage
- Score
- HP des vaisseaux (si possible)

