



<b>NOM DE VOTRE PROJET :</b>	<b>SISYPHE.IO</b>
<b>MEMBRES DE L'ÉQUIPE :</b>	<b>KILLIAN - MILANI</b>
<b>MEMBRES DE L'ÉQUIPE :</b>	<b>SIMÉON - GILLET</b>
<b>MEMBRES DE L'ÉQUIPE :</b>	<b>KYLIAN - ROUSSEAU</b>
<b>MEMBRES DE L'ÉQUIPE :</b>	<b>ABDELMALIK - MOUSSAOUI</b>
<b>MEMBRES DE L'ÉQUIPE :</b>	<b>TRISTAN - BIBENS</b>
<b>NIVEAU D'ÉTUDE :</b>	<b>TERMINALE</b>
<b>ÉTABLISSEMENT SCOLAIRE :</b>	<b>LYCÉE EDMOND-PERRIER</b>
<b>ENSEIGNANTE/ENSEIGNANT DE NSI :</b>	<b>M. MADRIAS</b>

## > PRÉSENTATION GÉNÉRALE :

Sisyphé.io est un projet qui revisite le mythe antique de Sisyphe pour le transformer en un jeu à énigmes basé sur le concept du Sokoban. Outre les graphismes, l'ambiance sonore, les mécaniques inédites et les 5 mondes proposés à la base ainsi qu'un menu paramètres afin d'optimiser le confort de jeu, son atout principal est un éditeur de niveaux complet et facile d'utilisation. En effet, après avoir complété le tutoriel du jeu (monde 1), vous verrez deux nouveaux boutons se débloquent sur le menu principal : "Éditeur" et "Ouvrir". Le premier exécute un deuxième programme qui permet de créer, ouvrir et sauvegarder vos propres niveaux en mettant à votre disposition toutes les mécaniques du jeu. Le second, couplé à un système de gestion de niveaux sauvegardés dans des matrices au sein de fichiers au format json, permet d'ouvrir n'importe quel niveau compatible situé sur votre machine.

Ce projet est né d'une discussion entre les membres de l'équipe au sujet d'un devoir que nous avons à rendre pour un cours de NSI : nous avons à notre disposition un programme d'une cinquantaine de lignes avec le minimum nécessaire pour lancer un jeu fonctionnel en Tkinter, et carte blanche pour les améliorations dans les limites du programme de terminale. Nous avons alors décidé d'utiliser plusieurs points étudiés durant l'année : fonctions récursives, programmation orientée objet, matrices, modularité et bases de données SQL, afin d'en faire un Sokoban. Mais comment le rendre intéressant et original ? Après nous être donné quelques jours de réflexion chacun de notre côté sur cette problématique, nous avons réussi à arriver à une idée intéressante : Pourquoi ne pas permettre à tout le monde de créer ses niveaux, pour qu'il puisse ensuite les partager avec ses amis ? Ainsi Sisyphé.io est né, et malgré quelques difficultés durant les premiers jours de développement pour savoir quelle démarche suivre, quelles fonctionnalités ajouter, etc. nous nous sommes vite accordés sur l'idée d'un jeu ne demandant ni rapidité, ni réflexes.

L'objectif principal du jeu n'est pas de proposer une histoire complète ou une intrigue à couper le souffle : les mondes mis à votre disposition à la base, même s'ils ont été longuement pensés puis adaptés par nos soins, ne sont là que pour montrer l'étendue de ce qu'il est possible de réaliser avec notre projet. Ce dernier est aussi volontairement pensé pour ne demander aucune expérience préalable dans le gaming afin de ne laisser personne de côté : pas de monstre ni de barre de vie, pas de game over ; juste de la réflexion. Cependant, il est toujours possible de combattre pour le titre du joueur le plus rapide ou pour le record de score dans chaque monde si vous le souhaitez : c'est à cela que servent la minuterie en jeu ainsi que la base de données des scores. Nous pensons qu'il est nécessaire de proposer aux gens des jeux de partage adaptés à tous les publics, et c'est le besoin principal auquel Sisyphé.io essaie de répondre à notre petite échelle.

## > ORGANISATION DU TRAVAIL :

Killian MILANI	Développeur de l'éditeur de niveaux, du système de sauvegarde/lecture des niveaux en json, des paramètres, du module base de données. Compilation du jeu en .exe (incluant chemins d'accès relatifs.)
Siméon GILLET	Développeur de la mécanique spéciale du monde 2, du système de déroulement du jeu, des collisions du jeu. Créateur de niveaux, bêta-testeur.
Kylian ROUSSEAU	Développeur des mécaniques spéciales des mondes 3 à 5, des collisions des mécaniques spéciales, du module des langues. Créateur de niveaux, créateur des pixel-arts.
Abdelmalik MOUSSAOUI	Développeur du site internet, des modules de la musique et du son ainsi que des images, des compteurs en jeu (FPS compris). Créateur de niveaux.
Tristan BIBENS	Développeur de la base de l'éditeur de niveaux, des menus (sauf paramètres), des tutoriels pour chaque monde. Créateur de niveaux, bêta-testeur.

La répartition des tâches s'est organisée de la manière suivante : nous avons besoin de deux bêta-testeurs pour le jeu, d'un développeur principal de l'éditeur de niveaux pour s'occuper de l'implémentation de toutes ses mécaniques ainsi que de la sauvegarde/de l'ouverture de niveaux, d'un développeur principal du jeu de base pour s'occuper des menus, des collisions et du déroulement du jeu, et d'un développeur principal pour les mécaniques. Ensuite, lorsque la création d'un site internet nous est venue à l'esprit, il fallait qu'un membre du groupe s'en charge en priorité. Pour les modules, chacun pouvait intervenir lorsqu'il le souhaitait, et il en allait de même pour les textures, les textes, et tous les détails de ce genre.

En dehors du lycée, le projet a pris à chacun d'entre nous entre cinq et huit heures, soit environ trente-trois heures en tout. Il faut ajouter à cela quatre heures d'avancement en classe et au minimum deux heures de délibération pour un total d'environ quarante heures passées dessus au cours de trois semaines (sans compter toutes les documentations et le temps utilisé pour faire des recherches sur les différents modules.)

En ce qui concerne les logiciels, chacun d'entre nous pouvait développer le jeu depuis son IDE Python de prédilection, mais celui que nous avons retenu pour la démonstration est Thonny car il permet une installation simple et rapide des modules qui ne sont pas présents dans une installation de base de Python (Pillow et Pygame). Pour le développement du site internet, l'application Xampp a été particulièrement utile pour nous afin de simuler un environnement de serveur web en local sur nos machines et par conséquent de vérifier l'intégrité des pages et des liens de redirection lorsqu'il serait publié. La communication autour du projet s'est faite exclusivement via l'application Discord afin de séparer les idées, les questions et les discussions dans des canaux distincts. Pour le partage du code, GitHub a été notre plateforme de prédilection.

## > LES ÉTAPES DU PROJET :

Ne voulant pas seulement reproduire ce qui avait déjà été fait dans le genre du Sokoban, nous avons d'abord eu l'idée d'adapter le mythe de Sisyphe et de nommer le jeu de cette manière, puis de baser toute son esthétique autour de cela, mais ce n'était pas assez. Une idée nous vint alors à l'esprit : pourquoi ne pas permettre aux gens de se retrouver autour de ce jeu ? Le rendre accessible à tous, laisser les utilisateurs créer leurs propres énigmes que leurs amis pourront ensuite tenter de résoudre le plus efficacement possible ? C'est de ces interrogations que sont nés l'éditeur de niveaux ainsi que la gestion des niveaux en matrices au sein de fichiers json pour faciliter le partage de ces derniers.

Ensuite, nous nous sommes occupés de la création et de l'implémentation des mécaniques inédites pour étendre ce qu'il est possible de faire avec le jeu. Dans le même temps, certains d'entre nous ont fait des recherches pour les musiques et les sons du jeu pendant qu'un autre développait le module de traduction, puis nous avons commencé la création des textures. Lorsque le graphiste est revenu vers nous avec les illustrations, nous étions prêts à coder tous les menus du jeu, et son esthétique de manière générale.

Après avoir fini les quelques modules et fonctionnalités restantes, nous avons créé les niveaux puis procédé à une chasse aux bugs avec notre classe de NSI ainsi qu'avec nos proches respectifs afin de compléter le travail des bêta-testeurs. Enfin, nous avons changé tous les chemins de répertoires pour qu'ils soient relatifs puis nous avons compilé le jeu en un exécutable qui crée un fichier paramètres et une base de données dans le répertoire Appdata/local/Sisyphe.io de l'ordinateur.

## > FONCTIONNEMENT ET OPÉRATIONNALITÉ :

À l'heure du dépôt, tout ce qu'il nous tenait à cœur d'inclure dans le projet a été implémenté, peaufiné, testé et approuvé. Nous avons inclus chaque point du programme de NSI que nous voulions : fonctions récursives, programmation orientée objet, matrices, modularité et bases de données SQL, mais nous souhaitions également rajouter des niveaux bonus avec des contraintes de temps, de déplacements ou autre, un scénario plus défini pour le jeu ainsi que des classements globaux pour les meilleurs scores de chaque monde. Ces changements ont cependant été abandonnés pour des séances de debug plus approfondies.

Comme expliqué plus haut, nous avons procédé à une chasse aux bugs avec notre classe de NSI ainsi qu'avec nos proches afin de filtrer ce qui aurait pu échapper aux bêta-testeurs par manque de recul sur le projet. En ce qui concerne sa facilité d'utilisation, la mise à disposition d'une version exécutable permet à n'importe quel ordinateur possédant ou non Python de le lancer. À l'intérieur du jeu, nous avons mis en place de nombreux tutoriels à chaque nouvelle mécanique montrée dans le jeu, ainsi que des paramètres pour permettre aux utilisateurs de personnaliser les contrôles comme ils le souhaitent.

Développer un projet avec autant d'éléments interconnectés, malgré l'aide de GitHub pour la résolution des conflits et l'intégration plus simple des modifications, n'est pas une tâche simple. Cependant, une bonne communication en dehors du lycée et des moments de discussion autour du projet en présentiel nous ont permis de nous répartir les tâches de manière efficace et de nous entraider. Tkinter est également un module parfois limité en ce qui concerne le développement d'interfaces complexes, et nous nous sommes retrouvés par moments à devoir contourner ces restrictions.

## > OUVERTURE :

Pour l'évolution du projet sur le moyen terme, nous aimerions intégrer la possibilité de créer des mondes entiers personnalisés dans l'éditeur de niveaux, puis de charger le dossier complet du monde dans le jeu. Comme dit plus haut, la possibilité d'ajouter un scénario complet au jeu est également une amélioration que nous aimerions mener à bien, ainsi que l'ajout de niveaux bonus après la fin du dernier monde, comme un "ultime challenge" qui rendrait la fin du jeu plus attrayante. Pour finir, nous voulions également ajouter un moyen de sauvegarder son meilleur score pour chaque monde dans une base de données afin de pouvoir se comparer aux autres utilisateurs, mais cela n'a pas pu être finalisé dans la limite du temps imparti.

En prenant du recul sur la réalisation du projet, la façon que nous avons eu de développer les différents aspects du jeu lui a donné un aspect assez minimaliste. C'est en même temps un atout et un handicap pour nous, car cela rend la familiarisation avec l'environnement plus simple, et rend par la même occasion l'objectif que nous nous étions fixé, le partage, plus facile d'accès, mais lui donne aussi un côté assez ennuyeux de prime abord. De plus, la façon de gérer les collisions n'est pas la plus optimale, et certaines restrictions que nous avons dû appliquer à l'éditeur auraient pu être évitées en adoptant une approche différente sur son développement (e.g : pouvoir ajouter plus d'une paire de portails par niveaux.)

Si cela était à refaire, notre organisation ne serait pas à changer, car malgré une première semaine assez difficile, nous avons assez vite mis en place une liste de bonnes pratiques lors de l'implémentation de nouvelles fonctionnalités qui nous a permis d'économiser un temps précieux. Cependant, nous changerions la façon de diviser le programme principal pour le scinder en davantage de modules, car sa longueur rendait la résolution des conflits assez fastidieuse lorsque nous mettions de grosses mises-à-jour en ligne. Si nous pouvions réaliser ce projet à nouveau en dehors du cadre d'un devoir de NSI, il serait peut-être plus judicieux d'utiliser un module prévu pour la création de jeu comme Pygame pour l'interface plutôt que Tkinter qui, même s'il met à disposition de nombreuses fonctions et méthodes assez complètes, reste assez limité par endroits.

En ce qui concerne le développement de nouvelles compétences, chacun de nous a énormément évolué dans la communication et le travail en groupe : cette expérience nous a permis de mettre en place des bonnes pratiques qui nous serviront dans l'enseignement supérieur, et de graver dans nos esprits certaines réalités de l'informatique. D'abord, il est normal de ne pas réussir ce que l'on veut faire du premier coup, et même si cela est frustrant, on trouve toujours une solution adaptée. Ensuite, il est normal d'avoir besoin d'aide, surtout lorsqu'on manipule un module nouveau qu'un autre membre du groupe connaît déjà. Ces deux points ne sont que des exemples de ce dont ce projet nous a permis de prendre conscience au fil de réalisation.

Enfin, nous avons pensé ce projet dans le but de permettre à tout le monde de se retrouver pour quelques minutes autour d'un jeu calme et reposant, de mettre au défi ses proches sans avoir besoin de compétence particulière en gaming ou en informatique. Nous pensons qu'il existe de nos jours malheureusement trop de jeux qui mettent à l'écart les joueurs occasionnels, et même si Sisyphé.io n'est pas le jeu le plus complet, ni celui qui possède la plus grande capacité de rayonnement, loin de là, il nous tenait à cœur de mener le projet à bien en gardant ce but en tête. Cela favorise donc l'inclusion selon nous.