



</>
LES TROPHÉES NSI

Édition 2022

**DOSSIER DE CANDIDATURE
PRÉSENTATION DU PROJET**

NOM DU PROJET : Les Chroniques d'Arcadia

> PRÉSENTATION GÉNÉRALE :

• *Créer un jeu de rôle textuel avec un système de combat, plusieurs énigmes, des interactions avec l'environnement et des personnages non-joueurs, avec une représentation graphique dans la console Python. Nous avons travaillé sur l'expérience immersive, en implémentant un système de "karma" et de "morale" affecté par les choix du joueur. Nous sommes restés sur une utilisation de niveau terminale de Python.*

• *Origines : Fusion de nos deux premiers projets de P.O.O. réalisés en cours de NSI (jeu de rôle + jeu de combat tour par tour)*

Intérêts :

- *La rejouabilité est un des aspects majeurs. Le joueur commence sa partie en choisissant entre 3 classes de héros différentes, et aura des décisions importantes à prendre tout au long de l'aventure. La mort du héros fait partie de l'expérience, il sera parfois mené à mourir et ressusciter pour résoudre des énigmes, ou pour vaincre certains ennemis.*
- *Ainsi, les décisions du joueur seront toujours pris en compte. Et ces mécanismes sont présents afin de l'impliquer davantage, de créer chez lui des émotions de satisfaction, de frustration, de victoire...*
- *Le monde du jeu est fermé, c'est-à-dire qu'il est construit sur 350 cases où le joueur peut aller dans les 4 directions cardinales. Pour cela chaque case est paramétrée sous forme de liste contenant les 4 cases voisines (None si absence de porte ou de case), et un dictionnaire définissant les éléments présents. Ce dictionnaire possède tous les éléments nécessaires pour la fonctionnalité du jeu.*
- *Le combat est un jeu en tour par tour, avec une réserve d'actions utilisant des points d'énergie. 4 actions différentes sont possibles : se déplacer, attaquer, compétence (attaque spéciale), ne rien faire. Il existe aussi des actions particulières durant des rencontres spéciales comme par exemple "sauter dans l'abîme".*
- *Un système de sauvegarde utilisant le module pickle est nécessaire au regard des cinq heures environ pour finir le jeu.*

> ORGANISATION DU TRAVAIL :

• L'équipe Mathéo Bideault et Lucas Hall, élèves de terminale NSI. L'image d'illustration a été réalisée par Enzo Poilane, élève de terminale.

• Répartition des tâches :

- Mathéo créé seul les fichiers : `fichier_principal.py`, `fonction_projet.py`. et `classe.py` contenant le fonctionnement du jeu hors combat.
- Lucas a créé seul les fichiers : `représentation.py`, `mod10.py`, `alimcombat9.py` et `Combat10.py`
- Les autres fichiers ont été créé ensemble. Lucas a pris en main les combats, la représentation graphique et les dialogues. Mathéo s'est chargé du cœur du programme : la fonctionnalité, les interactions hors combat, les énigmes...

• Organisation du travail : Depuis que nous avons décidé de participer aux Trophées NSI, nous travaillons ensemble presque tous les jours au sein et hors de l'établissement scolaire. Grâce à l'application Discord – et secondairement la plateforme scolaire e-lyco-, sur laquelle nous avons pu communiquer et faire évoluer nos programmes. La quantité de travail a été importante une première fois lors de la restitution du projet POO avant, durant et après les vacances de la Toussaint.

LES ÉTAPES DU PROJET :

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

Origines de deux projets POO créés à la Toussaint 2021 :

- Le projet de Lucas était un jeu de combat tour par tour inspiré de League of Legends (15 heures en cours et 15 heures de travail personnel). Ce jeu était orienté "joueur contre joueur".
- Le projet de Mathéo était un jeu de rôle textuel inspiré du système D20 (15 heures en cours et 35 heures de travail personnel, dont 25 heures pendant les vacances de la Toussaint). Celui était constitué d'un village, d'un labyrinthe et d'une rencontre finale; soit une carte de 100 cases.

A l'annonce de la création de "les trophées NSI" : Idée de fusion des deux projets en un seul jeu d'aventure.

Durant les vacances d'hiver un autre projet était demandé et n'a pas permis une avance technique du jeu. Cependant des idées d'améliorations ont germées : enrichissement de l'histoire, création d'énigmes, rythmer le jeu et l'utilisation d'une variable pour enregistrer les choix du joueur.

Début avril, les premières tâches étaient :

- Créer un rendu graphique,
- Ajouter les énigmes,
- Faciliter les améliorations de la liste des salles de la carte (construction et alimentation d'un dictionnaire).

Pendant les vacances de Printemps :

- Modifier le programme de combat "joueur contre joueur" en combat "joueur contre machine",
- Ajout des interactions dans les cases,
- Correction de nombreux bugs (environ 50).

Ces dernières semaines :

- Finalisation de l'histoire qui s'est construite tout au long du projet,
- Finalisation de l'alimentation (dialogues, armes, compétences, personnages),
- Personnalisation de combats avec des ennemis particuliers,
- Correction des derniers bugs,
- Création de la vidéo et de la documentation.

Le travail à partir de janvier 2022 a été d'environ de 12h en cours et d'environ 100h pour Lucas et environ 150h pour Mathéo.

L'organisation a été précise au début et milieu de projet (tableau de tâches avec leur validation), mais elle n'a pas pu être suivie durant ces 3 dernières semaines.

> FONCTIONNEMENT ET OPÉRATIONNALITÉ :

- Ce qui est terminé :

- La représentation du jeu sur la console
- Les combats : les rencontres aléatoires et 9 combats spécifiques.
- L'ensemble du jeu (histoire, fin, résurrections...)

- En cours de réalisation :

- Il reste encore quelques bugs à corriger.
- L'insertion des quêtes secondaires.

- Reste à faire :

- Prendre le temps de tester dans son intégralité le jeu pour identifier les bugs restants,
- Distinguer le "karma" de la "morale". Pour l'instant, elles sont identiques,
- Instaurer la difficulté adaptative,
- Traduire en anglais l'ensemble des textes,
- Durant le combat, il reste à implémenter le système "D20". Il faudra aussi mieux intégrer les statistiques des héros, et enfin prendre en compte l'armure, à l'image de l'arme qui elle est finie.
- Les consommables (potions de soin, de mana, d'énergie...) sont aussi à créer.

- Approches mises en œuvre pour vérifier l'absence de bugs et s'assurer de la facilité d'utilisation du projet :

- Dans un premier temps, les fonctions ont été testées indépendamment. L'utilisation du print() localisé a été très utile.
- Il est plus difficile de tester l'ensemble du programme. La majorité des bugs viennent de l'interaction entre les fonctions.
- Un contact a été réalisé avec l'association de jeux vidéo ffjv, et particulièrement celle de Laval, RedFox Gaming. Dès que le jeu est fini, soit à la fin de cette semaine, un test pourra être réalisé par les joueurs de l'association. Un retour plus riche pourra être réalisé.

- Difficultés rencontrées et solutions apportées :

- La récupération des effets - par exemple, l'activation d'un levier pour ouvrir une porte - et de l'associer aux autres effets - ici, l'ouverture de la porte - a été difficile. Une modification des enregistrements dans le dictionnaire a été une solution efficace, mais a augmenté sa complexité, une liste de variable au lieu d'un enter modifiable. Cependant, cette solution plus complexe a rendu la correction de bug plus facile.
- La réunion des deux premiers projets a comporté des difficultés en particulier dans la fusion des classes de joueurs. Deux points de vue différents ont été présents : celui de l'incarnation, la classe "héros", et celui du jeu la classe "joueur". Pour simplifier sans reprendre l'ensemble du travail, le joueur est devenu un attribut du héros.

> OUVERTURE :

- *Idées d'améliorations :*
 - *Distinguer le « karma » de la « morale ». Par exemple, la morale aura une influence sur la fin du jeu, tandis que le karma a un effet à court terme et évolue tout au long du jeu.*
 - *Ajouter un aspect aléatoire durant les combats. Pour l'instant les attaques réussissent automatiquement.*
 - *Adapter la difficulté tout au long du jeu, afin qu'elle s'adapte au choix du joueur. L'objectif est d'améliorer son expérience de jeu.*
- *Stratégie de diffusion pour toucher un large public :*
 - *Présenter aux élèves de première NSI à notre lycée.*
 - *Présenter le projet durant le concours RoboRAVE, Innov'bot : concours de robotique à Craon où 400 collégiens et lycéens seront présents.*
- *Analyse critique du résultat (si c'était à refaire, que changeriez-vous dans votre organisation, les fonctionnalités du projet et les choix techniques ?)*
 - *Quelques fonctions sont à optimiser. Parfois, le même calcul est réalisé trop de fois.*
 - *Avoir plus de retour pour mieux maîtriser la difficulté, et aussi mieux adapter la vitesse du texte,*
 - *Prévoir plus de temps dans la réalisation de la documentation du projet.*
 - *Le dictionnaire est plus facile à utiliser à plusieurs qu'une liste. Par exemple pour accéder à une arme, et ses caractéristiques, il faut parcourir toute la liste (ou partiellement) pour la retrouver. Avec un dictionnaire, il suffit d'avoir la clé. Il est donc plus rapide, plus facile à utiliser et donc plus judicieux.*
 - *Cependant, nous sommes assez surpris par le nombre d'heures total consacré pour ce projet. Le projet été fort agréable à concevoir de sa conception à sa réalisation. La création de tout un univers, essayer d'innover dans la conception des jeux-vidéos nous encore plus motivé pour le finir.*

DOCUMENTATION

- **Spécifications fonctionnelles :**

- Lancer uniquement le programme « fichier_principal.py »
- Détacher la console python et la basculer en plein écran pour un meilleur confort.
- Respecter les commandes indiquées (le programme peut s'arrêter sinon).
- Utiliser les touches « zqsd » pour se déplacer (un rappel est systématiquement réalisé).
- La balise de résurrection permet de revenir à la case où elle a été posée, après une mort du héros. Attention, toutes les actions et choix sont toujours sauvegardés.
- Si les combats sont trop longs, le mage, avec son bâton à équiper, permet de vaincre les ennemis en un seul coup.
- Un code « secret » existe pour se téléporter. Il faut taper « /TP » durant le choix des actions (seulement) et ensuite écrire les coordonnées de la case de destination. Les coordonnées correspondent par la valeur horizontale (verte) puis verticale (bleu), cf. plan détaillé en bas de ce document. Par exemple en tapant « /TP » puis « 1111 », cela permet de se téléporter au centre du village et ainsi passer le tutorial.

- **Spécifications techniques :**

- Python version 3.6 au minimum
- Les modules utilisés : colorama, pickle, time, random, sys et os
- Windows est préférable

- **Illustrations, captures d'écran : Solution de la première énigme.**

- Principales touches utilisées pour jouer :



- Chaque lettre représente un élément ou un personnage, consultable dans le programme « représentation » et « alim-combat », comme par exemple :

```
pics="W"          murN="      "
levier="L"        porteON="  ^  "
porte=None       porteFN="  _v_ "
fontaine_rez="R"  murWxE=" | "
monstre="0"       porteOW="< "
bouton="B"        porteFW="> "
truc_pour_mort="S" porteOE="> "
PNJ="p"          porteFE="< "
coffre="C"        porteOS="  _v_ "
Esc="E"          porteFS="  ^  "
table="T"         rien="      "
chaisse="C"
maison="M"
maison_casser="m"
```

