Rapport

Mini\_ Projet N°1,Groupe 16

(Malak Baradia – Ancona Enzo – Apetovi Assignage-Critelli Lucas- Lefevre Noham – Woiche Jonas) Bloc-1,année académique (2025-2026)

Jeu de rôle en python

* Introduction :

Le but de ce projet est de créer un jeu uniquement utilisable via des instructions en python , dans lequel le joueur peut créer un charactère (character) avec des arguments (character,variety,reach,player\_life) et les faire évoluer ,afin d’interagir (attack) avec la créature .

Le jeu nous a permis d’exploiter les connaissances qu’on avait et les notions qu’on a vu durant les séances de cour (boucles,random,chaine de formatage)

* Le déroulement du travail :
* 1ère séance obligatoire(10/10/2025) :On a lu les consignes des jeux, essayé de comprendre ce qu’on doit faire et on a réparti les tâches entre nous [start the game, création et amélioration du charactère (Malak),création de la créature (Lucas),attack(Noham et enzo ),pouvoirs spécifiques(Jonas).

2ème séance obligatoire (13/10) :On a préparé les spécification des fonctions demandés par les assistants , on a détaillé et expliqué chaque fonction et cela été encadré par les assistants .

On a assisté au séance de td et tp (14/10 et 15/10) :On a essayé des parties de code et on a regroupé le travail , de plus on a assisté à la séance de lire et comprendre pour pouvoir rédiger le rapport du projet .On a rencontré des problèmes lors de la vérification des fonctions .

* Corps du rapport :

Le jeu est divisé en plusieurs fonctions, réparties dans deux fichiers :

* mini\_projet → contient les fonctions principales du jeu.
* gaming\_tools.py → contient les fonctions utilitaires (création, modification et accès aux données des personnages et de l’équipe).

Le programme est composé de plusieurs fonctions qui sont logiquement reliés :

* start\_game() :Elle remet le jeu à zéro (reset\_game()), donne 50 pièces d’or à l’équipe.
* create\_character(character, variety):Permet au joueur de créer un personnage avec un nom et un type de son choix (character , variety).Les caractéristiques (strength , player\_life ) dépendent du type choisi (dwarf : force et vie élevées (10–50),elf : force et vie moyennes (15–25), healer, wizard, necromancien : force et vie faibles (5–15)). La portée dépend également du type (longue : dwarf, elf /courte : healer, wizard, necromancien). Une fois le personnage créé, il est ajouté au jeu via add\_new\_character().
* upgrade\_character(character) :Elle sert à améliorer(strength , player\_life)pour 4 pièces d’or avec un tirage aléatoire qui détermine le succès de l’amélioration. Elle vérifie d’abord si le joueur possède assez d’argent et si le personnage existe.
* create\_creature() :La fonction à pour but de créer une créature (creature)avec un nom choisis par le maitre du jeu , une force (strength) , une vie (player\_life) et une portée (reach) données aléatoirement.
* attack(character, \_creature):Cette fonction se charge des interactions (combats , attack )entre les charactères et la créature .Compare la portée du personnage et celle de la créature pour donner les dégâts .
* Le schéma ,teste et résultat :

start\_game

create\_character

creat\_creature

Upgrade\_charac-ter

Special\_ability

attack

* Conclusion :

Ce projet a consolidé nos compétences dans la mise des spécifications des fonctions ainsi que dans l’utilisation des connaissances acquises au cour du cours , en plus il nous a aidé à développer l’esprit d’équipe au sein du groupe et diviser les taches entre nous.