

# Rapport

Trinôme : Barbaud Marius / Garoux Ewan / Patel Lucah

## Éléments ajoutés :

- **Gameplay**

- **Interface Graphique** : l'interface graphique utilise **pygame**, en ouvrant une fenêtre on traduit chaque élément de la Map en un objet dessiné avec les **Sprites** grâce à la fonction **donne\_carte** dans la classe Ecran, pour les déplacements on utilise les flèches de déplacement au lieu du **getch**. Pour rafraichir l'écran la fonction **mettre\_a\_jour** permet un jeu continu et stable. La partie droite de l'écran est consacré à l'Hero avec **dessine\_sante** et **dessine\_inventaire**.
- **Point d'expérience** : l'attribut d'instance de classe Hero : **xp** augmente chaque fois le héros tue un monstre, il y a un cap d'xp : **xp\_cap**, quand ce cap est atteint le héros augment de niveau avec la fonction **earn\_xp** et gagne de l'hp de plus le cap d'xp augment aussi avec la fonction **level\_up**.
- **Inventaire limitée** : dans la fonction **take** de la classe Hero il y a une vérification de la longueur de l'inventaire s'il est plein alors l'élément est détruit.
- **Diagonale** : Dans l'attribut de classe Game : **\_actions**, on a ajouté la touche '**a**', '**e**', '**w**', '**c**' qui sont associées respectivement aux coordonnées (-1, -1), (1, -1), (-1, 1), (1, 1) pour gérer le déplacement en diagonale.

- **Objet**

- **Armures** : l'attribut d'instance de la classe Créature : armure, augment lorsque le héros rencontre un bouclier dans la Map.

- Salles

- **Gestion des salles** : On a choisi d'isoler la salle de Trésor et Boutique en parcourant les salles dans la Map pour trouver celle avec le trésor et de ne pas la décorer avec des monstres et des équipements ou de l'or.
- **Piège** : Un élément n'ayant pas d'hp mais qu'une force lorsqu'une créature rencontre cet élément la créature reçoit des dégâts équivalents à la force du piège et puis est enlevée de la Map.
- **Trésor** : Un élément lors de la rencontre avec le Hero parcourt son inventaire avec : **check\_elem**, pour vérifier s'il possède une clé, si cela est le cas le Hero obtient un Equipment au hasard avec : **randTake** et la clé est supprimée de l'inventaire avec : **remove**. Si l'héro ne possède pas de clé un message s'affiche expliquant qu'il doit avoir une clé pour accéder au trésor.
- **Boutique** : Lorsque le héros rencontre le shop la variable : **shopping** est affectée la valeur True, puis le héros parcourt une while boucle jusqu'à qu'il choisisse une action, pour acheter il suffit de choisir l'équipement voulu pour vendre il faut appuyer sur la touche 's'. La fonction **draw\_shop\_item** permet de garder les éléments vendus par le marchand sur l'étage s'ils sont vendus l'élément disparaît. Les fonctions **value** et **verification** de la classe shop vérifient respectivement la valeur de l'élément acheter et si l'élément peut être vendue, c'est à dire si l'héro a assez d'or.

- Monstre

- **Rapides** : Une créature qui a l'attribut de classe Créature : **is\_fast**, cela lui permet de d'effectuer un pas en plus ou une attaque en plus dans la méthode **move** dans Map.
- **Invisibles** : Une créature qui a l'attribut de classe Créature : **is\_invisible**, cela lui permet de d'effectuer des déplacements sans pouvoir être vu, il apparaît dès qu'il rencontre le héros.
- **Poison** : L'araignée peut empoisonner s'il est tué grâce à l'attribut d'instance **poisonned** qui devient True pour l'héro. Quand l'héro effectue un mouvement si l'attribut **poisonned** est True le héros perd 2 hp et l'attribut est réinitialisé à False