

Projet Mac Gyver Escape par Pierre Le Thiec

lien vers git : <https://github.com/ometeore/McGiver.git>

démarche :

La construction du jeu s'est faite en trois temps ; la première moitié ayant eu pour but de développer les fonctionnalités du jeu uniquement sur le terminal sans utiliser l'utilitaire pygame, il résulte d'un programme fonctionnel mais assez dur à analyser pour un œil non averti.

Cette partie est celle qui m'a fait faire le plus de « choix » techniques sur la manière dont est construit le code, notamment dans le découpages des classes et leurs interdépendances. Il en résulte encore à l'étape finale de nombreuses traces dont certaines qui peuvent sembler superflues mais qui ont eu un sens lorsque le jeu nécessitait d'être affiché par des print et des `__repr__`.

La seconde phase est survenue à partir du moment où par le terminal j'ai pu afficher une matrice de 15*15 sur laquelle chaque élément possède un attribut le définissant aux yeux du plateau et du héros. L'objectif fut alors de l'ouvrir dans une fenêtre en associant des templates à chaque élément et à donner un aspect graphique plus agréable au tout. Plus que du remplacement de mon ancien code cette partie a nécessité une réécriture massive d'une bonne partie du code allant parfois jusqu'à en modifier certains aspects de sa logique. Elle m'a aussi fait faire un peu de retouche photoshop (redimensionnement).

Enfin une fois que l'affichage pygame et la structure logique du jeu furent rendues opérationnelles, il a fallu soigner les détails. Créer un menu pour que l'utilisateur puisse revenir et sortir du jeu facilement, donner des affichages plus soignés au plateau en général et « nettoyer le code ». J'ai utilisé le logiciel pylint pour essayer de respecter au plus près la norme PEP8.

Pylint :

Voici mes différentes notes avec pylint :

mc_giver_escape.py → 1,64/10

classes/hero.py → 9,55/10

classes/plateau.py → 7,88/10

classes/case.py → 8,00/10

classes/seringue.py → 8,17/10

La principale source d'erreur vient d'un problème de détection de pylint (notamment pour 7 pts sur mc_giver_escape.py) des variables contenu dans pygame.locals.

mc_giver_escape.py:7:0: E0611: No name 'K_LEFT' in module 'pygame.locals' (no-name-in-module)

difficultés rencontrées :

Le fait d'avoir d'abord travailler sur une représentation type terminal, à fait que mon code n'était pas conçu pour pygame lorsque j'ai décidé d'y apporter de la couleur (l'affichage se faisait n'importe où les prints étaient réparti partout dans le code). Il reste des classes (la classe Case et celle Seringue) dont je remet en question l'utilisation, même si, si il fallait que je recommence, je pense qu'il serait plus « propre » de diminuer l'aspect procédural du code.

C'est le projet où j'utilise pour la première fois git hub (et la dernière fois que je ne l'utilise pas). Je dois reconnaître que je ne m'attendait pas à la puissance de l'outil et je n'y suis pas rentré dedans avec beaucoup de volonté ayant du mal à percevoir la manière dont cela pourrait m'aider.

Je n'ai compris que tardivement (cette semaine) la vraie force du commit dont le rôle se confondait dans mon esprit avec le push.

Les transparences graphiques m'ont posé des problèmes puisque je n'ai pas voulu les résoudre sur photoshop mais plutôt en utilisant les fonctions de pygame (choix que je ne ferai pas pour un projet plus sérieux).