



LES
TROPHÉES **NSI**

Édition 2024

DOSSIER DE CANDIDATURE
PRÉSENTATION DU PROJET



PROJET :

Pokémon MG
POKÉMON MANAGEMENT GAME

MEMBRES : Yanko Lemoine
Mathias Farvacque

NIVEAU D'ÉTUDE : Terminale

ÉTABLISSEMENT SCOLAIRE : Lycée Raymond Queneau, Villeneuve d'Ascq (59)

ENSEIGNANT DE NSI : Stéphane Ramstein

PRÉSENTATION GÉNÉRALE

Qu'est-ce que Pokémon MG ?

Pokémon MG est un jeu vidéo de gestion basé sur l'univers de Pokémon. C'est un jeu se rapprochant des jeux de type « Rogue-like », c'est à dire dans lequel le joueur va devoir combattre des adversaires pour gagner de l'expérience et du butin, lui permettant ainsi de progresser. À la différence des jeux Pokémon classiques, **Pokémon MG** est un jeu de gestion : le joueur ne pourra pas contrôler physiquement un personnage du jeu mais va incarner un dresseur Pokémon novice et aura la possibilité d'exécuter un certain nombre d'actions chaque tour. Les actions qu'il réalisera auront pour but de renforcer son équipe de Pokémon. À la fin de chaque tour de jeu, le joueur devra affronter un puissant dresseur et doit impérativement gagner sinon la partie se termine. Le but du jeu est donc de continuer l'aventure le plus longtemps possible avant d'être battu par un autre dresseur !

D'où vient cette idée de projet ?

Ce projet est né il y a près d'un an : M.Ramstein nous présentait alors en cours de NSI le jeu « Terminus », un jeu en ligne conçu pour nous apprendre les commandes du terminal sous Linux. Le jeu se présentait sous la forme d'une aventure dans laquelle, pour avancer, on devait se déplacer dans un répertoire de dossiers grâce aux commandes Linux. Yanko s'inspirant de ce jeu a eu l'idée de coder en python un jeu similaire mais dans l'univers de Pokémon en remplaçant les commandes Linux par des mots clés comme `/spawn` ou `/fight` car nous venions tout juste d'apprendre à utiliser les fonctions de bases de python comme `print()` et `input()` et le principe de conditions avec les mots clés `if` et `else`. Par la suite, ce projet a évolué et est devenu bien plus ambitieux qu'un simple jeu jouable en console : on voulait créer un vrai jeu avec une vraie interface graphique ! Donc on s'est renseigné et avec l'aide de M.Ramstein on a découvert le module `pygame` de python et au fur et à mesure de ce qu'on apprenait en cours de NSI, on améliorait le projet pour finalement aboutir à **Pokémon MG**.

Quels sont les objectifs du projet ?

Étant nous-mêmes fans de jeux vidéos, nous avons pour objectif de réussir à coder, à deux, à partir de rien, un jeu vidéo de qualité pouvant rivaliser avec les jeux auxquels on joue au quotidien en utilisant ce qu'on apprend en cours de NSI et en bénéficiant de l'aide de M.Ramstein et des stagiaires (M.Braun et M.Delplace).

ORGANISATION DU TRAVAIL

Quel est le rôle de chacun et pourquoi ?

Mathias

Temps passé sur le projet : ~ 200h

Mathias s'est essentiellement occupé du back-end, c'est à dire les aspects du code plutôt axés sur des calculs mathématiques : notamment la conception de fonction mathématiques cohérentes comme les fonctions qui définissent la manière de déterminer le niveau d'apparition des Pokémon selon le niveau du joueur, son équipe, le tour de jeu actuel, etc. Mathias est plus à l'aise avec les maths que Yanko donc le fait que ce soit lui qui code plutôt les fonctions de calculs internes au jeu s'est fait assez naturellement.

Yanko

Temps passé sur le projet : ~ 300h

Yanko s'est chargé de coder toute la partie front-end du code, c'est à dire tout ce qui touche à l'interface graphique et l'affichage des éléments du jeu. Il s'est aussi occupé de créer la structure du code et les relations entre les différentes parties du code et modules. C'est également lui qui a créé tous les designs du jeu. Yanko est très à l'aise avec Photoshop et avec tout ce qui touche au graphisme donc il s'est très naturellement occupé de créer les fichiers images et de les implémenter dans le code. Et étant donné que l'idée du projet vient de lui, il a une vision beaucoup plus précise de ce qu'il attend et c'est donc beaucoup plus simple de travailler ainsi.

Quand travaille-t-on sur le projet ?

On a essentiellement réalisé le projet durant l'année de terminale en cours de NSI mais aussi en dehors des cours dans le cadre de l'atelier scientifique du lycée. Certains aspects du projet ne pouvant cependant pas être réalisés sur place au lycée comme le design, une partie du projet a également été réalisée en dehors du lycée, pendant les week-end et les vacances. Étant donné nos ambitions de coder un jeu de bonne qualité, on a dû consacrer énormément de temps au projet et il aurait été impossible de tout réaliser sur le temps scolaire.

Comment s'organise-t-on pour travailler ensemble ?

Pour pouvoir travailler en parallèle sur le projet sans créer de conflits entre ce que chacun code, on a créé un répertoire Github pour bien structurer le projet et avoir une sécurité. Le répertoire est le suivant : github.com/Yanko77/PokemonMG
Pour coder en python, on utilise la version communautaire et gratuite de l'IDE Pycharm de l'éditeur JetBrains qui est spécialement conçu pour le langage python. De plus, ce logiciel offre la possibilité de travailler en simultané via la fonctionnalité « Code with me ». Donc pour pouvoir s'aider mutuellement, on travaillait, la majorité du temps, ensemble grâce à cette fonctionnalité de l'IDE.



* Personnage du jeu : le Maître dresseur.

Plus précisément ce que chacun a codé...

Yanko

Affichage de tous les éléments du jeu.

Gestion de l'expérience utilisateur:

- Interactions Clavier/Souris : Boutons, Déplacement d'éléments par le joueur.
- Gestion des interactions entre les éléments du jeu amenés à se superposer.

Gestion des animations

Gestion des actions en temps réel.

Mathias

Fonctionnalité de sauvegarde et de chargement de la sauvegarde

Fonctions internes de calculs :

- Calculs des dégâts lors des combats
- Calculs du scaling de la puissance des Pokémon et dresseurs.
- Algorithme des PNJ (Personnages Non-Joueurs) et de leurs actions.
- Algorithmes de recherche d'objets dans la boutique.

Ny-Harena, un ami extérieur au projet faisant partie du même groupe de classe de NSI, s'est occupé des effets sonores du jeu dont la bande-son originale !! Ces derniers ne sont pas encore implémentés dans le jeu mais ça ne saurait tarder !

FONCTIONNALITÉ & OPÉRATIONNALITÉ

Où en est le projet ?

À la date du 29 mars, le jeu est entièrement jouable et fonctionnel. Le projet est finalisé à hauteur d'environ 85%. C'est à dire qu'il nous reste encore quelques fonctionnalités à ajouter pour qu'on puisse considérer le projet comme achevé. On est cependant très satisfait de l'état du jeu, au moment où on le présente.

Comment sont détectés les bugs ?

Pour détecter des bugs, on a fait appel à des bêta-testeurs, Martin et Corentin, qui font partie de la classe de NSI et qui jouent très régulièrement au jeu. On fait souvent des réunions pour leur demander leur avis sur certaines futures fonctionnalités et on leur demande des retours sur la manière de jouer et sur des potentielles améliorations.

Comment garantir une facilité d'utilisation ?

Nous souhaitons créer un jeu accessible donc nous avons un regard très critique sur les ressources demandées par le jeu. Mathias réalise très régulièrement des tests de performance du jeu sur plusieurs machines, plus ou moins puissantes, pour vérifier si le jeu est correctement optimisé et ne demande pas plus de ressources que nécessaire. On est conscient que puisqu'on est relativement débutant en programmation, notre jeu sera évidemment moins bien optimisé que les jeux actuels. Le langage python n'est d'ailleurs pas le plus optimisé puisqu'il doit être compilé avant d'être exécuté ce qui le rend plus lent. Grâce à ces tests réguliers, on a pu trouver de gros problèmes d'optimisation et les corriger. Par exemple, lorsqu'on perdait une partie, pour en relancer une nouvelle, on se contentait de réinstancier notre classe Game gérant l'ensemble du jeu. Cependant, on a remarqué que plus on relançait ainsi le jeu, plus le jeu utilisait de ressources en mémoire vive (RAM). Cela est dû au fait qu'on ne peut pas forcer python à supprimer un objet de la mémoire vive car il est censé se charger d'automatiquement faire le nettoyage des éléments qui ne sont plus référencés et de les supprimer. Or il ne le faisait pas. On a donc dû revoir notre façon de relancer le jeu. Désormais on ne fait plus de réinstanciation mais on réinitialise simplement la classe Game. C'est de cette façon qu'on essaye de garantir une facilité d'utilisation du jeu. Et on est aussi évidemment aidé par nos bêta-testeurs qui nous donnent leur avis de joueur sur la manière de jouer et donc la facilité d'utilisation du jeu.



Liste des fonctionnalités du projet.

Terminé :

- Écran d'accueil
- Actions : SPAWN, TRAIN, ITEMS, EVOL
- Sac d'objets
- Équipe
- Système de combat
- Système de sauvegarde
- Système de shop : Achat & Vente
- Optimisation des ressources :
Performances / Espace

En cours :

- Équilibrage : Gestion de la difficulté du jeu
Équilibrage des statistiques des
Pokémons/Objets/Dresseurs
- Debugging
- Gestion de certains effets spéciaux :
Prix d'objets variables
Attaques à effet longue durée (Healing)

À faire :

- Sound design : Implémenter les musiques.
- Actions : GRIND Ajouter des effets sonores :
- Écran de fin Bruits d'attaques
Bruits d'objets
Bruits de clics
- Crédits du jeu - Paramètres
- Mode plein écran

Quelles difficultés ont été rencontrées ?

La difficulté principale du projet est qu'on apprend et on s'améliore de jour en jour grâce à ce projet. De ce fait, notre manière de coder a évolué et les premières lignes de code, celles à l'origine du projet et sur lesquelles est basé tout le jeu, sont de moins bonne qualité. Cela nous contraint par moment à complètement réécrire un module.

À part cela, on ne rencontre pas de réelles grosses difficultés, si ce n'est le temps, car coder un jeu prend un temps considérable et nous ne sommes que deux. Cela peut être un avantage car c'est beaucoup plus simple d'être sur la même longueur d'onde et de se répartir le travail quand on est que deux, mais c'est aussi un inconvénient car il faut passer beaucoup de temps à coder. Et si on avait compté les heures, on aurait sûrement abandonné il y a bien longtemps.

LES ÉTAPES DU PROJET



OUVERTURE

Quelles sont les nouvelles fonctionnalités à venir ?

Nous souhaitons achever le projet d'ici la fin de l'année scolaire. D'ici là, on va essayer d'ajouter autant de fonctionnalités que possible pour rendre le jeu le plus complet possible. Il nous reste encore l'action GRIND à coder, il faut aussi implémenter les effets sonores car on a l'avantage d'avoir une bande-son originale ! Il serait également intéressant d'ajouter des paramètres pour améliorer l'expérience utilisateur car on a eu beaucoup de retours sur le fait que le jeu ne puisse, par exemple, pas être mis en plein écran, ce qui est un peu dommage.

Des idées d'amélioration pour le projet ?

L'avantage avec les jeux vidéos c'est qu'il est très simple de trouver des idées d'amélioration du jeu parce qu'il suffit de demander à nos joueurs leur avis. De plus, l'univers du jeu vidéo est très créatif et, de ce point de vue là, on peut dire que notre projet peut s'améliorer indéfiniment. Une idée d'amélioration finale pourrait être de créer une version multijoueur du jeu en ligne. Mais cela dépasse de très loin nos compétences actuelles et le cadre de la spécialité NSI. Mais cela reste une idée intéressante.

Comment auraient-on pu mieux faire ?

Peut-être qu'il aurait été judicieux d'utiliser un autre langage que python pour coder notre jeu car python n'est pas le langage le plus adapté dans ce domaine. Le langage C aurait pu être intéressant à étudier aussi car c'est celui utilisé dans le supérieur. Mais ce projet nous a justement permis d'apercevoir les limites des moyens techniques qu'on a utilisés.

Que tire-t-on de ce projet ?

Réaliser ce projet a été une excellente expérience car cela nous donne un aperçu de ce que c'est que de gérer un projet en équipe et en autonomie. Cela nous a, sans aucun doute, aidé dans notre apprentissage de la NSI et de la programmation en général et nous a permis de grandement progresser. Quand on regarde les toutes premières lignes du projet, on voit très vite tout le chemin parcouru...