

Rapport projet Java - Quête du progrès

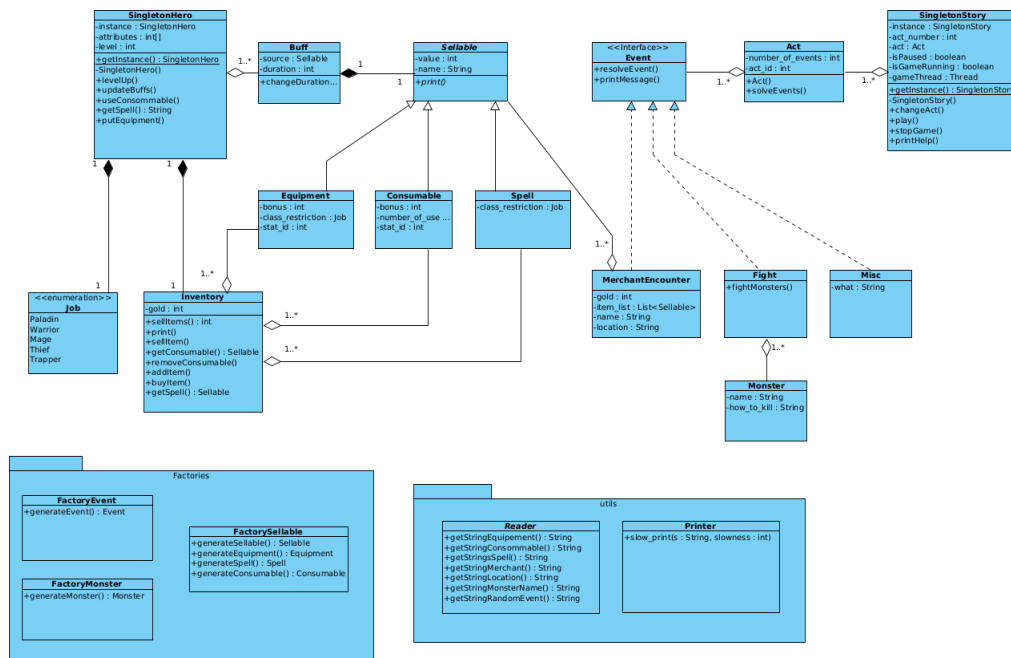
BRUNEAU Briac, BOUCHARD Corentin

1 Description du projet

Le but ce projet était de reproduire un "jeu" ressemblant à [Progress Quest](https://github.com/sunxbriac/quete_du_progres) directement dans la console. Le "joueur" peut choisir une classe, puis apprécier son personnage évoluer dans un univers med-fan, avec des événements aléatoires, des combats, du loot... Le code source est disponible ici : https://github.com/sunxbriac/quete_du_progres

2 Diagramme de classe

(L'image du diagramme est disponible directement en png dans le repo git.) Quelques fonctions auxiliaires, getters, setters ont été omis pour ne pas surcharger le diagramme.



Les classes Story et Hero sont centrales à ce projet. Le pattern Singleton a été utilisé, ce qui a permis un accès plus efficace dans différentes classes du programme et aussi pour rendre plus safe le multithreading. La classe Hero gère toutes les informations sur le personnage du joueur et ce qu'il peut faire. La classe Story gère l'avancée du jeu, la génération d'événements aléatoires, la communication avec l'utilisateur via le terminal. La classe abstraite Sellable représente tous les objets que le personnage pourra obtenir par des achats à un marchand, ou en les trouvant sur un monstre. Ces éléments qui doivent être régulièrement générés le sont via une Factory (ainsi que les monstres et les événements).

3 How to use

A l'exécution, le programme est normalement assez descriptif pour ne pas nécessiter de notice. Au début de l'aventure on choisit sa classe, et le jeu commence. Entrez "pause" pour mettre en pause les événements (la pause devient effective à la fin d'un événement, ce qui peut prendre quelques temps

si c'est un évènement fight avec plusieurs monstres à combattre.). Depuis la pause, "help" permet de connaître les commandes disponibles.

4 Organisation et difficultés

Un temps important au début du projet a été consacré à la conception, assez peu de changements ont été fait entre le diagramme initial et le projet final. Conception faite, nous sommes partis chacun d'un côté pour définir les classes et les méthodes, sans nécessairement compléter la logique des fonctions mais pour avoir le plus vite l'ossature complète du projet. Briac a commencé sur les classes proches de la classe Hero et Corentin des classes proche de Story. On d'abord été fait un affichage fonctionnel de base, la gestion du multithreading pour mettre en pause le jeu, la génération aléatoire et lecture du contenu généré dans des fichiers. La logique d'utilisation d'objets, et de gestion des attributs ont été développés à la fin.

Assez peu de difficultés ont été rencontrées durant le projet. La première était que nous ne travaillons pas sur le même IDE (IntelliJ/VScode), et cela a posé quelques soucis d'exécution et de path pour les fichiers. Sur la programmation, le plus difficile a été de gérer le multithreading pour permettre à l'utilisateur d'entrer des commandes alors que l'affichage et le jeu continue.