

Mini RPG Lite 3000

Guillaume Lachaud

11 mars 2022

CHAPITRE 1 : START OF AN ADVENTURE

I'm going on an adventure !

Bilbo Baggins, *The Hobbit*

Bienvenue dans Mini RPG Lite 3000, le plus simple des RPGs. Êtes-vous une bande de jeunes héros prêts à relever tous les défis ? C'est parti !

Votre objectif est de vaincre tous vos ennemis. Chaque combat gagné vous fait progresser et affronter des monstres encore plus terrifiants.

FONCTIONNEMENT

Au début d'une partie, on choisit le nombre de héros. Cela déterminera le nombre d'ennemis à affronter par tour. Ensuite, on choisit la classe de nos héros : ils peuvent être **Hunter**, **Warrior**, **Mage** ou **Healer**. Chaque héros possède une arme et une armure, ainsi que des potions et de la nourriture.

Une partie se comporte de plusieurs combats, jusqu'à la défaite de l'équipe. À chaque nouveau combat, il est possible de tomber sur un boss. Pendant un combat, chaque joueur peut décider d'attaquer, de se défendre, ou bien même d'utiliser un consommable, comme manger du lembas ou boire une potion. L'ordre d'attaque des joueurs et des ennemis est déterminé aléatoirement au début de chaque combat.

Chaque classe a des attaques différentes. Le **Hunter** tire des flèches (*jusqu'à n'en plus avoir*), le **Warrior** tape frappe au corps à corps, le **Healer** soigne avec des sorts tandis que le **Mage** utilise la magie contre les ennemis.

Si l'équipe gagne un combat, chaque personnage peut choisir entre plusieurs options :

- augmenter son armure ;
- augmenter les dégâts de son arme ;
- augmenter l'efficacité de la potion et de la nourriture ;
- augmenter le nombre de potions ou de nourriture ;
- augmenter le nombre de flèches (pour le **Hunter**), diminuer le coût en mana pour les sorciers, ou l'efficacité de leurs sorts.

CHAPITRE 2 : DIAGRAMME UML

Map-making had never been a precise art on the Discworld. People tended to start off with good intentions and then get so carried away with the spouting whales, monsters, waves and other twiddly bits of cartographic furniture that the often forgot to put the boring mountains and rivers in at all.

Moving Pictures, Terry Pratchett

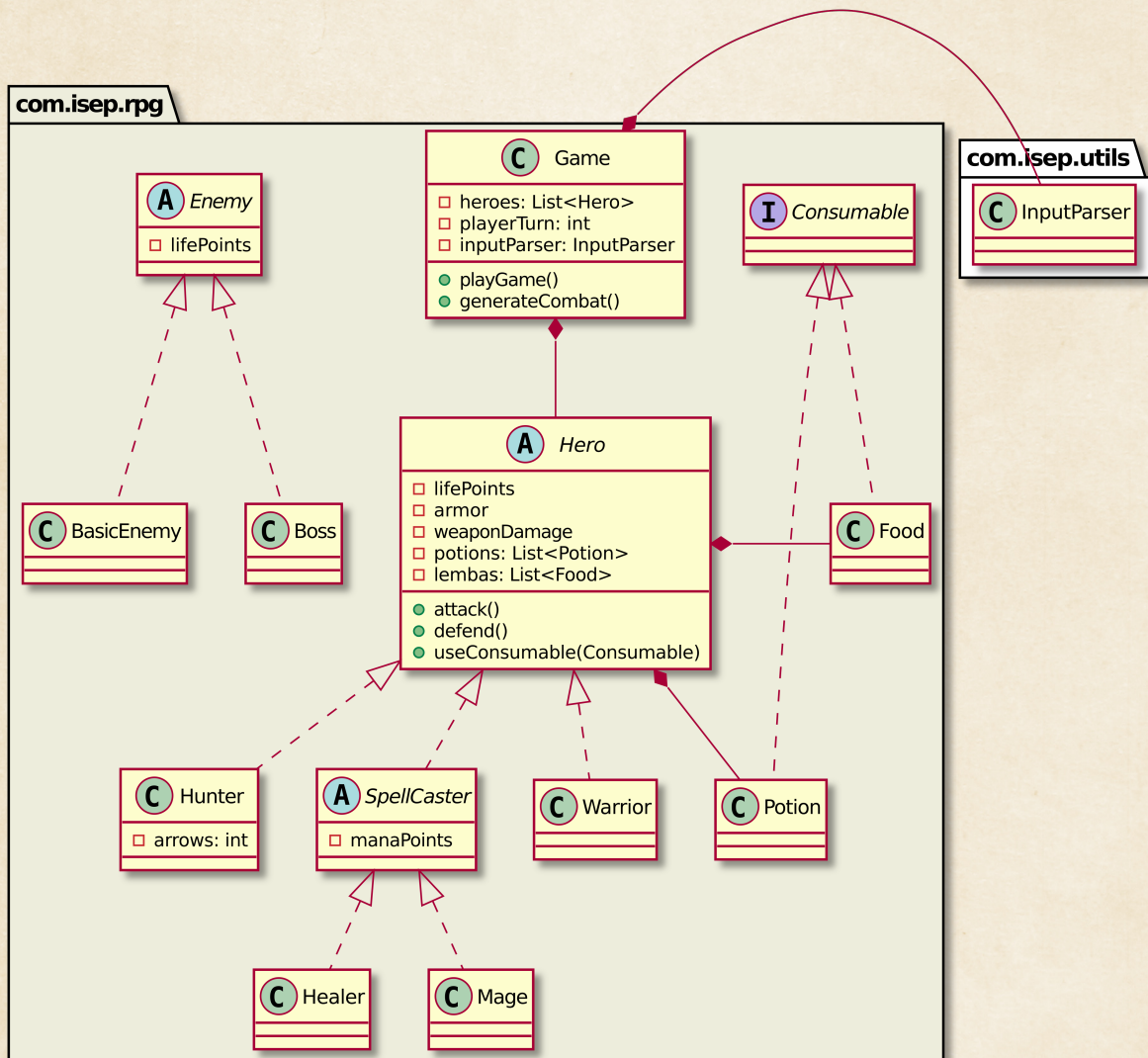


FIG. 2.1 : Diagramme UML du jeu

CHAPITRE 3 : TESTING YOUR CODE

Programmers and software engineers who dive into code without writing a spec tend to think they're cool gunslingers, shooting from the hip. They're not. They are terribly unproductive. They write bad code and produce shoddy software, and they threaten their projects by taking giant risks which are completely uncalled for.

Joel on Software, Joel Spolsky

Afin de jouer dans les meilleures conditions possibles, il faut s'assurer que certaines règles sont respectées. En voici quelques exemples :

- Si l'on prend plus de dégâts que l'on a de PV, on meurt ;
- si aucun joueur n'est en vie, on a perdu ;
- si l'on n'a pas assez de mana, on ne peut pas lancer le sort ;
- on inflige bien les dégâts dans la bonne fourchette de dégâts ;
- la quantité de flèches diminue quand on tire ;
- la quantité de mana diminue après avoir lancé un sort ;
- l'attribution de récompense fonctionne bien après une victoire.

CHAPITRE 4 : BUT WHAT ABOUT ME ?

There isn't a parallel of latitude but thinks it would have been the equator if it had had its rights.

Mark Twain

Votre mission, *should you choose to accept it*, est de programmer *Mini RPG Lite 3000*®. Vous devez implémenter les classes du diagramme UML, en ajoutant des attributs, des méthodes ou des classes là où vous pensez que cela est nécessaire. Vous devez aussi écrire des tests unitaires afin d'obtenir le score de *code coverage* le plus élevé possible.

Vous pouvez commencer par faire tout le squelette des classes suivi des tests unitaires, ou procéder classe par classe, en ajoutant les tests unitaires au fur et à mesure.

Le TP est à rendre en envoyant un lien vers le projet Git à votre encadrant.

