

Projet final

Contexte

Vous venez de rejoindre un studio de développement de jeux vidéos en tant que développeur backend. Vous avez été recruté ainsi qu'un autre développeur pour travailler à deux sur une nouvelle API REST à destination du prochain jeu vidéo du studio. Ce jeu est un jeu de cartes à collectionner, à faire évoluer et avec lesquelles les joueurs peuvent affronter les cartes d'autres joueurs. Ces cartes sont des héros, avec chacun leurs spécificités et leur rareté. Votre mission est de développer une API REST en partant des différents concepts du jeu présentés dans ce document ainsi que de développer les fonctionnalités demandées.

Technologies obligatoires

Git
Github

Technologies recommandés

Java 17 ou supérieur
IntelliJ
Maven 3

Compétences requise pour la réalisation de ce projet

Des connaissances en algorithmique
Des connaissances basiques en probabilités
La maîtrise de Git, d'un langage de programmation objet et du développement backend ainsi que des connaissances sur REST
Être capable de mettre en place une architecture hexagonale
Être en mesure d'écrire un code clean, respectant les principes SOLID, et de le tester

Concepts de jeu

Les héros

Tous les héros possèdent plusieurs caractéristiques communes :

- Un nom
- Un nombre de points de vie
- Un nombre de points d'expérience
- Une puissance
- Une armure
- Une spécialité
- Une rareté (commun, rare, légendaire)
- Un niveau allant de 1 à 100

La puissance

Un héros avec une puissance X retire X points de vie au héros attaqué.

L'armure

Lorsqu'un héros est attaqué il se voit imputer un retrait de points équivalent à ce calcul : puissance de l'attaquant - armure du défenseur. Par exemple, un héros avec 10 d'armure est

attaqué par un autre héros avec 20 points de puissance : $20 - 10 = 10$. Le héros attaqué perdra donc 10 points.

Les spécialités

Chaque spécialité de héros octroie des caractéristiques de base différentes

- **Tank**
 - Points de vie au niveau 1 : 1000
 - Puissance au niveau 1 : 100
 - Armure au niveau 1 : 20
 - Puissance supplémentaires contre mages : 20
- **Assassin**
 - Points de vie au niveau 1 : 800
 - Puissance au niveau 1 : 200
 - Armure au niveau 1 : 5
 - Puissance supplémentaires contre tanks : 30
- **Mage**
 - Point de vie au niveau 1 : 700
 - Puissance au niveau 1 : 150
 - Armure au niveau 1 : 10
 - Puissance supplémentaires contre assassins : 25

Selon la rareté d'un héros, ses statistiques de bases sont relevées de la manière suivante :

Commun : +0%

Rare : + 10%

Légendaire : + 20%

Les niveaux

À chaque fois qu'un héros remporte un combat, ce dernier remporte 1 point d'expérience. Tous les 5 points d'expérience gagnés le héros monte de niveau. Chaque gain de niveau voit les caractéristique du héros augmenter de 10%.

Exemple d'un mage commun de niveau 1 qui passe au niveau 2 :

- PV : $700 + 10\% = 770$
- Puissance : $150 + 10\% = 165$
- Armure : $10 + 10\% = 11$

Inscription joueur

Pour se créer un compte joueur il suffit de renseigner un pseudo. Au moment de la demande d'inscription attribuer un total de quatre jetons pour ouvrir des packs de cartes héros et créer un deck vide.

Ouverture d'un pack de cartes

Il existe deux type de packs, le pack argent et le pack diamant. Chaque pack donne au joueur un certain nombre de cartes, avec une probabilité de niveau de rareté définit.

Pack argent :

Jetons requis pour ouverture : 1

Nombre de cartes : 3

Probabilité de rareté :

Légendaire : 5%

Rare : 20%

Commune : 75%

Pack diamant :

Jetons requis pour ouverture : 2

Nombre de cartes : 5

Probabilité de rareté :

Légendaire : 15%

Rare : 35%

Commune : 50%

À chaque fois qu'un joueur ouvre un pack de cartes, les cartes gagnées sont ajoutées à son deck.

Deck

Le deck d'un joueur est l'endroit où ce dernier stocke ses cartes. Il n'y a pas de limite au nombre maximum de cartes possédées.

Système de combat

Chaque joueur peut accéder à la liste des joueurs inscrits et à leurs decks. Il peut ensuite décider d'engager un combat en tour par tour contre une carte d'un autre joueur avec une carte de son propre deck. Le premier héros dont les points de vie tombent à zéro à perdu. Tous les 5 combats gagnés, le joueur remporte un jetons d'ouverture de pack. Afin de garantir un semblant d'équité, un joueur ne peut attaquer un héros que si ce dernier à un niveau au moins égal à celui du héros qu'il a choisit d'utiliser pour attaquer.

Fonctionnalités attendues

1. Créations des héros en base de données
2. Rechercher les héros disponibles
3. Création d'un compte joueur avec solde de jetons et deck en base de données
4. Ouverture de packs et ajout des cartes au deck du joueur
5. Rechercher des joueurs et visualiser leurs decks
6. Engager un combat entre un héros et celui d'un autre joueur
7. Pouvoir retrouver tous les combats d'un héros (héros adverse, résultat du combat)

Critères de notation

- Le projet respecte l'architecture Hexagonale : 5 points
- Le projet respecte les principes SOLID : 5 points
- Le code du projet est correctement testé et respecte les autres principes abordés durant le cours : 5 points
- Le projet répond au besoin exprimé (compile, s'exécute correctement et propose toutes les fonctionnalités) : 5 points
- Point bonus : le projet utilise la librairie Vavr ou toute autre librairie permettant l'écriture du code selon les principes de la programmation fonctionnelle