

Concepts objets et Java

1ère année du Cycle Ingénieur en Informatique

Projet individuel

Plateforme de jeux vidéo multijoueurs

Année 2022-23



Introduction au projet

- ▶ Développement en langage Java d'une application (simplifiée) pour une plateforme de **gestion d'un réseau social organisé autour de parties de jeux vidéos multijoueurs**
 - objectif : mettre en œuvre de manière simple les notions abordées dans le module
- ▶ Projet à réaliser individuellement
 - partie commune : possibilité de se concerter pour la modélisation
 - parties optionnelles
- ▶ Les modalités de rendus sont précisées à la fin de ce document
 - la date de rendu est celle pour le module : **dimanche 23 avril 2023**
 - le projet contribuera à **30% de la note du module**
 - le rapport du projet est dans un format imposé

Cahier des charges : joueurs humains

- ▶ Un **joueur** humain inscrit sur la plateforme
 - est identifié par un pseudo (unique), un email et une date de naissance
 - est inscrit à la plateforme avec un **profil de membre** (↷)
 - peut éventuellement être **enfant** ou **tuteur**
 - un enfant doit avoir au moins un et au plus deux tuteurs adultes membres de la plateforme
 - un adulte peut avoir un certain nombre de membres enfants dont il est tuteur
 - peut avoir un certain nombre d'**amis** membres de la plateforme
 - un enfant ne peut avoir que des amis enfants
 - un adulte ne peut avoir que des amis adultes
 - possède une et une seule **machine de jeu** (ex. PC, PS3, GBA)
 - possède un ensemble de **jeux** pour sa machine

Cahier des charges : joueurs bots

► Un **joueur bot**

- n'a pas formellement à être inscrit dans la plateforme
- doit être ajouté par l'administrateur de la plateforme
- implémente un certain nombre de stratégies automatiques ("intelligences artificielles") pour un certain nombre de jeux
- dans la plateforme, au plus un seul joueur bot permet de jouer à un jeu particulier

Cahier des charges : profils de membres

- Un joueur est inscrit avec un **profil de membre** qui définit des **restrictions** sur la plateforme portant sur :
- (a) le nombre maximal de jeux possédés
 - (b) la possibilité d'acquérir ou offrir des jeux
 - (c) le nombre d'amis
 - (d) la possibilité d'inviter des joueurs comme amis

	(a)	(b)	(c)	(d)
standard	50	oui	100	oui (autres joueurs et bots) ses enfants uniquement
gold	illimité	oui	illimité	oui (autres joueurs et bots) ses enfants uniquement
enfant	30	non	10	non autres enfants uniquement
bot	illimité	non	(sans objet)	non

Cahier des charges : jeux vidéo

- ▶ Un **jeux vidéo** (simplification)
 - dispose nécessairement d'un mode multijoueur
 - (s'il est renseigné sur la plateforme)
 - ne permet que des parties multijoueurs à 2 joueurs
 - peut être joué en multijoueur depuis des machines différentes
 - permet des parties où l'un des joueurs au plus est un **bot**
 - une partie a pour résultat la victoire de l'un des deux joueurs
 - ce résultat n'est conservé que si les deux joueurs sont humains

Cahier des charges : collection de jeux vidéo


- ▶ La **collection de jeux vidéo** de la plateforme
 - est obtenue par lecture d'une ressource statique :
https://raw.githubusercontent.com/stef-aramp/video_games_sales/master/vgsales.csv
 - les données sont au format CSV (*comma-separated values*) :
Rank, Name, Platform, Year, Genre, Publisher, NA_Sales, EU_Sales, JP_Sales, Other_Sales, Global_Sales
 - exemple d'entrée :
2701, Portal 2, PC, 2011, Shooter, Valve Software, 0.33, 0.32, 0, 0.11, 0.76
 - on considère que tous les **jeux** remplissant un ensemble de conditions données possèdent un mode multijoueur et un module de jeux pour un bot
 - par ex., tous les jeux postérieurs à 2010 dont les ventes globales sont supérieures à 500 000 copies
 - l'ensemble des **machines de jeux** connues de la plateforme apparaissent dans la liste des jeux retenus

Actions permises par l'application

1. Connexion à la plateforme

- soit en tant qu'administrateur ou pour un pseudo
- (simplification) pas de mots de passe

2. Inscription d'un nouveau joueur (mode administrateur)

 un profil d'enfant ne peut être inscrit que par l'un de ses tuteurs

3. Affichage des informations sur un joueur

- mode administrateur : sans restrictions
- mode joueur
 - sans restrictions pour les joueurs **standard** et **gold**, incluant leurs jeux possédés avec leur ratio de victoire et leur nombre d'amis
 - pour les joueurs **enfant**, affichage simplifié du nombre de jeux possédés et du nombre d'amis, sans restriction pour leurs tuteurs

Actions permises par l'application

4. Invitation d'un joueur comme ami (mode joueur)
 - (simplification) acceptation automatique si ajout d'ami possible
5. Retrait d'un joueur comme ami (mode joueur)
6. Ajout d'un tuteur à un joueur **enfant** (mode joueur)
7. Affichage des informations pour un jeu
 - incluant le nombre de joueurs possédant le jeu
 - nom du **bot** permettant de jouer au jeu
8. Création d'un **bot** pour un **jeu** (mode administrateur)

Actions permises par l'application

9. Acquérir un jeu (mode joueur)
10. Offrir un jeu
 - mode administrateur : sans restrictions
 - mode joueur : uniquement à un joueur ami
 - ⚠ un joueur **enfant** ne peut acquérir un jeu que si l'un de ses tuteurs le lui offre
11. Proposer une partie multijoueur à un joueur ami
 - (simplification) indication immédiate du vainqueur de la partie
12. Désinscription d'un joueur

Actions permises par l'application : en option

1. Recherche d'un jeu

- parcours par type de machine ou par genre
- recherche d'un jeu par mots clés
- recherche d'un jeu par préfixe du nom

2. Affichage de statistiques (mode administrateur)

- classement des jeux les plus joués entre humains
- meilleurs joueurs (ou amis) par jeu
- ...

3. Évolution du profil des joueurs (mode administrateur)

- passage à l'âge adulte d'un joueur **enfant**
- passage profil **standard** ↔ **gold**

4. Implémentation d'une interface humain-machine évoluée

Aspects pratiques

► Réalisation en langage Java

- application des principes vu lors du module
- implémentation d'une interaction simple en mode console
- gestion appropriée des erreurs
- documentation du code (JavaDoc)



composition du rendu

- fichier archive (JAR) avec l'ensemble des sources (. java)
- compte-rendu de projet dans le document fourni (cf. eCampus)



rendu du projet : **23 avril** par email

- to : aurelien.max@universite-paris-saclay.fr
- titre : [Polytech > Et3] rendu projet Java NOM ETUDIANT.E