

UE21 – Informatique appliquée

Projet Java

HelHa
Pluquet Frédéric
2019-2020

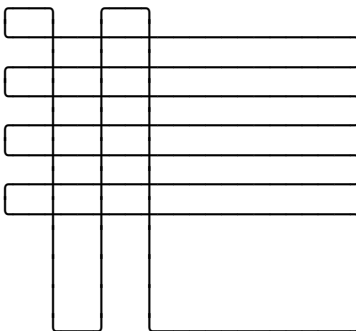
On vous demande de produire une application permettant de simuler des chemins de fer avec des trains qui roulent dessus. Elle sera composée de 3 programmes communiquant en réseau : un serveur, un affichage et une télécommande.

Une vidéo des programmes est disponible ici : <https://www.dropbox.com/s/gfzahkzr6tw4mqb/Projet.mp4?dl=0>

Le serveur

Quand le serveur démarre, il ouvre un fichier dont le nom est passé en paramètre du main (dans String[] args) afin de lire le tracé des chemins de fer et les trains s'y déplaçant. Le fichier est composé de la manière suivante :

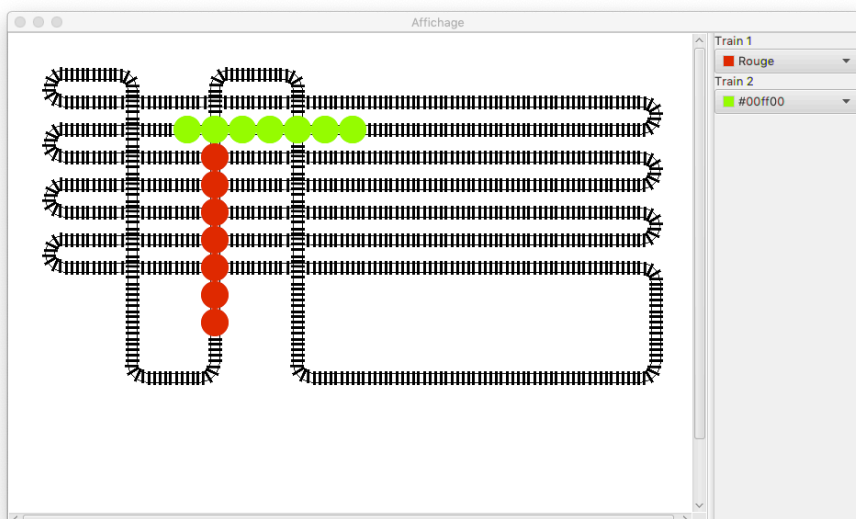
- une ligne avec la largeur **L** du terrain (20 par exemple)
- une ligne avec la hauteur **H** du terrain (30 par exemple)
- ensuite **H** lignes de **L** caractères UTF-8 qui définissent le terrain, par exemple :



- finalement, une ligne par train du type "X Y N P", dans laquelle :
 - X et Y est la position (basée en 0,0) qui où est placée la locomotive du train
 - N est le nombre de wagons (≥ 0 car la locomotive n'est pas comprise)
 - P est une lettre (T, L, B, R) qui définit la direction initiale du train (Top, Left, Bottom, Right).
 - Par exemple, 4 5 3 R représente un train de 3 wagons, avec la locomotive en (4,5) et qui se dirige vers la droite.

Une fois le fichier lu (et transformé en modèle orienté objet), le serveur attend les clients (affichage ou télécommande).

L'affichage



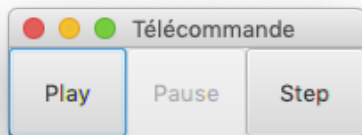
Le deuxième programme est un affichage de la carte qui devra ressembler à l'image ci-contre.

Le panneau de droite permet de choisir la couleur de chaque train (en fonction du nombre de trains présents sur la carte).

Nous devons pouvoir ouvrir plusieurs clients sur le même serveur. Les trains se déplaceront de la même manière partout. Attention : les trains ne rentrent pas en collision. Ils se mettent en arrêt s'ils se trouvent devant un train jusqu'au moment où cela devient libre.

Les clients peuvent se fermer (se déconnectant donc du serveur) tandis que d'autres restent ouverts.

La télécommande



La télécommande permet de mettre la simulation en marche (Play), en pause (Pause) et d'avancer pas à pas (Step). Tous les clients sont alors mis à jour comme il le faut.

Il peut y avoir plusieurs télécommandes connectées en même temps au serveur. Elles sont alors toutes notifiées et mises à jour lorsque l'état de la simulation change (en cours de simulation ou arrêtée).

Critères de cotation

- Il faut que la compilation se déroule sans problème.
- Toutes les fonctionnalités doivent être implémentées.
- Il faut bien gérer les exceptions (réseaux, fichiers, entrées utilisateurs, ...).
- Il faut que tous vos modèles soient testés complètement avec des tests d'unité.
- Il ne faut pas de complexité ou d'optimisation inutiles. Restez simple mais pas simpliste (par exemple, la duplication de code ne donne pas un code simple).
- Les interfaces graphiques doivent être les plus proches possibles de celles présentées dans la vidéo.
- Il ne faut pas de duplication dans la gestion des vues communes des programmes 1 et 3.
- Chaque classe et chaque méthode doivent avoir une responsabilité clairement définie.
- Il faut que vous soyez cohérent avec vos propres règles de codage ainsi que celles de Java.
- Chaque méthode ne doit pas faire plus d'une dizaine de lignes.
- Chaque classe et méthode doit être documentée en Javadoc.
- Veillez à une bonne séparation en modèle, vue et contrôleur.

Modalités de remise

Vous pouvez travailler individuellement ou par groupe de 2. Chaque groupe *reçoit* un repository GIT (envoyez-moi un mail avec votre nom (ou vos deux noms) et vos logins gitlab pour le recevoir) sur lequel vous committez régulièrement l'avancement de votre projet (au moins 1 fois par semaine). Votre repository doit contenir un fichier *README.md* qui contiendra l'avancement de votre application (ce qui a été fait) et le planning prévu pour terminer le projet. Ce document doit être mis à jour toutes les semaines (maximum le dimanche soir).

Le dernier commit avant le **15 mai 2020 à 23h59** sera considéré comme votre remise finale.

L'examen de laboratoire sera un ensemble de modifications à effectuer sur votre projet endéans un temps imparti.

Bon travail !