

FLAMME ROUGE PROJET POO 2 IUT CHARLEMAGNE

<u>Programme</u>:

- 1- Environnement de développement ;
 - Description des principales classes ;
 - Choix liés à la programmation ;
- 3- Difficultés ou anomalies rencontrées ;
- 4- Enfin, conclusion.

1. Environnement de développement

Je commence par la description de la base utilisée, le choix de l'environnement de travail ainsi que les étapes fondamentaux de la conception de ce projet.

Description de la base utilisée :

- Le système d'exploitation: Windows 7 Intégrale
- Langage de programmation: Java, version 1.8
- Environnement de développement: Notepad++
- Outil d'execution : cmd

Class Fil:

Définition des attributs de la classe à partir de l'énoncé de projet, des vidéos de simulation sur youtube dont je donnerais les liens dans la partie conclusion.

Il sont au nombre de (6) six, les attributs de la classe;
 Un constructeur avec arguments;
 Des accesseurs et quetteurs en fonction des attributs, ainsi que d'autres méthodes;

Class Case:

Formé de deux Fil (Gauche et Droite);

Il sont au nombre de (5) cinq, les attributs de la classe;
 Un constructeur avec arguments;
 Des accesseurs et guetteurs en fonction des attributs, ainsi que d'autres méthodes;

Class Circuit:

```
Formé de d'un ArrayList de Case;
```

Il sont au nombre de (4), les attributs de la classe;
 Un constructeur avec arguments;
 Des accesseurs et guetteurs en fonction des attributs, ainsi que d'autres méthodes;

Class Cycliste:

Il sont au nombre de (4), les attributs de la classe;
 2 ArrayList (Côté occupé et numéro de case occupés);
 Un constructeur avec arguments;
 Des accesseurs et guetteurs en fonction des attributs, ainsi que d'autres méthodes;

Class Joueur:

Ayant de deux Cyclistes (Rouleur, Splinteur);

Il sont au nombre de (7), les attributs de la classe;
 Un constructeur avec arguments;
 Des accesseurs et guetteurs en fonction des attributs, ainsi que d'autres méthodes;

Class Carte:

Il sont au nombre de (5), les attributs de la classe;
 Un constructeur avec arguments;
 Des accesseurs et guetteurs en fonction des attributs, ainsi que d'autres méthodes;

Class Plateau:

Ayant de deux Cyclistes (Rouleur, Splinteur);

- Il sont au nombre de (16), les attributs de la classe;

Un constructeur avec arguments;

Des accesseurs et guetteurs en fonction des attributs, ainsi que d'autres méthodes ;

Stack < Carte > cartesVisibleS

Stack < Carte > cartesNonVisibleR

List < Stack < Carte >> emplacementNonJoueeSplinteur;

ArrayList < Carte > list4CartePiocheSplinteur;

Sont des exemples de arraysList carte utilisées pour implementé le plateau de carte ;

J'ai défini un plateau comme étant une SDD à 2 emplacements : Les cartes jouées et non jouées;

L'emplacement des cartes jouées, ne pourrait contenir qu'une carte, celle jouées;

L'emplacement des cartes non jouées à 2 piles (celles des cartes piochées et remis et celles des non piochées

d'abord);

Class Jouer et Jeu:

Ayant de nombreux attributs et méthodes :

Jeu : elle contient des méthodes créer pour être utilisées dans Jouer, ou pour modéliser les actions du jeu. Elle utilisent les méthodes, accesseurs et guetteurs des autres classes pour fonctionner.

Jouer: elle contient certes certaines petites méthodes ou instructions regroupées, mais en majorité son rôle est faire Jeu.

Nb: Pour plus d'infos sur le jeu, je vous propose de lancer l'application sur la console de windows et d'accéder au menu numéro 0;

Class Test:

Pour chaque classes Fil, case, ... une classe test à été developpé pour voir si les méthodes donnaient le résultat attendu; elles contiennent 2 méthodes chacunes : l'une pour le constructeur et l'autre pour les autres méthodes;

Difficultés ou anomalies rencontrées

En général, pas grand chose,

- En premier, J'ai perdu quelques jours sur la modélisation, la machine à état du projet. La machine à état du projet, m'a beaucoup plus ouvert les "jeux" sur la façon de faire fonctionner le code;
- En second lieu, une fois que j'ai commencé à coder, j'ai pas eu de difficultées majeures, sauf celles liées à l'écriture des données sur le fichier externes, oui j'ai fais un fichier texte qui fera le log de toute une session de jeu;
- En conclusion, j'ai beaucoup appris sur le projet, la manipulation des collections, la gestion des exceptions ainsi que les mécanismes d'IO.

Détail de jeu

Comment jouer ?

- dézipper le dossier du projet
- ouvrir l'invite de commande cmd dans le dossier
- recompiler : javac *.java
- exécuter : java Jouer et c'est Ok !

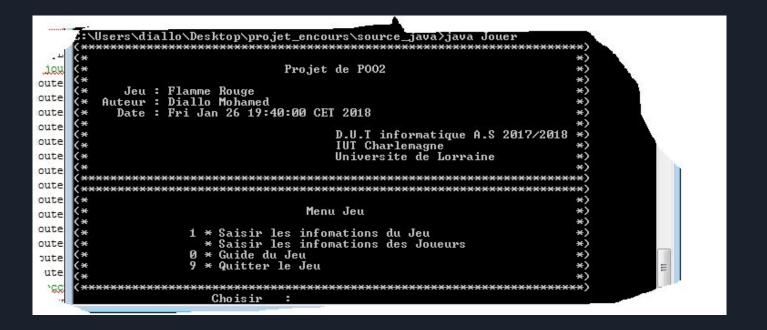
A la fin d'une session de jeu, vous pouvez consulter le log(actions, choix, résumé du jeu), en ouvrant le fichier sauvegarde;

Nb: Pour le fichier sauvegarde et guide, dans mon code j'utilise un chemin vers ce fichier, donc si vous n'avez le même chemin, ce qui risque d'être le cas, il faudra modifier le fichier Jouer.java pour mettre votre propre chemin.

Capture d'écran

```
C:\Users\diallo\Desktop\projet_encours\source_java>javac *.java
Note: Jouer.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

C:\Users\diallo\Desktop\projet_encours\source_java>
```



Capture d'écran

```
Bienvenu dans le Guide de Flamme Rouge !

FLAMME ROUGE

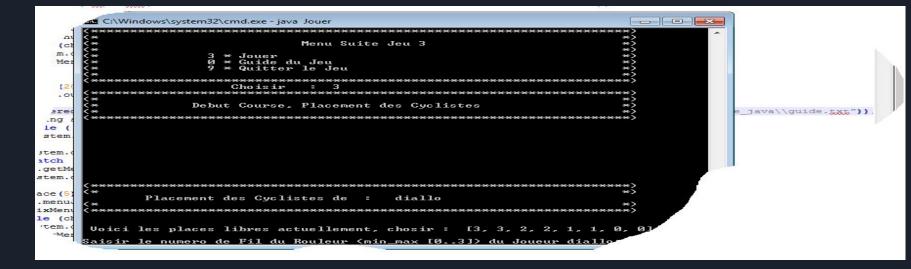
.e monde du cyclisne, la flamme rouge est un drapeau qui indique le dernier letre: la dernière ligne droite, le moment où il faut tout donner!

Rouge est un jeu de cours rapide et tactique où chaque joueur contrôle un de deux coureurs: un rouleur et un sprinteur. Le mais le jeu est le premi sur à franchir la ligne d'arrivée avec un de ses coureurs. Les joueurs dép le un tevancé. Profitez de l'aspiration des scoureurs, négociez vos mais autention aux coups de fatigue!

Sont expliquées en 5 minutes et l'immersion est totale. Qui inscrira ns l'histoire de la grande course ?

RèGLES SPECIFIQUE A CETTE VERSION

Le vous proposez, vous avez tous les débuts :
```



Capture d'écran

```
C:\Windows\system32\cmd.exe - java Jouer
                                                                                                               0 0
 **
           Splinteur > S Couleur VERT, Position : 66, Cote : G
   rafr
   @par
   @par
public
 for (J
  if (c
                                           Classement Final
   if
                                                                                                                                ur().get(
    c2.
   if
                                                                                                                                ur().get(c
    c1
           doal, couleur: ROUGE, score: 141
Rouleur > R Couleur ROUGE, Position: 75, Cote: G
Splinteur > S Couleur ROUGE, Position: 66, Cote: G
            rmp, couleur : VERT, score : 140
           Rouleur > R Couleur UERT, Position : 74, Cote : G
Splinteur > S Couleur UERT, Position : 66, Cote : G
           Il a fallut 27 etapes, pour faire la course ! Fã@licitation
```

Merci!

Nb: Pour le fichier sauvegarde et guide, dans mon code j'utilise un chemin vers ce fichier, donc si vous n'avez le même chemin, ce qui risque d'être le cas, il faudra modifier le fichier Jouer.java pour mettre votre propre chemin, sinon ça risque de ne pas marcher!