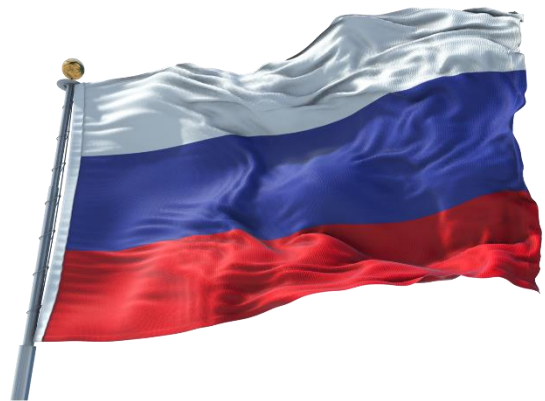


# Rapport Projet Java

## Jeu de roulette Russe

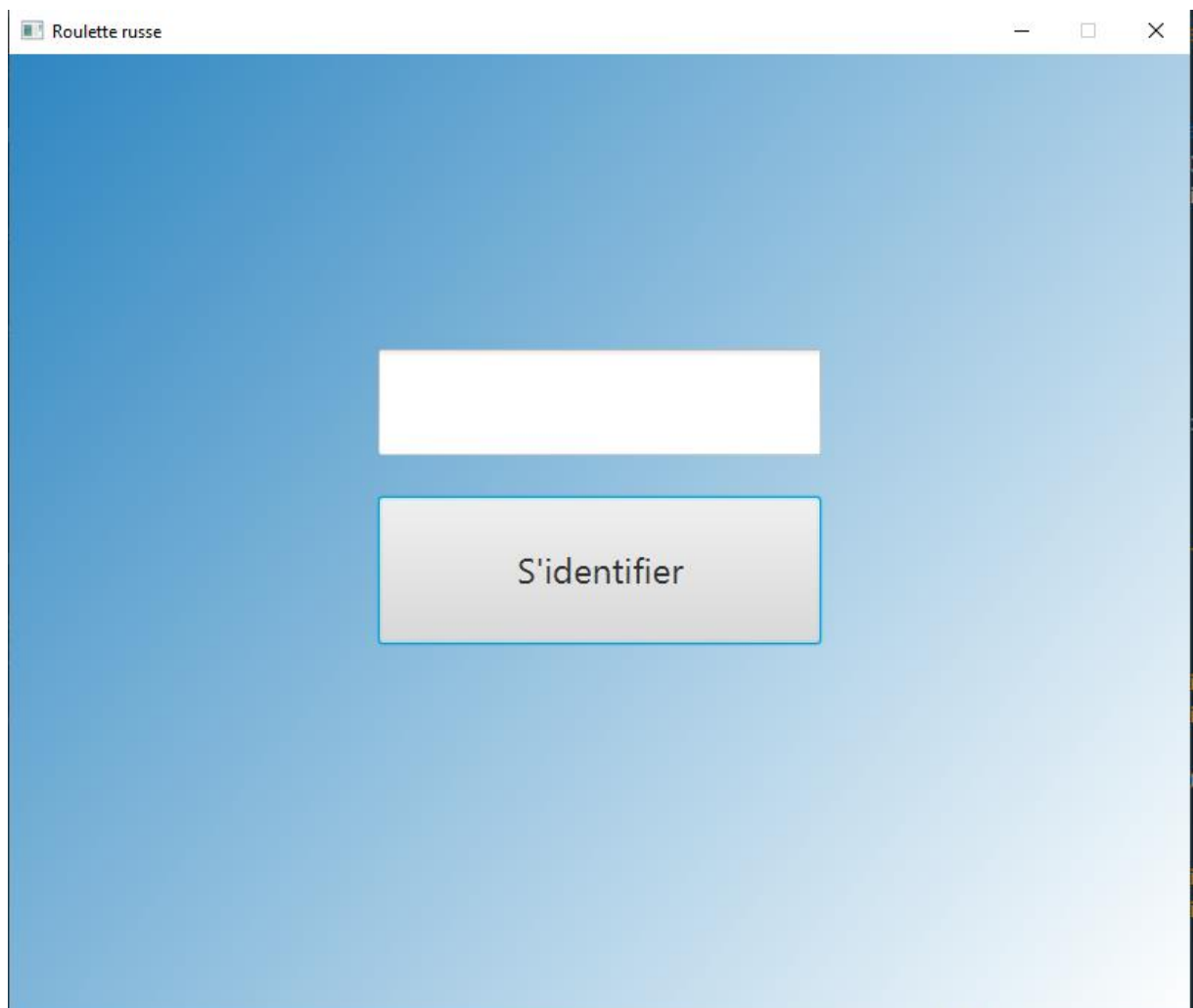


# I – Présentations des fonctionnalités

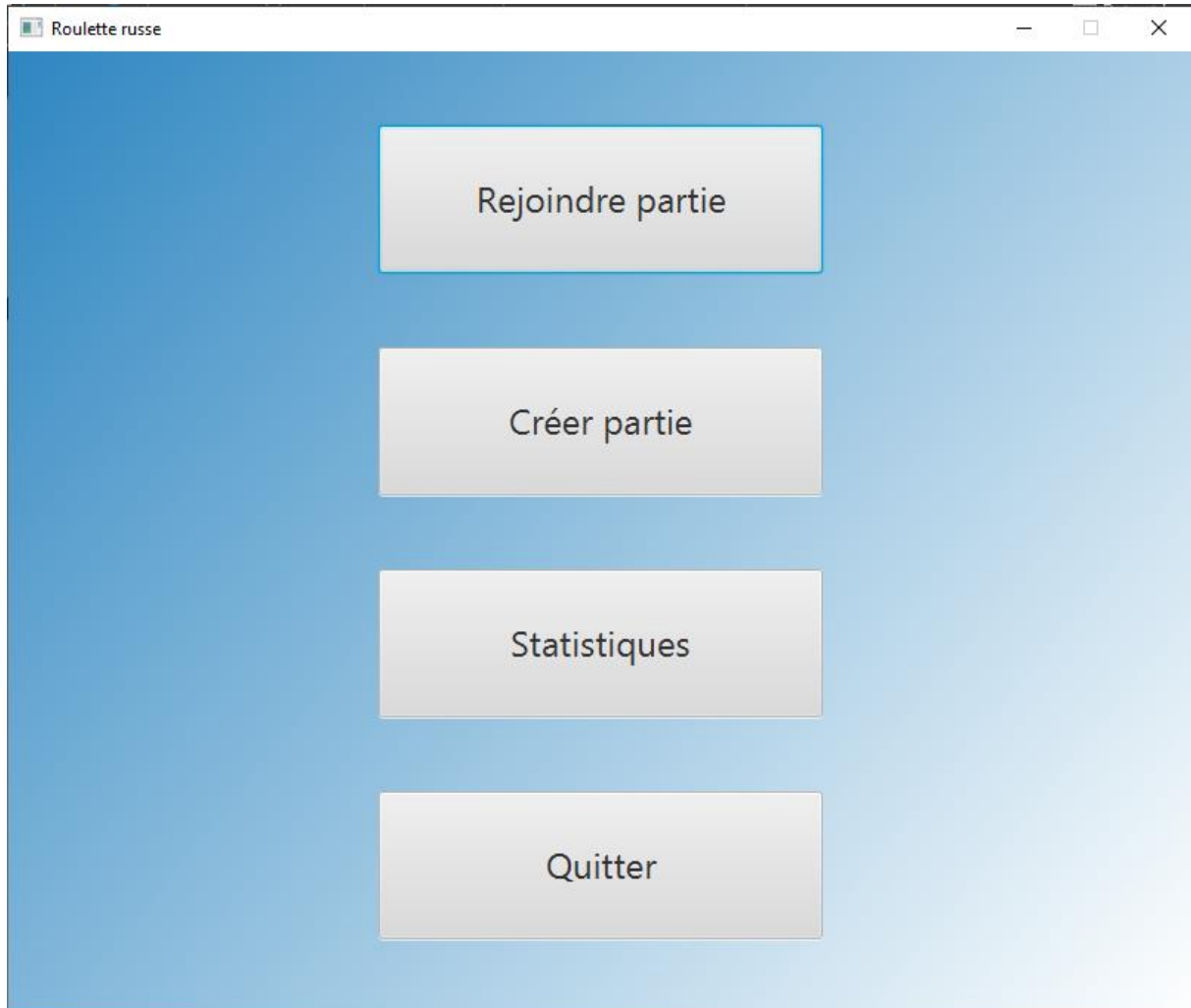
Voici le déroulement d'une partie :

*Joueur1*

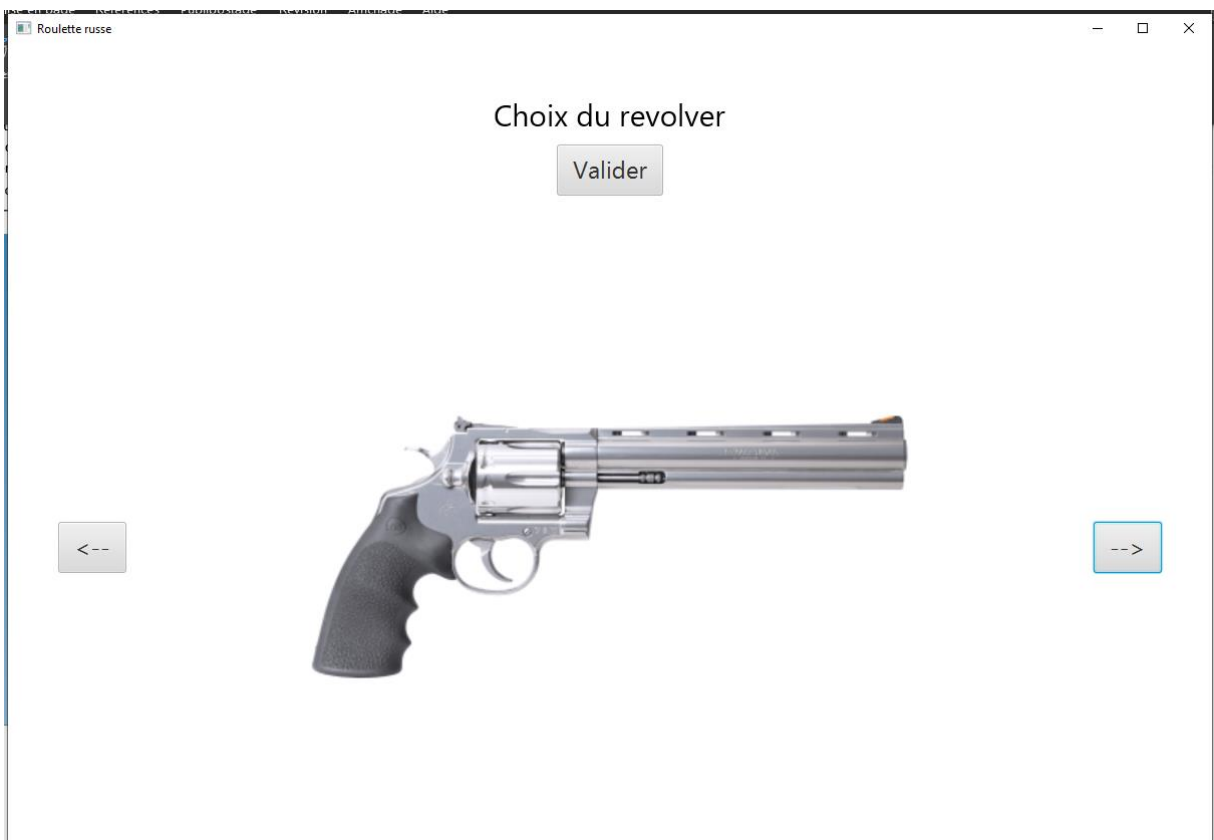
1. Lancement de l'application
2. Le joueur s'identifie en tapant son nom
  - a. S'il existe, on récupère ses précédents scores
  - b. Sinon, on crée le joueur dans la BDD



3. Choix dans le menu
  - a. Créer une partie
  - b. Rejoindre une partie
  - c. Quitter
4. Le joueur choisie de créer une partie

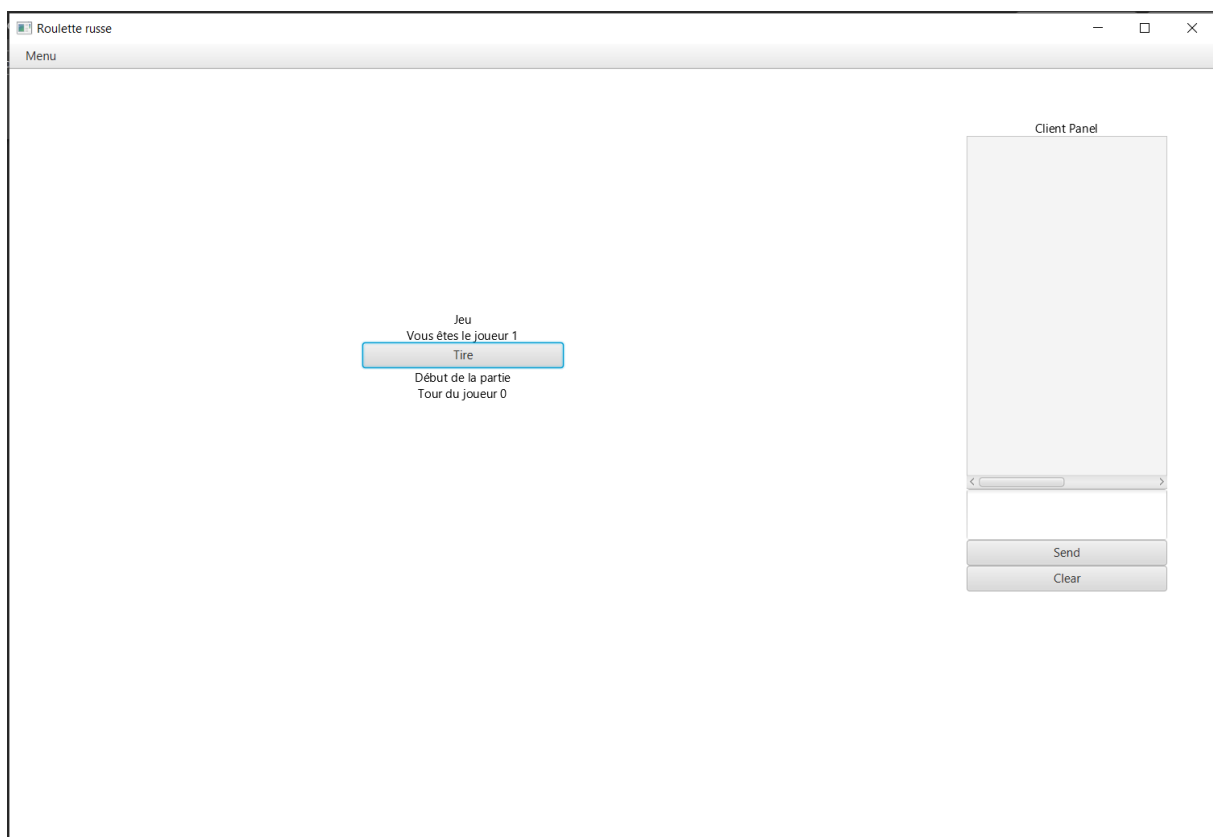


5. Le joueur attend que le second rejoigne la partie avant de commencer le paramétrage.
6. Le joueur paramètre la partie
  - a. Indicateur de difficulté : La taille du barillet/Modèle de Revolver : 8-6-5



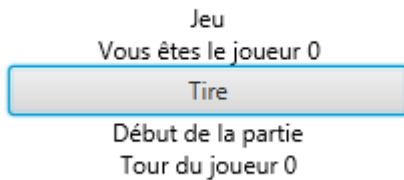
## *Joueur 2*

1. Lancement de l'application
2. Le joueur s'identifie en tapant son nom
  - a. S'il existe, on récupère ses précédents scores
  - b. Sinon, on crée le joueur dans la BDD
3. Choix dans le menu
  - a. Créer une partie
  - b. Rejoindre une partie
  - c. Quitter
4. Le joueur choisie de rejoindre une partie
5. Le joueur attend que l'hôte finalise les paramètres de la partie

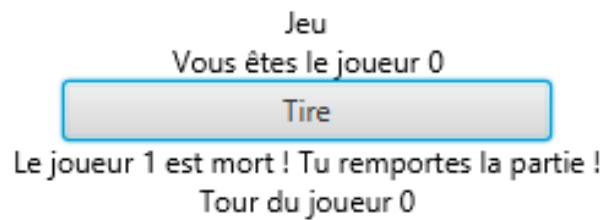
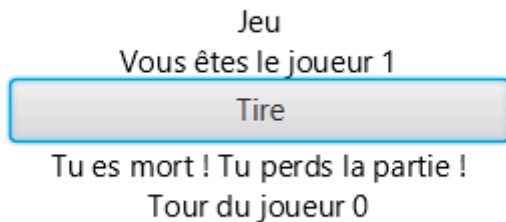


*Commun*

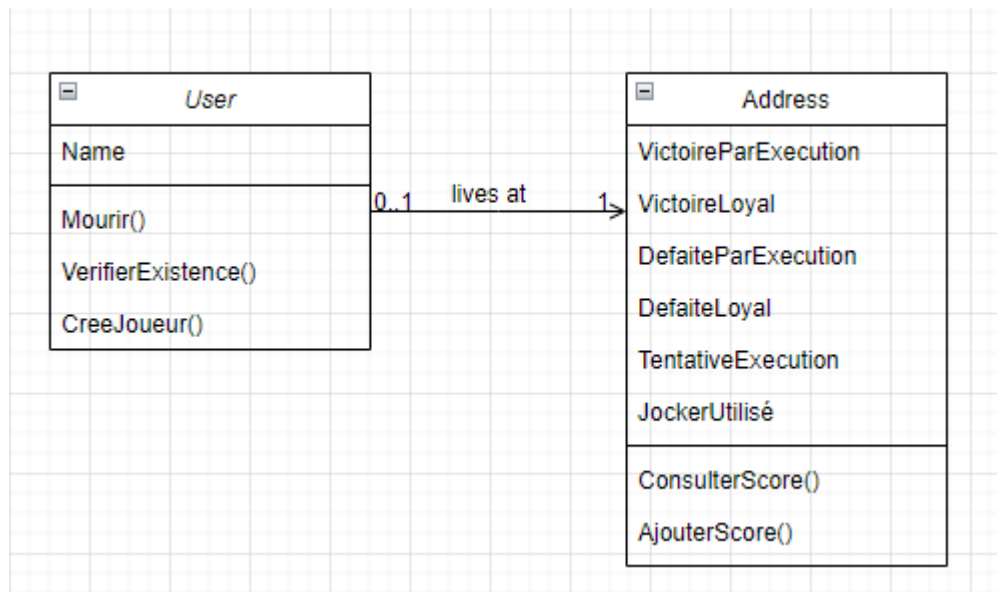
1. Lancement de la partie
2. L'interface graphique indique à qui est le tour de tirer.



3. 2 choix d'action possible
  - a. Se tirer dessus
    - i. Cas 1 : \*click, la partie continue et on passe à l'autre joueur
    - ii. Cas 2 : \*BANG, la partie est terminée et l'autre joueur gagne
  - b. Utiliser le chat
4. Après la mort d'un joueur, affichage du message de défaite ou de victoire



## II – Architecture générale de l'application



Technologies utilisées :

Pour ce projet les technologies à utiliser nous sont déjà imposées. Nous allons utiliser le langage Java, avec la librairie JavaFX pour l'interface graphique, ainsi qu'une librairie de socket pour la partie réseau. L'IDE sera IntelliJ ou Eclipse selon les préférences des membres de l'équipe. La base des données sera administrée avec MySQL.

GitHub sera utilisé pour sauvegarder à distance le projet et effectuer la gestion des versions.

# III – Gestion de projet

Bon Coté:

- Gains de Compétence en Java
- Sujet Intéressant
- Gains de Compétence en Gestion de Projet

Mauvais Coté:

- Projet un poil trop ambitieux sur certains points notamment les animations
- Mauvaise gestion du Temps
- Certaines fonctionnalités initiales avortés

## **Introspection Yann :**

Pour moi ce projet était une opportunité pour développer mes capacités en java.

Pourtant je ne pense pas les avoirs tant que ça développer puisque j'ai le sentiment que le côté « graphique » du projet nous a beaucoup limité. En effet j'ai passé la plupart du temps à débbugger des problèmes lié à l'interface graphique en laissant de côté la partie algorithmique et architecturale. De plus je ne pense pas que JavaFX soit la librairie idéale de nos jours pour réaliser des jeux. Elle est très limité au niveau animations et graphismes. Nous aurions du nous orienter sur un autre type d'application plutôt qu'un jeu.

## **Introspection Mustapha :**

Étant un débutant total en java, il m'a été assez difficile d'appréhender le projet dans ces débuts. Le cahier des charges n'a pas posé de problème mais la programmation a été un exercice assez ardu. Pour combler cela, les premières ligne et grand axe ont été réaliser en Peer Programing avec Yann ce qui m'a donné les grandes lignes et des bases solides pour continuer ma montée en compétence et l'avancement de mes parties du projet. Ce fut une expérience plus que concrète avec un objectif clair et c'est ce qui m'a plu. Enfin, les nombreux problèmes liés au logiciel avec JavaFX et MySQL server nous à fais perdre pas mal de temps en classe qu'on aurait mieux utilisé en programmation.