

SHANNON PONG :

Présentation :

Ce jeu est une évolution du jeu d'arcade pong, paru en 1972. Le projet est configuré avec Gradle et en utilisant JavaFX.

Le jeu est inspiré du tennis de table en vue de dessus, et chaque joueur s'affronte en déplaçant la raquette virtuelle de haut en bas, de façon à garder la balle dans le terrain de jeu.

Dans notre projet, vous trouverez de multiples améliorations, comme par exemple la possibilité de jouer en multijoueur à 4, en deux contre deux. Si vous jouez seuls, vous aurez le choix entre essayer de battre l'ordinateur ou tenter de battre votre propre record dans le mode infini !

Excepté dans le mode infini, les parties se jouent sous forme de manches, le joueur choisit au préalable le nombre de manches qu'il souhaite effectuer. Chaque manche se joue en 3 points. Sur le jeu, il est affiché pour chaque joueur le nombre de manches qu'il a remporté sur le nombre de manches totales.

Un dressing est également disponible dans le menu, vous pouvez ici modifier la couleur de votre balle, des raquettes et du terrain comme bon vous semble.

Les paramètres vous permettront de gérer les effets sonores et la musique, ainsi que de changer toutes les touches du jeu à votre guise.

Enfin, notre pong améliorer vous propose de jouer une partie classique avec des pouvoirs ! Le fonctionnement des pouvoirs se fait à l'aide d'une jauge attribuée à chaque joueur, lorsque celle-ci se remplit, le joueur a la possibilité de prendre un bonus, ou d'envoyer un malus à son adversaire.

Vous trouverez comme bonus l'agrandissement de la raquette, l'accélération de la raquette, le pouvoir d'inverser la direction de la balle et un smash. En ce qui concerne les malus, vous aurez la possibilité de rendre la balle invisible, de rétrécir la raquette de votre adversaire ou de la ralentir, et aussi de lui inverser ses commandes !

Les bonus sont affichés en rouge et les malus en bleus voici un tableau répertoriant les lettres associées à chaque bonus/malus :

BONUS	MALUS
accélération de la raquette : B	balle invisible : I
agrandissement de la raquette : G	rapetisser la raquette : P
smash : S	ralentir la raquette : S
inverser la direction de la balle : R	commandes adverses inversées : R

Télécharger pong :

Pour télécharger notre pong et même participer à son développement, rien de plus simple, il vous suffit de cloner le dépôt sur votre machine, pour cela, il suffit de cliquer sur le bouton 'clone' de GitLab et de choisir la manière dont vous voulez cloner le projet, nous vous conseillons en https.

Execution, compilation:

Pour compiler, il suffit d'exécuter, depuis le répertoire pong :

```
$ ./gradlew build
```

Pour exécuter, il suffit d'exécuter, depuis le répertoire pong :

```
$ ./gradlew run
```

Jouer :

Pour jouer, voici un tableau répertoriant les touches par défaut pour chaque action :

joueur\action	HAUT	BAS	DÉFENSE	ATTAQUE
JOUEUR 1	Z	S	D	Q
JOUEUR 2	UP	DOWN	LEFT	RIGHT
JOUEUR 3	CTRL	ALT		
JOUEUR 4	O	L		

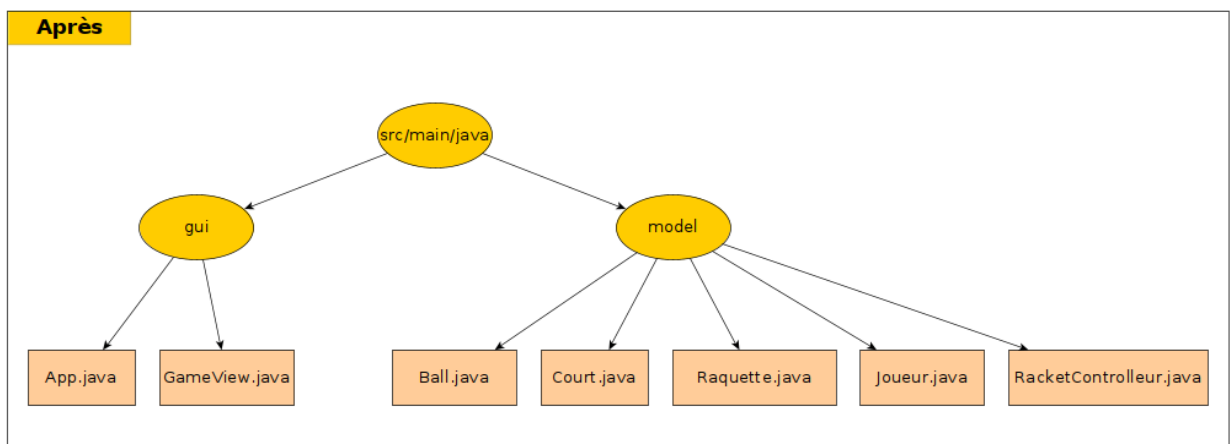
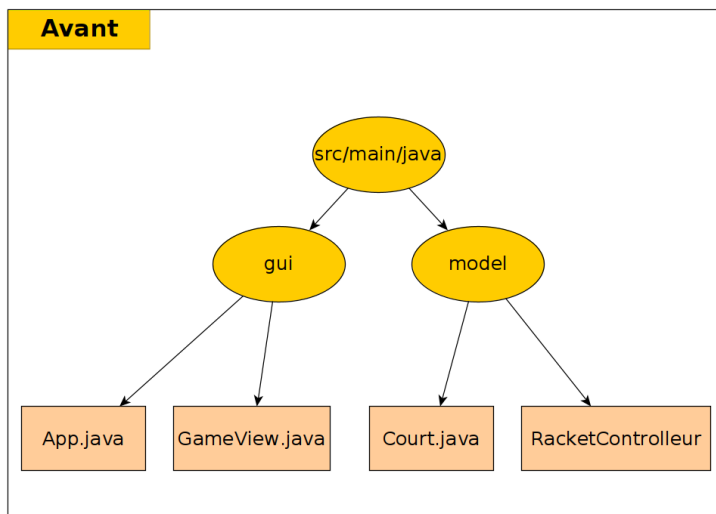
Pour jouer en mode classique avec les pouvoirs, il faut les activer lors du choix du nombre de manches.

Vidéo :

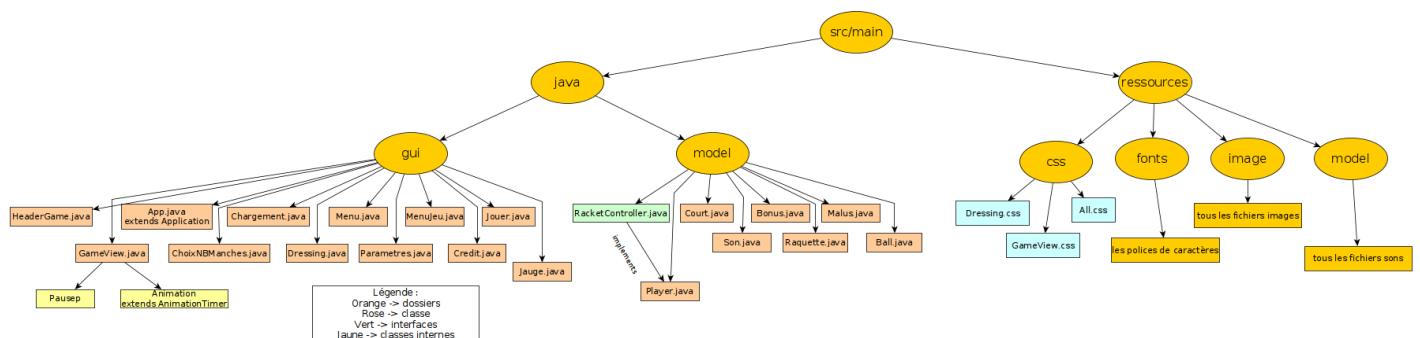
Lien vers notre vidéo finale de fin de projet :

<https://youtu.be/NZch0f1YdjY>

Graph : voici les graphes avant/après de la factorisation



Voici le graphe de notre arborescence finale :



Organisation des Sprints :

1er Sprint

- Utilisation de git + compte rendu de commandes (Manel)
- Compréhension du code + commentaire (Ronen)
- Compréhension du code + compte rendu (Georges)
- Compréhension du code + correction de Georges et Ronen (Antoinette)
- Maquette du jeu (Georges) ? (si on peut la voir en vidéo ça peut être cool)
- Comprendre et se documenter sur JavaFx pour faire plus l'interface graphique (Alec et Léa)
- Bot qui suit la balle (Anaïs)

2ème Sprint

- Son (Ronen)
- Refactoring (Antoinette)
- Manel et Anaïs classe raquette
- Georges classe balle
- Ronen la classe player
- Faire un menu (Léa)
- menu paramètres (Georges)
- Interface (affichage : joueurs, barre de délimitation, gagnant) (Manel)

3ème Sprint

- musique (Ronen)
- faire les crédits (Léa)
- lié le menu aux scènes (Alec)
- logo de la team (Manel)
- changement du temps de réaction aléatoire du bot (Anaïs)
- Paramètres de balle, vitesse, direction, et en fonction de qui a marqué (Georges)

4ème Sprint

- model mvc finalement abandonné (Alec)
- bouton pour jouer contre le bot (Anaïs)
- Léa Création fichier CSS
- menu pause (Manel)
- mode infini (Antoinette)
- mode 2v2 (Georges)

5ème Sprint

- jauges bonus-malus (Georges)
- Création dressing (Manel)
- CSS de partout - Amélioration de l'interface (Léa)
- fin de partie (choisir le nombre de manches) Manel
- interface des paramètres (Alec)
- Organisation de fin de projet : scénario de la vidéo, Contributing.md, triage des issues (Antoinette)