

# Rapport de projet Pac-man

Touches de jeu :

<b>Touche du clavier</b>	<b>Action</b>
Flèches directionnelles	Déplacer le Pac-Man
Espace	Réinitialiser le jeu
Maj	Minimiser la fenêtre
H	Couper/relancer la musique
Échap	Quitter le jeu
<b>Touche de la souris</b>	<b>Action</b>
Bouton droit/gauche+déplacement	Déplace la fenêtre
Bouton central	Centre la fenêtre sur l'écran
Bouton gauche(sur un objet cliquable)	Le sélectionne et exécute son action

## I - Liste des fonctionnalités et extensions :

### **Fonctionnalités:**

- Déplacement automatique du Pac-man

Lorsqu'on donne une direction au Pac-man avec les flèches directionnelles, il va se déplacer automatiquement dans celle-ci jusqu'à ce qu'il rencontre un obstacle.

- Déplacement semi aléatoire des fantômes

Chaque fantôme choisit aléatoirement une direction. Une fois cette direction choisie, ils sont incapables de faire demi-tour. Ceci leur permet d'avoir un déplacement plus harmonieux.

- Changement de couleur après l'amassage d'un Super Pac-gomme

Le jeu vire au bleu après qu'on ait pris une Super Pac-gomme. Lorsque l'effet va s'estomper, l'écran se met à clignoter puis le jeu revient à la normale.

- Ajout d'un menu

Le menu permet de lancer le jeu ou de consulter le leaderboard.

- Ajout de musique

Pendant que l'on joue, la musique classique de Pac-man tourne en fond. Elle peut devenir quelque peu répétitive, c'est pourquoi il est possible de la couper avec la touche H.

### **Extensions:**

- Système de high-score

Nous avons implémenté un système de classement qui permet au joueur, lorsqu'il a gagné, d'entrer son nom pour pouvoir afficher son score dans le leaderboard. Avant de lancer une partie il est possible d'accéder au leaderboard, grâce au menu, afin de consulter les scores.

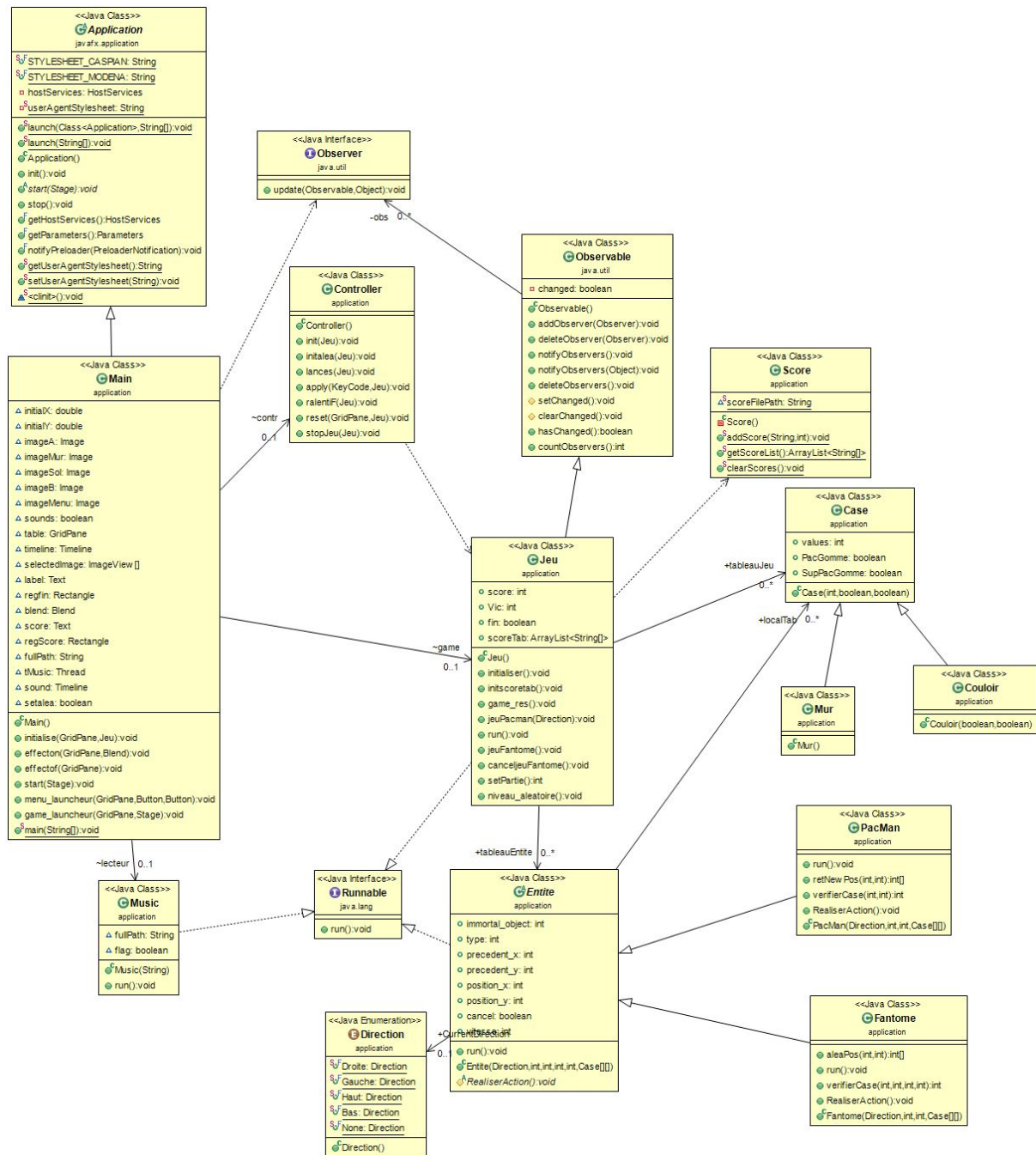
**~4h**

- Génération de carte aléatoire

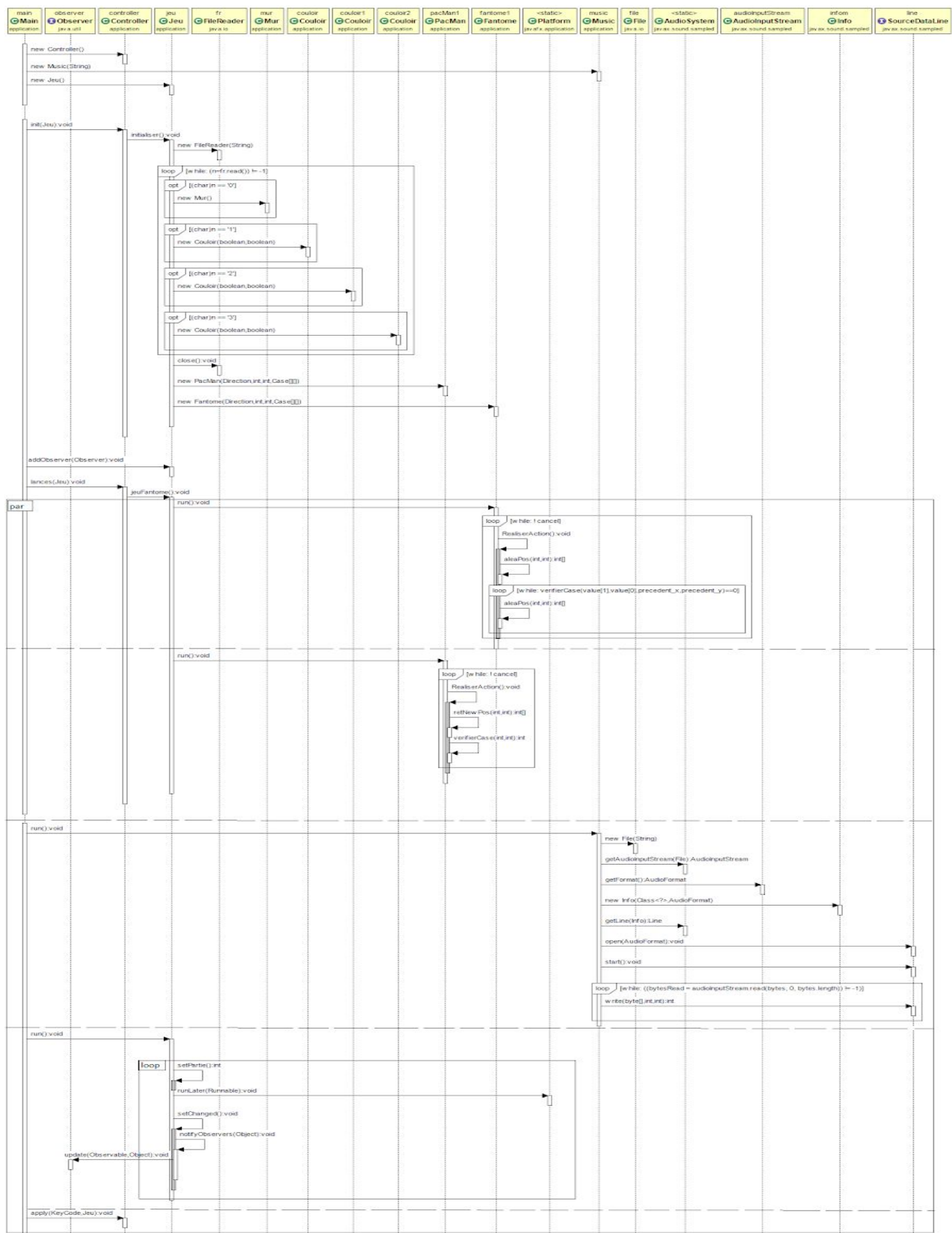
Avant de lancer une partie, le joueur a la possibilité de choisir s'il veut ou non que la carte soit générée aléatoirement au lieu d'avoir le niveau classique. Attention cependant car les cartes générées sont souvent beaucoup plus difficiles que celle de base. Attention, il faut restart la première partie pour que ça soit aléatoire.

**~1h30.**

## II.1) Diagramme des classes



## II.2) diagramme de séquence d'une partie avec un seul fantôme



### III - Justification de l'analyse :


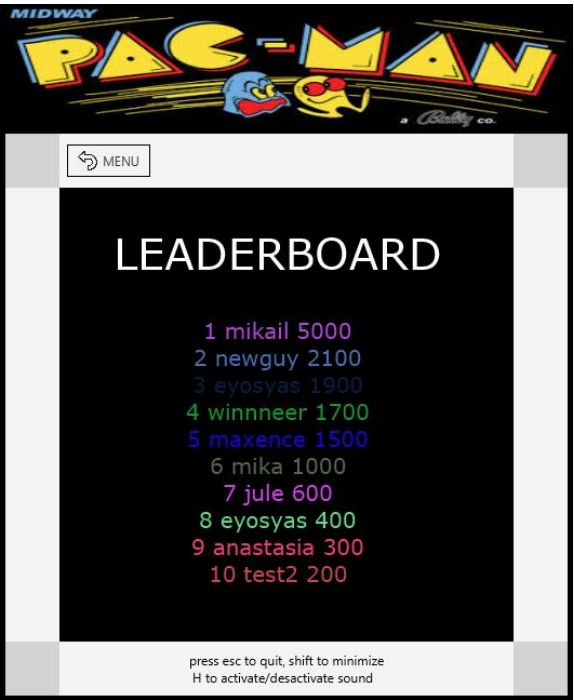
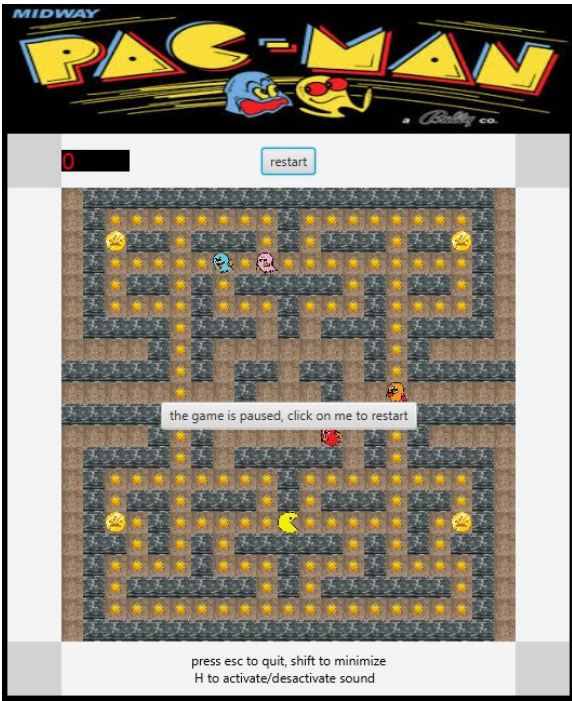
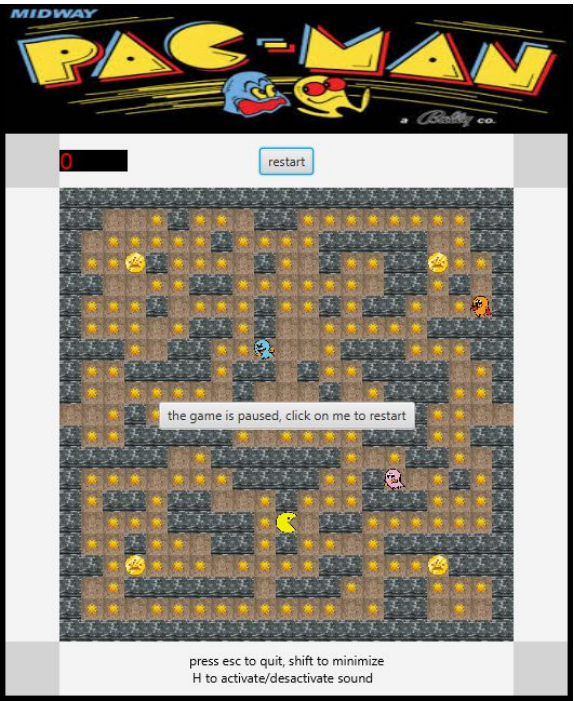
Afin de respecter le modèle MVC Strict, nous avons dû faire certains choix de modélisation précis.

Pour commencer, nous avons 2 parties distinctes qui sont le Modèle et la Vue. Ces parties, afin de communiquer entre elles, passent uniquement par le biais d'un contrôleur, qui est, pour des raisons de simplicité, stocké dans la Vue.

Il nous a été nécessaire de créer deux grilles correspondant à la surface de jeu. Une pour le Modèle et une pour la Vue. Ceci était indispensable afin de respecter le MVC.

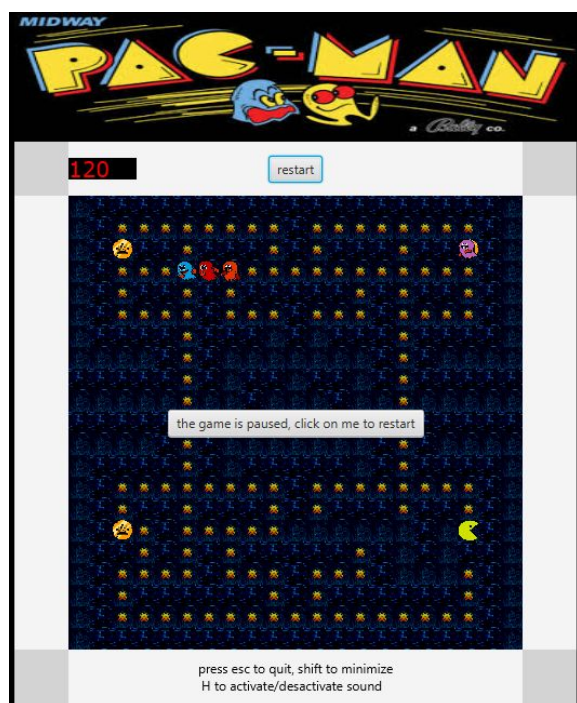
#### IV - Captures d'écran :

Voici des captures d'écran pour chaque cas notable de jeu.

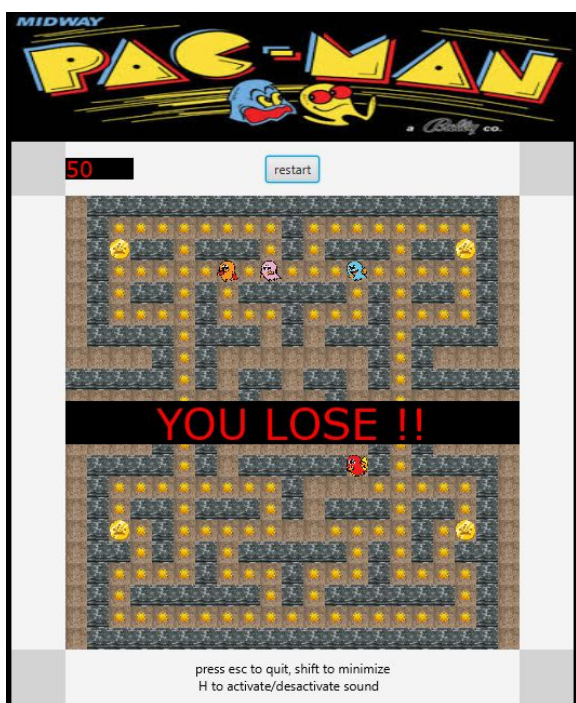
Menus	LeaderBoard
	
Jeu carte de Base	Jeu carte aléatoire
	



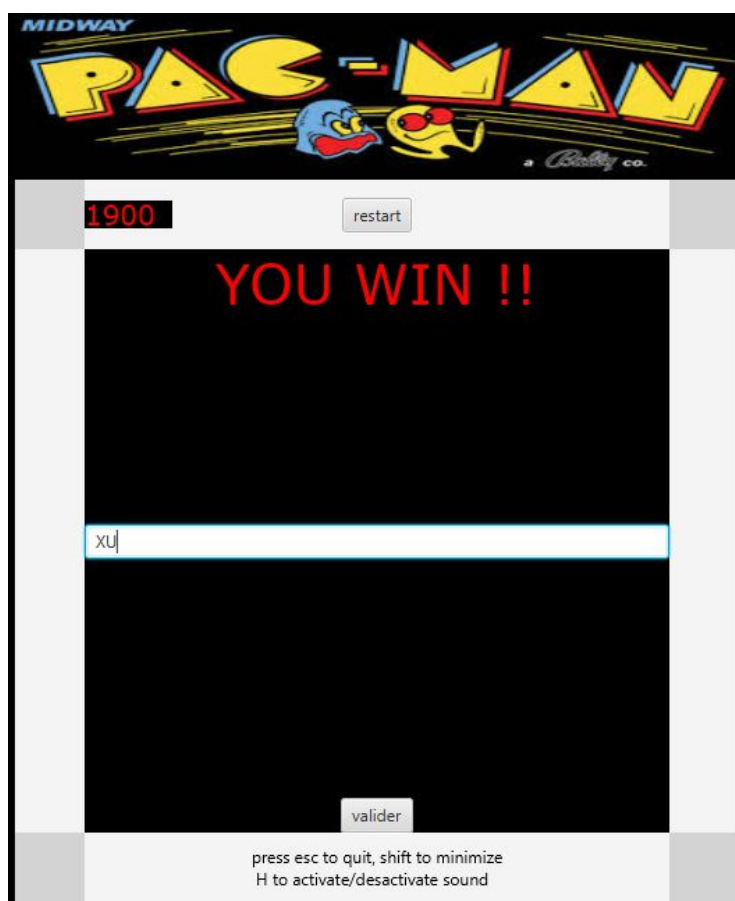
## SuperPacGomme



## Mort



## Victoire et insertion du nom



## V - Conclusion :

La réalisation de ce projet a été un bon moyen pour fortifier nos connaissances en Java. De plus, nous avons appris à mettre en place un pattern MVC afin d'avoir une modélisation la plus orientée objet possible. Enfin, nous avons perfectionné notre maîtrise des différents diagrammes, et plus précisément du diagramme des classes et de séquence.