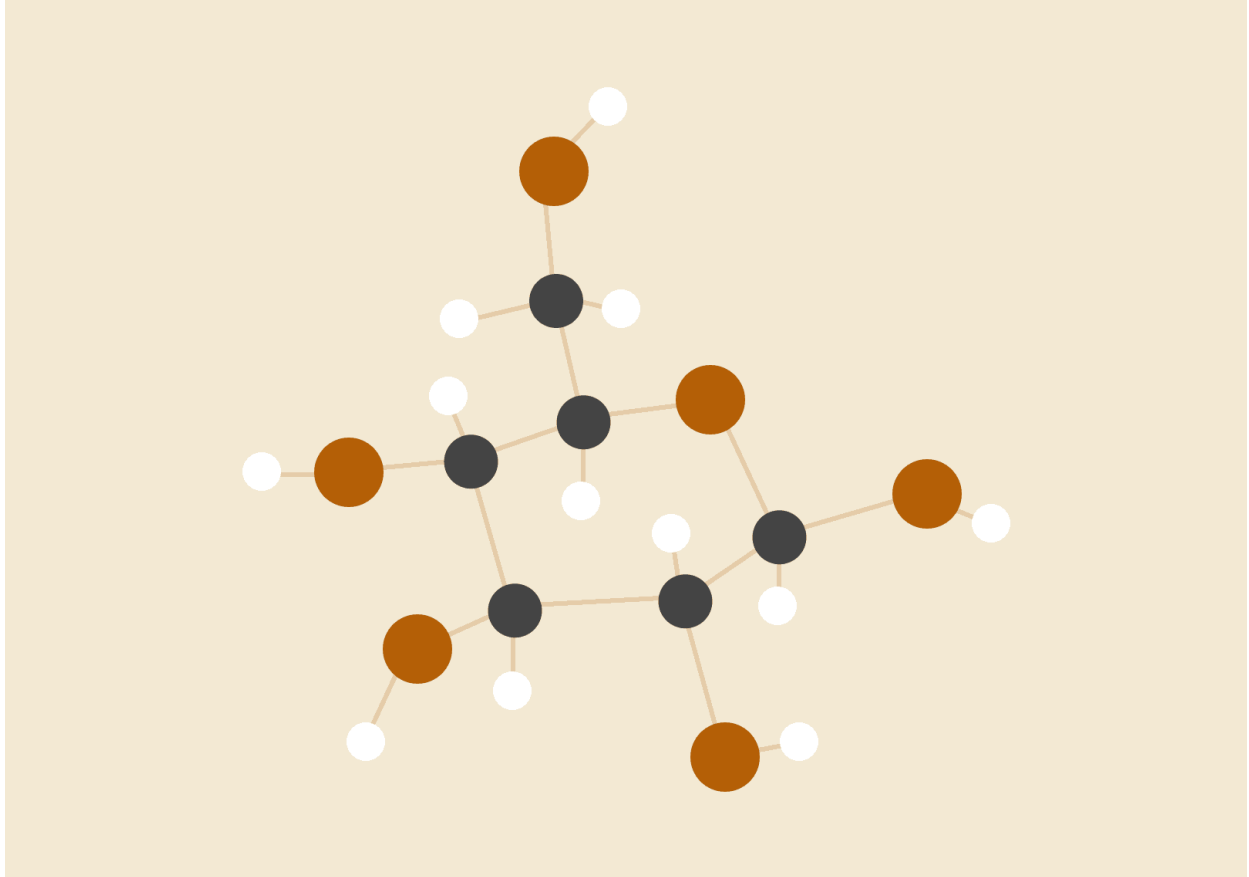


# RAPPORT : PROJET DOO JAVA

*Rapport de projet sur la réalisation d'un memory*



**PAR : Soler Florian & Thomas Robson**

15/06/2023

IaI - 1

# SOMMAIRE

SOMMAIRE.....	1
INTRODUCTION.....	2
DÉVELOPPEMENT.....	3
Préparation.....	3
Outils.....	4
Diagramme.....	5
Fonctionnement.....	6
RÉSULTATS.....	9
CONCLUSION.....	10
RÉFÉRENCES.....	11

## INTRODUCTION

Le projet consiste à créer un jeu de paires en Java (orienté objet), destiné à deux joueurs. Le jeu suit les règles classiques du jeu “Memory”, où les joueurs doivent trouver des paires de cartes identiques.

L'application comprend une fenêtre de paramétrage où les joueurs peuvent saisir leurs pseudonymes, choisir parmi différents thèmes de jeu (thème des cartes) et sélectionner la taille du plateau de jeu parmi plusieurs options. Une fois les paramètres définis, une fenêtre de jeu s'ouvre, affichant le plateau, les pseudonymes des joueurs et le score des joueurs. Les joueurs jouent à tour de rôle, et la partie se termine lorsque toutes les paires ont été trouvées ou lorsque les joueurs décident de l'arrêter.

Le projet a pour but de mettre en pratique les savoir enseignés durant les cours de développement et de suivre un “design paterne” adapté à notre logiciel.

## DÉVELOPPEMENT

### Préparation

Nous avons commencé par prendre connaissance des différentes fonctionnalités et éléments qui constituent notre jeu, il y a deux fenêtres principales :

1. Fenêtre de paramétrage : Cette fenêtre permet aux joueurs de saisir les paramètres du jeu. Elle comprend les éléments suivants :
  - Pseudonyme du joueur 1.
  - Pseudonyme du joueur 2.
  - Thème du jeu (par exemple : drapeaux de pays, animaux...).
  - Taille du plateau de jeu (par exemple : 4x3, 4x4, 5x4, 6x5, 6x6, 7x6).
  
2. Fenêtre de partie de jeu : Cette fenêtre affiche le plateau de jeu ainsi que le score des joueurs :
  - Cartouche du joueur 1 : Affiche le pseudonyme du joueur un et le nombre de paires trouvées.
  - Cartouche du joueur 2 : Affiche le pseudonyme du joueur deux et le nombre de paires trouvées.
  - Mise en évidence du cartouche du joueur en train de jouer.
  - Plateau de jeu : Affiche les cartes retournées ou non.

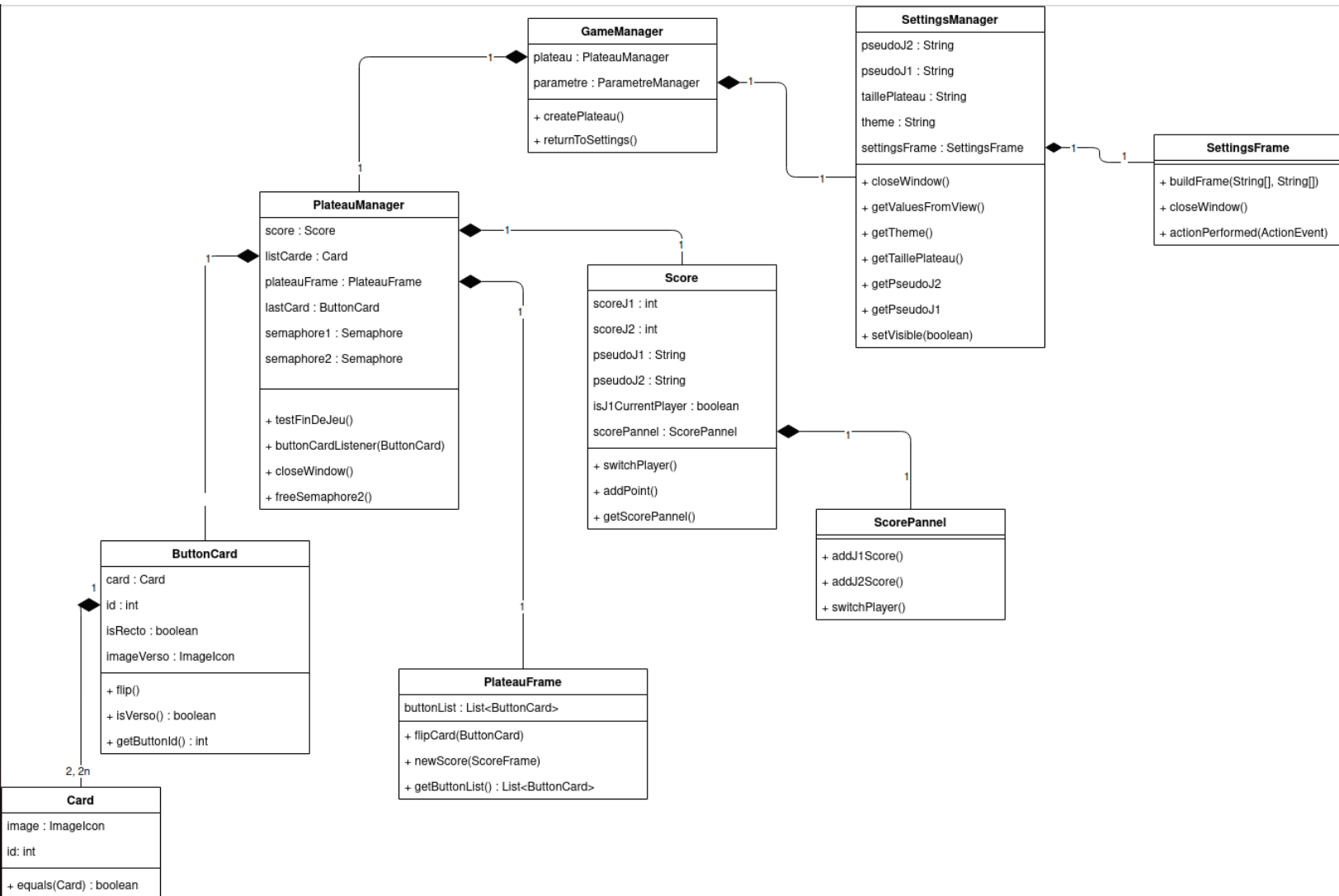
## Outils

Pour réaliser ce projet, nous avons dû utiliser une approche orientée objet, c'est pour cela que le diagramme de classe nous a été vraiment utile, car il nous permettait de suivre la structure que nous avions choisie.

Pour la réalisation du projet, nous avons travaillé avec Visual studio code pour garder une certaine liberté et, puisque nous connaissions déjà cet environnement de travail, ce qui n'était pas le cas avec éclipse ou encore IntelliJ.

## Diagramme

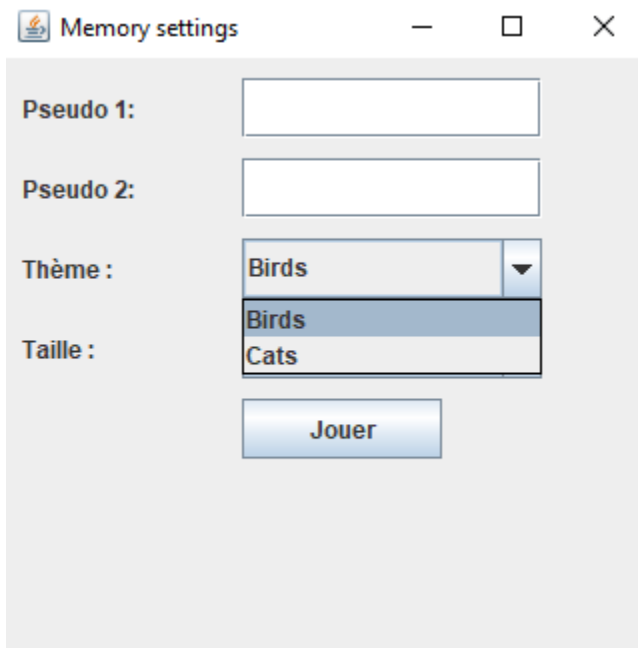
Voici le diagramme de classe qui représente l'architecture de notre logiciel et l'interaction des différentes classes entre elles.



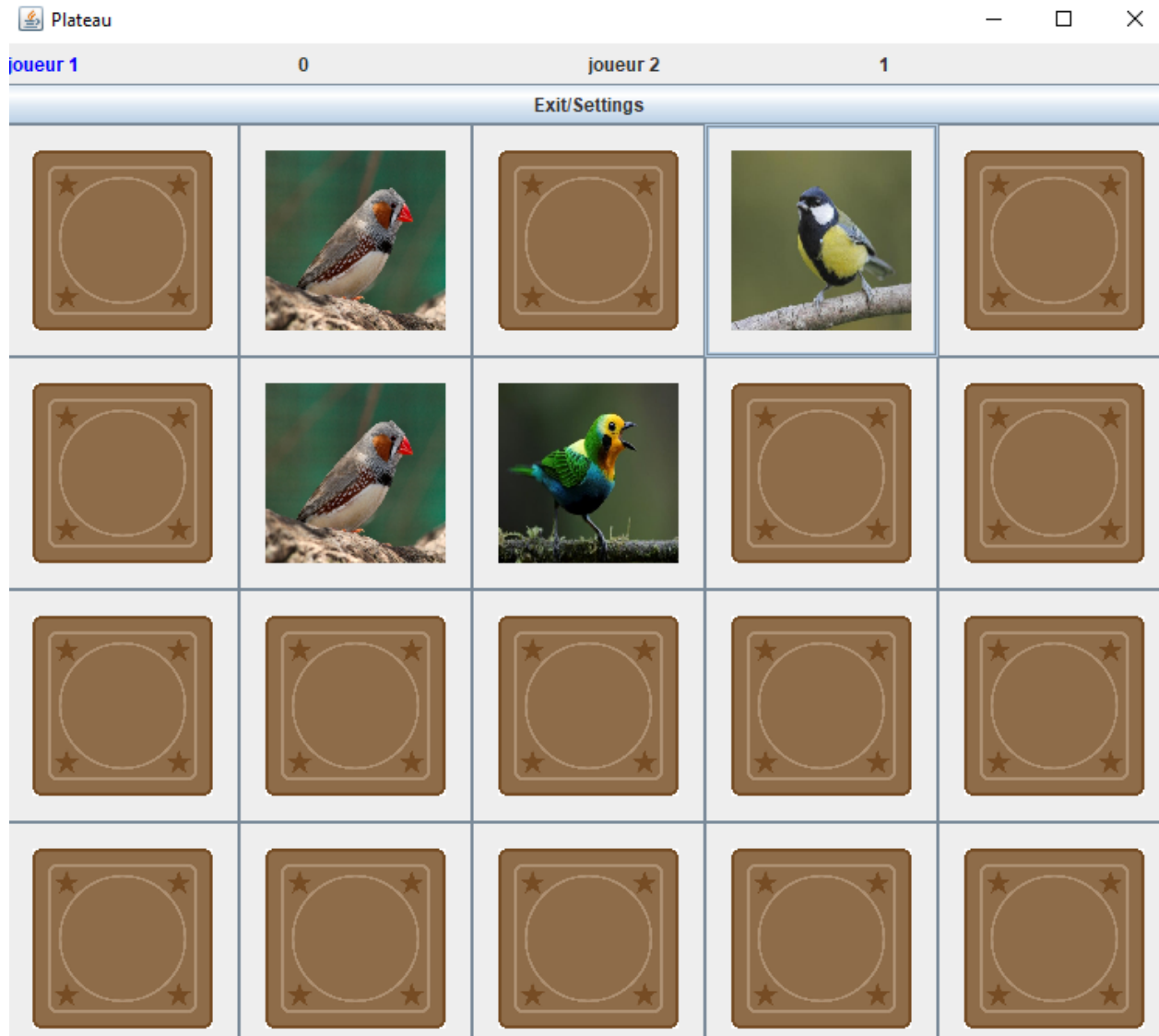
## Fonctionnement

Créer le GameManager suffit à lancer l'application. Il lance ensuite tour à tour la fenêtre des paramètres et la fenêtre de jeu. En effet, il est actionné par le bouton valider des settings et se bouton exit/settings du plateau.

L'objet SettingsManager sert à manipuler la SettingsFrame (ci-dessous) et à conserver les paramètres sélectionnés.

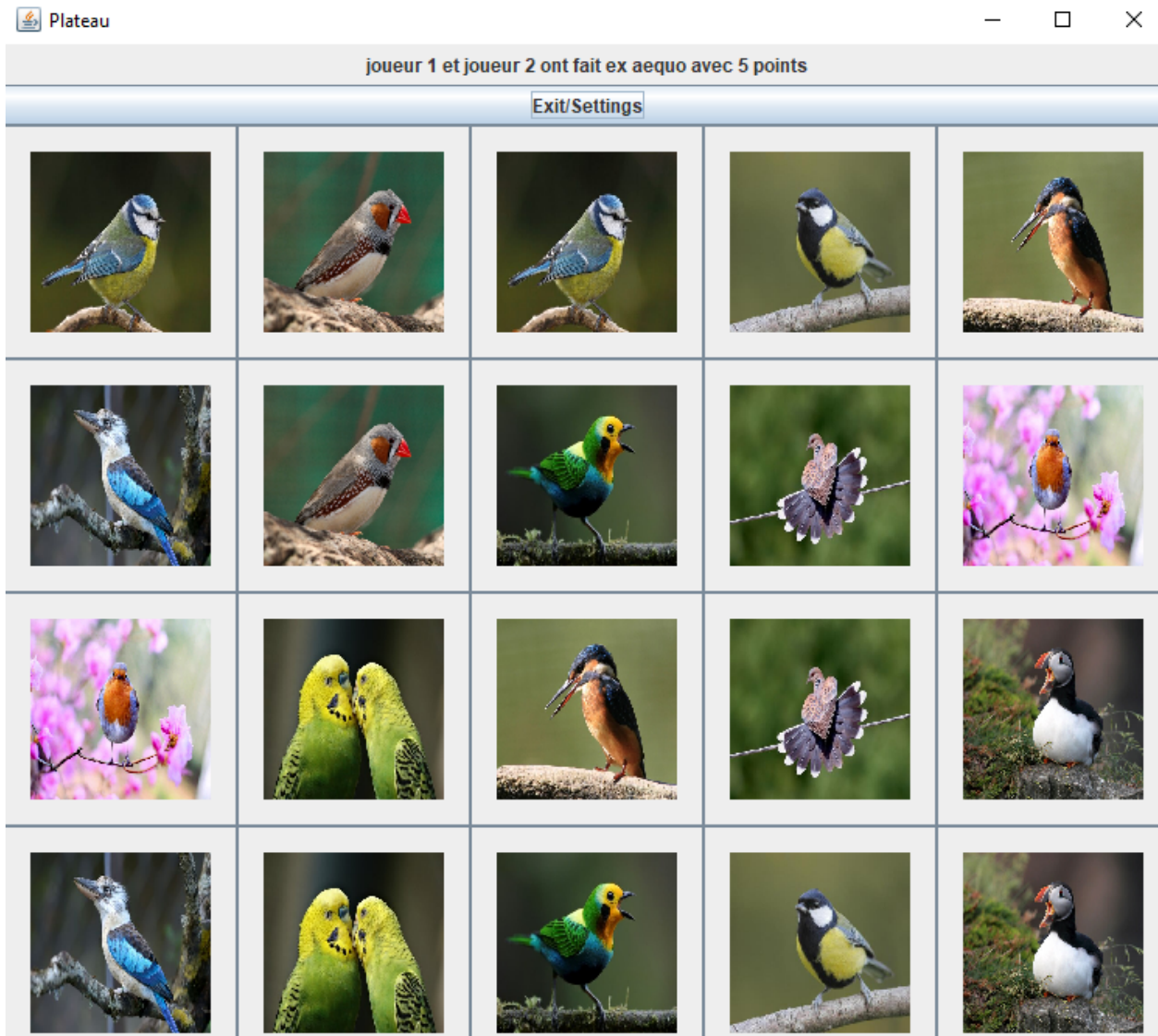


L'objet PlateauManager s'initialise avec un ScoreManager. Il se charge du PlateauFrame et de la boucle de jeu avec la sélection des cartes. Il communique l'avancée du jeu avec Score et lui demande de générer des ScorePannel pour le PlateauFrame (ci-dessous).





En fin de jeu, il demande le remplacement de ce ScorePannel par un message de fin (ci-dessous).



## RÉSULTATS

À la fin de ce projet, nous avons obtenu un logiciel fonctionnel remplissant l'ensemble du cahier des charges. De plus, nous avons réussi à garder une certaine cohérence entre les éléments graphiques et à l'aide de la librairie stretch icône qui nous a permis de redimensionner les images de manière dynamique dans la fenêtre.

## CONCLUSION

Le projet nous a beaucoup apporté d'un point de vue développement objet et design logiciel. Il nous a entraînés à mieux concevoir la structure d'une application avant son développement et les manières de la corriger pendant celui-ci. Nous avons également pu mettre en pratique nos connaissances acquises en cours sur la manipulation d'objets graphiques de la bibliothèque java swing. Enfin, d'un point de vue organisationnel, ce projet en binôme nous a permis d'expérimenter différentes méthodes pour nous répartir les tâches et les confronter avec leur efficacité et nos ressentis.

## RÉFÉRENCES

1. <https://docs.oracle.com/en/java/>.
2. Votre cours.
3. Pour finir, le cours de David GUENNEC.