



ÉCOLE CENTRALE LYON

COMPTE RENDU
BE1
RAPPORT

Jeux d'araignée

Élèves :
Melek KNANI / Mohamed MEZZI

Enseignant :
Stéphane DERRODE

5 octobre 2023

Table des matières

1	Introduction	2
2	Conception	3
3	Realisation	5
3.1	Environnements de développement matériel	5
3.2	Exécution	6
4	Conclusion	9

1 Introduction

Ce rapport a été rédigé dans le cadre de notre premier projet de BE en Applications Concurrentes, Mobiles et Réparties en Java. Il offre une vue d'ensemble de notre projet visant à développer un jeu d'araignée en utilisant le langage Java. Pour atteindre cet objectif, nous avons mis en pratique les compétences et les connaissances que nous avons acquises en cours. Dans les prochaines sections de ce rapport, nous allons détailler les différentes étapes de notre développement et présenter les solutions que nous avons mises en œuvre pour réussir à créer ce jeu d'araignée en Java.

Ce rapport est structuré en trois sections principales. La première section concerne la conception, où nous avons présenté en détail le diagramme de classe du jeu ainsi que toutes les fonctions. Ensuite, dans la deuxième section, nous avons abordé la réalisation du projet, en mettant en évidence les technologies utilisées et en fournissant des captures d'écran du jeu pour une meilleure compréhension. Enfin, dans la troisième section, nous réservons une conclusion globale à notre travail. Cette conclusion résume les objectifs initiaux du projet, synthétise les réalisations accomplies jusqu'à présent, et propose des perspectives d'amélioration future ainsi que le développement continu de notre jeu.

2 Conception

Dans cette section, nous allons présenter le diagramme de classe de notre jeu, accompagné d'une description textuelle de toutes les méthodes de chaque classe.

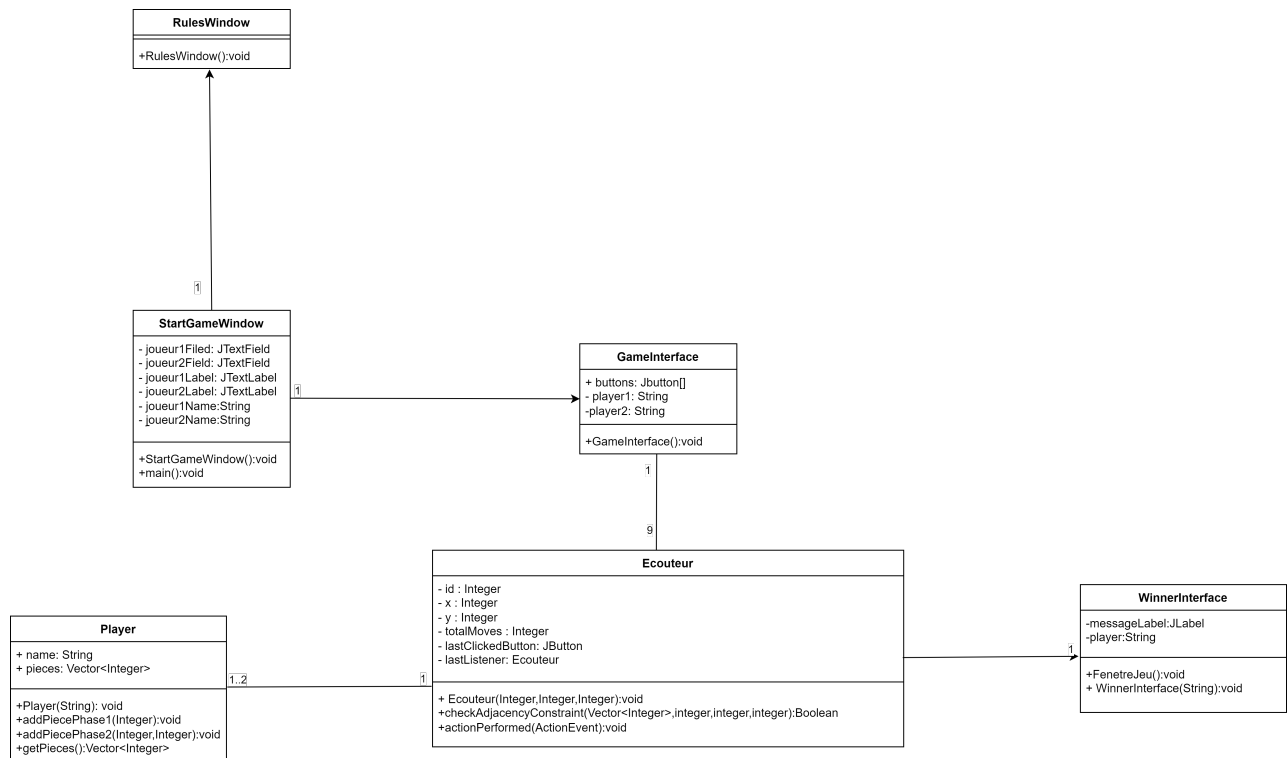


FIGURE 1 – Diagramme de classe

- **Player :**
 - **Player(String) :void** Le constructeur crée un joueur en lui assignant un nom et initialise le vecteur de pièces à vide.
 - **addPiecePhase1(Integer) :void** Ajoute une pièce à la liste de pièces en phase 1 du jeu.
 - **addPiecesPhase2(Integer, Integer) :void** Remplace une pièce par une autre en phase 2 du jeu.
 - **getPieces() :Vector<Integer>** Renvoie le vecteur de pièces.
- **Ecouteur :**
 - **Ecouteur(int, int, int) :void** Cette méthode permet d'initialiser un écouteur en prenant en compte l'identifiant du bouton et ses coordonnées dans la grille.
 - **actionPerformed(ActionEvent) :void** Cette méthode contient la logique du jeu du point de vue de chaque bouton.
 - **checkAdjacencyConstraint(int, int,int) :boolean** Cette méthode permet de vérifier si un mouvement en phase 2 du jeu est adjacent ou non en prenant en compte l'id du bouton cliqué, ainsi que deux coordonnées.
- **StartGameWindow :**
 - **StartGameWindow() :void** Initialise l'interface d'accueil et démarre le jeu.
- **GameInterface :**
 - **GameInterface() :void** Initialise l'interface de jeu et la superposition des boutons

- **WinnerInterface :**
 - **WinnerInterface() :void** Initialise l'interface du gagnant
- **RulesWindow :**
 - **RulesWindow() :void** Initialise l'interface des règles du jeu

3 Realisation

Dans ce chapitre, nous plongerons au cœur du processus de développement de notre solution. Nous vous présenterons en détail nos choix technologiques et illustrerons le fonctionnement de la solution à l'aide de captures d'écran pour une meilleure compréhension.

3.1 Environnements de développement matériel

Java : est un langage de programmation orienté objet polyvalent qui se distingue par sa portabilité sur différentes plates-formes. Il est utilisé pour développer des applications de bureau, web, mobiles et serveurs, en encourageant la structuration du code en objets pour une meilleure modularité et réutilisation. Java est également réputé pour sa robustesse et sa sécurité.



FIGURE 2 – Java

Swing : est une bibliothèque d'interface utilisateur orientée objet pour Java, permettant la création d'interfaces graphiques interactives et personnalisables dans les applications de bureau Java, tout en favorisant une conception modulaire et une expérience utilisateur conviviale

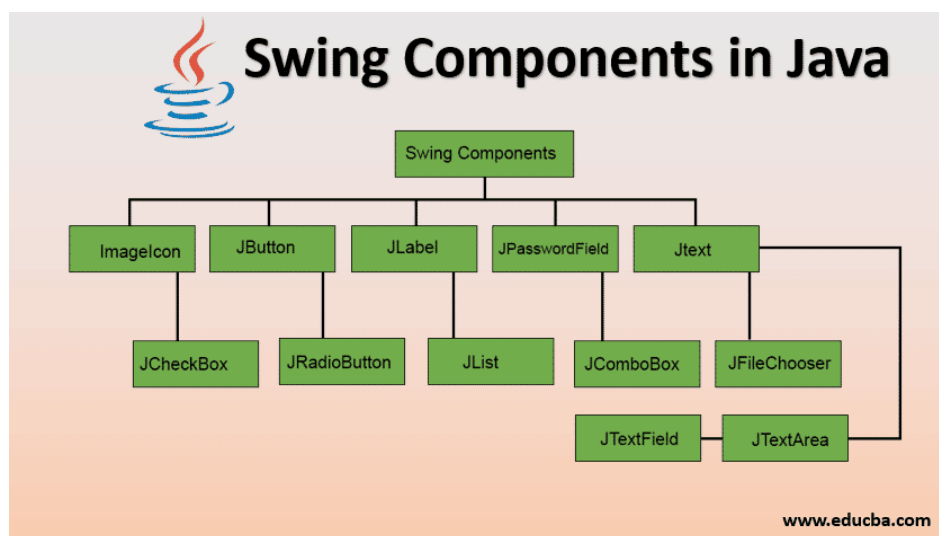


FIGURE 3 – Swing in java

3.2 Exécution

Cette section comprend des scénarios d'utilisation illustrés par des captures d'écran pour offrir une explication détaillée du fonctionnement de notre application.



FIGURE 4 – Écran de démarrage du jeu

Lorsque l'utilisateur démarre l'application pour la première fois, il est automatiquement redirigé vers une page d'accueil qui sert de point de départ du jeu. Sur cette page, les deux joueurs sont invités à entrer leurs noms avant de commencer à jouer. De plus, un bouton est disponible pour accéder à la page des règles du jeu, offrant ainsi une meilleure compréhension des mécanismes du jeu.



FIGURE 5 – Guide du jeu

Lorsque les joueurs accèdent à cette interface, ils sont dirigés vers une section explicative où les règles du jeu sont clairement présentées. offrant aux joueurs des instructions détaillées sur la manière de jouer, les objectifs du jeu, les mécanismes de base, et tout autre élément essentiel nécessaire pour comprendre et apprécier pleinement l'expérience de jeu.



FIGURE 6 – Interface de jeu en cours

Cette interface présente une capture d'écran du jeu en cours, où chaque joueur a sélectionné trois boutons. Les joueurs sont distingués par des couleurs, avec l'un étant représenté en rouge et l'autre en bleu. Cette disposition permet aux joueurs de visualiser les choix et les actions de chacun dans le jeu, ce qui contribue à une expérience de jeu interactive et stratégique

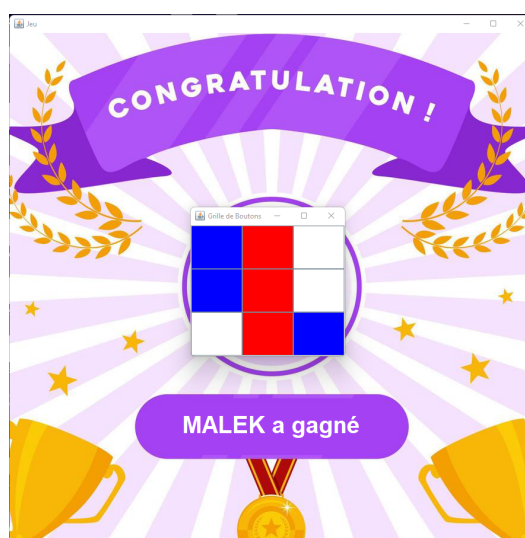


FIGURE 7 – fenêtre de victoire

Quand un joueur réussit à aligner trois cases de sa couleur, qu'elles soient disposées verticalement, horizontalement ou en diagonale, une fenêtre de victoire s'affiche, mettant en évidence le nom du joueur gagnant. Cette fenêtre de victoire offre une récompense visuelle immédiate, permettant au joueur victorieux de savourer sa réussite et aux autres joueurs de prendre connaissance du résultat de la partie de manière instantanée.

4 Conclusion

L'objectif principal de ce projet était de concevoir et développer un jeu en utilisant Java et la bibliothèque Swing. Ce jeu est conçu pour mettre en évidence les compétences acquises en Java et Swing, tout en offrant une expérience de jeu stratégique qui encourage la réflexion et la compétition tout en s'amusant.

Le jeu que nous avons créé est un jeu stratégique qui met en avant le sens de la réflexion des joueurs. L'objectif du jeu est d'aligner trois cases de la même couleur pour gagner, avec les couleurs rouge et bleue disponibles. Cette mécanique de jeu simple encourage les joueurs à élaborer des stratégies pour atteindre cet objectif.

En ce qui concerne le design du jeu, nous avons veillé à ce qu'il soit attrayant et convivial. L'esthétique du jeu est soignée pour offrir une expérience visuelle agréable aux joueurs. De plus, le jeu comporte des éléments interactifs et des fonctionnalités qui le rendent plus intéressant et engageant.

En ce qui concerne les perspectives d'amélioration et de développement futur pour notre jeu, voici quelques axes à considérer :

Tournois et classements : Mettre en place un système de tournois en ligne avec des classements pour stimuler la compétition entre les joueurs. Les joueurs pourraient gagner des récompenses virtuelles ou des titres en fonction de leurs performances.

Intelligence artificielle : Ajouter un mode solo avec une intelligence artificielle (IA) capable de jouer contre le joueur, cela peut être un défi complexe. La conception d'une IA efficace pour résoudre le puzzle du jeu peut être complexe.