Projet Hex - Qualité de Développement

NOMS DES PARTICIPANTS :

ESTEVES Gabriel 203
LENOUVEL Louis 203
AMIRAT Nael 203

Table des matières

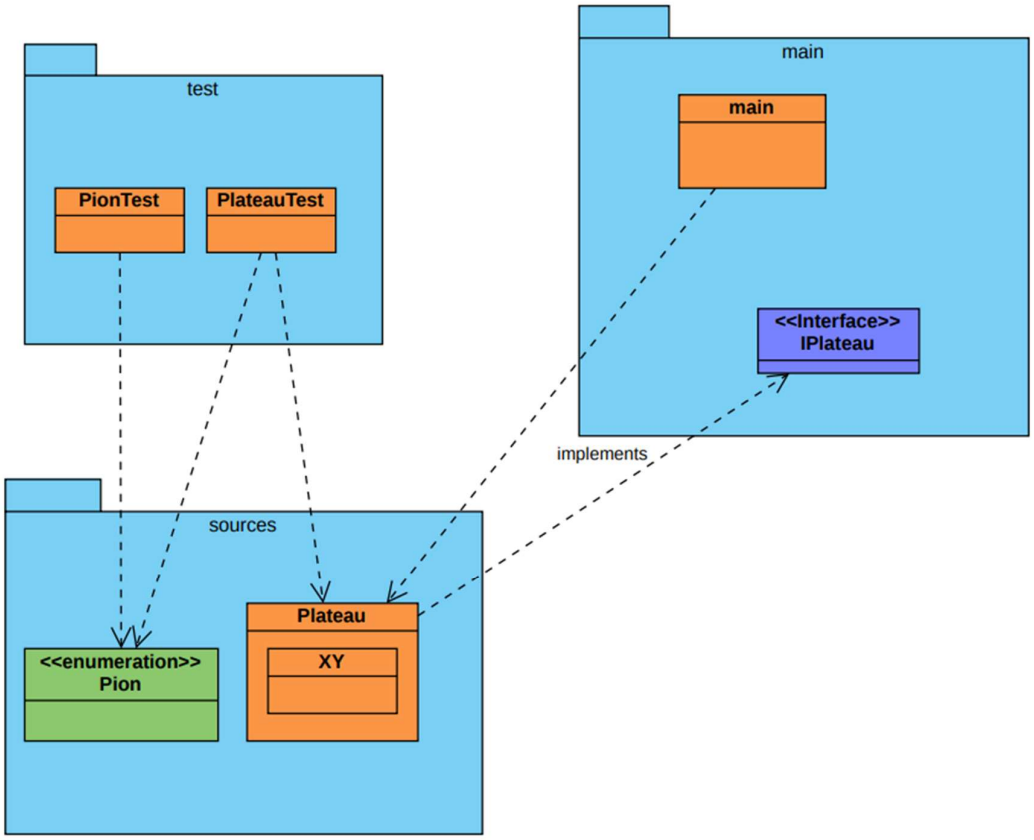
Présentation du projet

Brève présentation :

L’objectif de ce projet était de programmer un jeu intitulé « Hex» en plus d’utiliser tous types d’algorithmes qui font utiliser un grand nombre de fonctions dans le langage Java. Nous devions réaliser ce projet tout en respectant le principe SOLID, c'est-à-dire soigner notre code et établir une hiérarchie ainsi que choisir les parties qui seront publiques ou non. C’est un jeu qui se joue à deux joueurs. Le but du jeu est de créer un chemin entre son bord du plateau et son opposé.

Il est rassurant de savoir que notre code fonctionne correctement et que nous n'avons pas rencontré de problèmes majeurs. Nous avons travaillé de manière méthodique, avons pris le temps de tester et de déboguer notre code pour nous assurer qu'il fonctionne correctement.

Diagramme d’architecture



Liste des tests

Pour PlateauTest.java :

- Test du toString() et de la taille pour voir si cela renvoyait bien la grille voulue.
- Test pour jouer un coup, et afficher que le pion à la case où nous avons joué le coup est bien le pion que nous avons joué.
- Tests pour savoir si 3 positions sont valides.
- Tests pour savoir quels pions les joueurs possèdent
- Test pour savoir si lorsque nous choisissons le mode 1 (joueur vs joueur), le joueur 1 et le joueur 2 existent
- Test pour savoir si lorsque nous choisissons le mode 2 (joueur vs IA), le joueur 1 est un robot donc n'existe pas et le joueur 2 existe.
- Test pour savoir si lorsque nous choisissons le mode 3 (IA vs IA), le joueur 1 et le joueur 2 n'existent pas.
- Test pour savoir si la partie est finie juste en jouant 2 coups.
- Test pour connaître le joueur gagnant.
- Test pour savoir si l'inversement des pions fonctionne bien. Le joueur 1 possède les croix et le joueur 2 les ronds. Après inversement, le joueur 1 possède les ronds et le joueur 2 les croix.

Pour PionTest.java :

- Test pour savoir si la fonction ToString() d'un pion marche. Exemple : Pion.Croix.toString() = "X"
- Test de la fonction Get pour voir si lorsque nous lui demandons un symbole qui existe elle renvoie le pion correspondant . Sinon, on lève une exception.

Explication des tâches à accomplir

Nous avons déjà implémenté la fonctionnalité Inverser Pion qui permet de changer de pion après le coup du joueur 1 si le joueur 2 le souhaite.
Pour améliorer notre projet nous pouvons mettre en place un coup intelligent pour les joueurs simulés. Nous pourrions alors faire en sorte que le joueur simulé place d'abord un pion à sa destination ou son point de départ pour s'assurer une chance de gagner. Ensuite, il pourrait continuer ce chemin en parcourant le plateau. S'il rencontre un pion adverse en chemin, nous ferons en sorte qu'il le contourne.

Bilan du projet

Nous avons réussi à développer le jeu de Hex de manière efficace et à résoudre la plupart des difficultés rencontrées grâce à notre capacité à résoudre des problèmes de programmation. Les fonctions chercher() et estFinie() ont été les parties du code qui ont été les plus difficiles à mettre en œuvre.

Concernant ce qui peut être amélioré, il est important de continuer à prendre soin de notre code en suivant les meilleures pratiques de programmation et en veillant à ce qu'il soit bien organisé et facile à lire et à comprendre. Cela peut nous aider à éviter des erreurs de programmation, à rendre notre code plus facile à maintenir et à mettre à jour à l'avenir.