

Description diagramme de classe

Dans notre application, la class pion permet de créer et de gérer les pions nécessaires à son fonctionnement. Grâce à elle, on peut obtenir des informations sur ceux-ci comme leur position ou leur couleur.

La class Case correspond quant à elle, aux cases du damier de notre jeu. Elle permet de situer les pions sur le plateau via des coordonnées et indique si un pion se trouve sur elle.

Pour ce qui est de la class plateau, elle permet de créer le plateau en utilisant les classes case et pion. De plus, elle permet de gérer les pions, c'est grâce à elle que l'on peut déplacer les pions en indiquant les coordonnées de celui-ci et celle de l'endroit ciblé. Également, elle gère les règles car c'est elle, via la méthode déplacer, mangerAffiler, coupPossible et Possibilité manger qui va dire si le déplacement est possible et si cela entraîne la prise d'un pion adverse.

Concernant la class joueur, elle permet de gérer les joueurs qui sont dans la partie. On a accès à leur nombre de pions ainsi qu'à la couleur de ceux-ci et leur nom.

La class partie nous permet de lancer le jeu. Elle lance la création du plateau, crée les joueurs et détermine si un joueur a gagné.

La class Display contient les méthodes que nous allons abonner aux événements.

La class Dame permet d'instancier une dame et de la différencier des autres pions.

La class Program contient les abonnements entre événements et les méthodes de la class Display ainsi que la boucle de jeu.

La class Game permet de faire la persistance sur la class partie via l'interface ISerializer et lancer la partie.

De plus , afin d'ajouter de la persistance à notre projet il y a les classes JsonSerializer et XMLSerializer qui implémentent l'interface ISerializer. Cette interface comprend une méthode pour écrire dans un fichier et une autre méthode pour lire le fichier.