

Rapport de project APOO

Pierre Jouffroy

Alphée Grosdidier

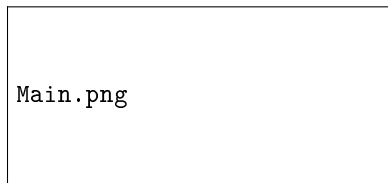
19 mai 2021

Table des matières

1 Présentation globale de la structure du programme

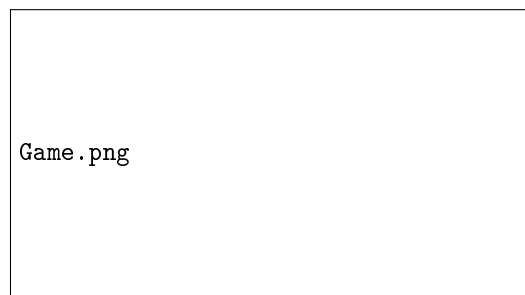
graphicx Le modèle (UML) de chacune des classes de notre projet, avec une courte explication sur leur utilité. Les - avant les attributs ou les méthodes veulent dire "private" les + veulent dire "public" les méthodes / attributs soulignés sont static

Class Main



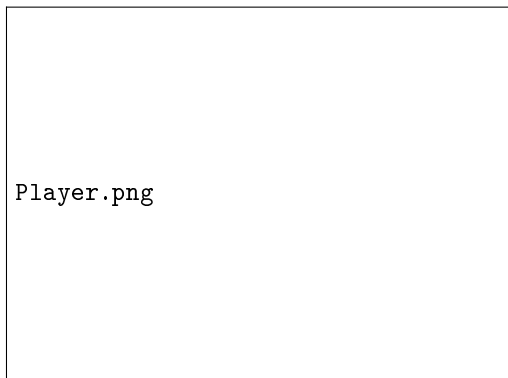
La classe Main qui va affiche les règles, demande quelle règle donc type de jeu on veut et lance une nouvelle partie avec les règles choisies dans une boucle

Class Game



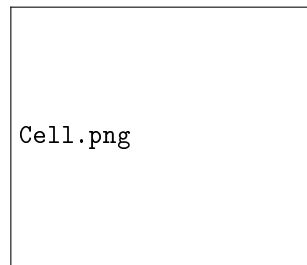
La classe Game est la classe qui gère le jeu, nouvelle partie, nouveau joueur, la boucle du jeu et la fin du jeu.

Class Player



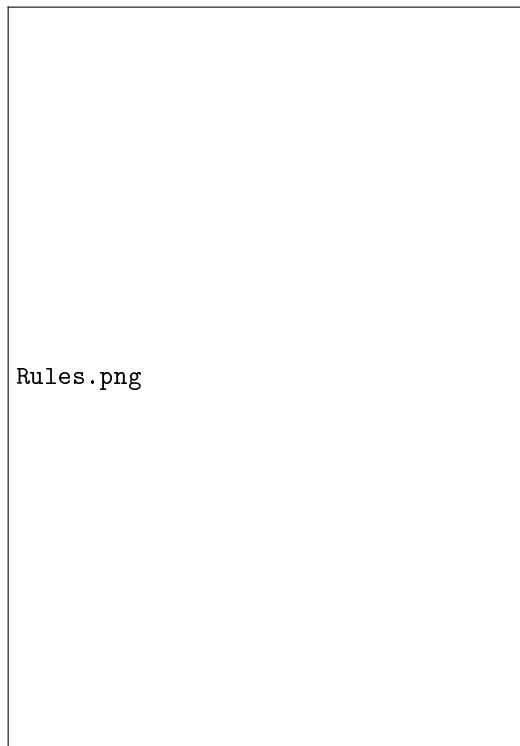
La classe qui gère le joueur, avec sa position et son stockage de pions rouge.

Class Cell



La classe qui gère les cases, avec les pions.

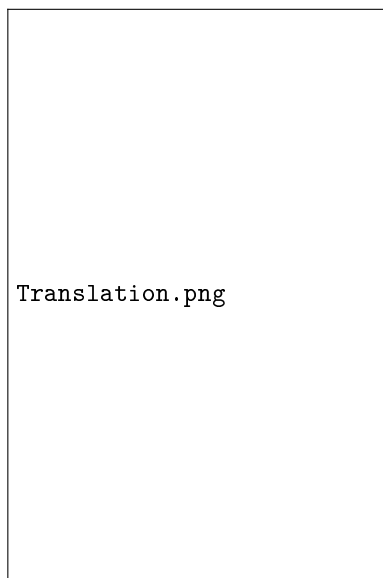
Class Rules



Rules.png

La classe regroupant les règles du jeu.

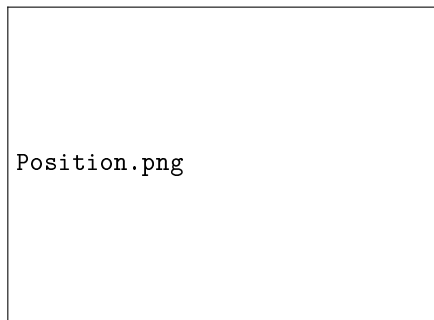
Class Translation



Translation.png

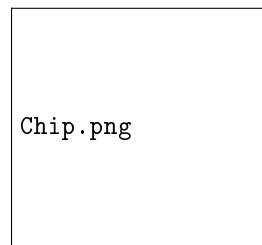
La classe regroupant tous les textes du jeu.

Class Position



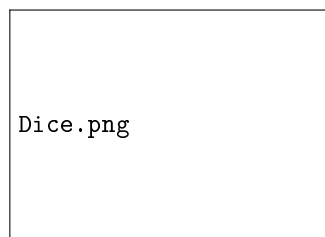
La classe qui sert a la position du joueur sur la grille.

Class Chip



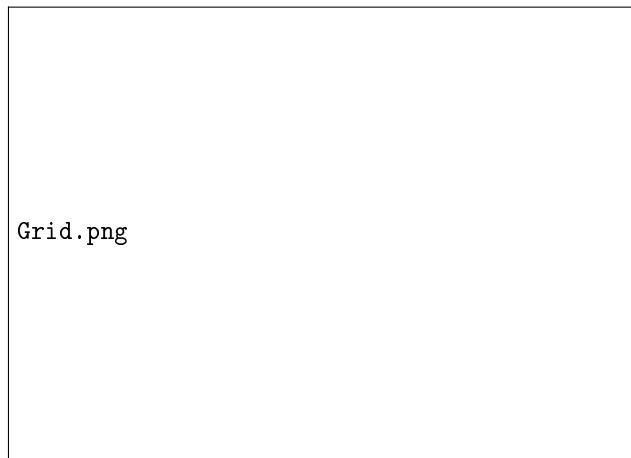
La classe des pions sur le plateau et dans les storages.

Class Dice



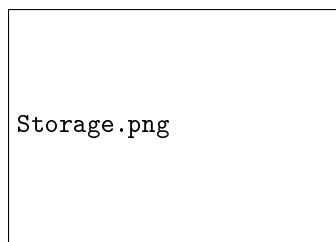
La classe qui gère le dé de la partie Génère un dé a x face et tire aléatoirement un nombre entre 1 et x.

Class Grid



La classe qui contient le plateau du jeu avec les positions des pions et la liste des pions vert hors du plateau.

Class Storage



La classe contenant les set de pion du joueur et du plateau de jeu.

- 2 Le fonctionnement de main et la compilation du programme**
- 3 L'algorithme de répartition des pions sur la grille et dans les banques**
- 4 Comment sont traitées les actions utilisateur**
- 5 Les bonus que nous avons ajouté**