

Rapport DEV4 :

On a choisi de suivre un « software design pattern » MVC. On a donc créé 4 packages : le Main, le Controller, le Model et le View (qui n'est pas encore représenté).

Le package Model est constitués de plusieurs classes différentes :

- Player
 - o Représente le/les joueurs. Cette classe possède comme attributs : un nom, une liste de balles qui lui sont attribuées au début du ainsi que la quantité de celles qui se trouvent encore sur le plateau de jeu. Cette classe nécessite l'énumération qui suit afin de bouger les balles qu'elle possède.
- Directions
 - o Énumération représentant les 6 directions possibles pour effectuer un mouvement dans le jeu.
- Hexagon
 - o Elle représente 1 hexagone sur le plateau de jeu. Comme attributs, elle a : un status, soit EMPTY, soit WHITE ou BLACK correspondant à la couleur de la balle qui est dessus.
- Board
 - o Cette classe représente le plateau de jeu. Elle possède un vecteur de vecteurs d'hexagones. Son pseudo-constructeur 'initBoard()' met en place un nouveau jeu. Sur ce plateau se trouvent des 'Ball' qui ont une couleur, un status 'onBoard' qui peut être vrai ou faux et une position. On note diverses actions que la balle peut prendre, par exemple fallProcedure(), qui, lorsqu'une balle d'un des joueurs tombe du plateau, diminue le compteur de balles chez celui-ci ainsi que change le status de cette balle de true à false pour l'attribut onBoard. Une autre méthode 'move()' déplace la balle de sa position à une autre position grâce à des indications comme la direction.
- Position
 - o Classe permettant de situer les balles de chaque joueur sur le plateau. Nous avons décidé d'utiliser un système de coordonnées à deux axes et cela afin d'éviter la redondance d'une des 3 coordonnées dans le positionnement des Ball.
- AbaPro
 - o Classe traduisant les commandes du jouer en déplacement valide pour la classe Player.
- Game et Model
 - o La classe Game et son interface Model vont permettre au 'Controller' d'interagir avec les différentes classes.

Ensuite, le package 'Controller' qui possède qu'une classe, la classe 'controller'. Celle-ci va commencer un nouveau jeu, contrôler le jeu de 'Play' à 'Play' (Un Play correspond à un mouvement du joueur avec les Ball noirs, suivi de la vérification du nombre de Ball hors du plateau. Par après, c'est au joueur avec les Ball blancs de jouer et on vérifie à nouveau ne no de Ball hors du plateau.). Ce Play va également inclure une vérification de l'état du jeu qui se termine au moment où au moins 7 balles d'une couleur sont tombés et ce grâce à la méthode isOver().

Finalement, le 'Main' qui va lancer un nouveau jeu si cela est souhaité.