Rapport DEV4 :

On a choisi de suivre un « software desing pattern » MVC. On a donc 4 package : le Main, le Controlleur, le Model et le View (qui n’est pas encore représenté).

Le Model se constitue de plusieurs sous classe différent; Le‘Player’ qui a un nom, qui connait toutes les balles de sa couleur et la quantité de celle-ci se trouvant sur le jeu. Il interagit avec ‘Direction’ pour bouger sur le plateau. Directions est une énumération de ses 6 coups possible.

La classe ‘Hexagon’ représente 1 hexagone sur le plateau de jeu. Il a un status, soit vide, soit une balle blanche ou noire. Le plateau de jeu, représenté par la classe ‘Board’ possède un vecteur de vecteur d’hexagone. Son pseudo constructeur ‘initBoard()’ met en place un nouveau jeu. Sur ce plateau se trouve 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 met tout en place pour qu’une balle c’est bien tombé du plateau ou encore un méthode ‘move()’ qui déplace la balle d’une position initiale à finale. La ‘Position’ sur le plateau aussi bien pour la ball que pour le joueur. On a choisi un système de coordonnées axiale. On voulait éviter la redondance d’une des 3 coordonnées dans le tableau.

Encore, la méthode ‘Game’ et son interface ‘Model’ qui vont permettre au ‘Controller’ d’interagir avec les différentes classes.

Ensuite, le package ‘Controller’ possède qu’une classe, le ‘Controller’. Celui-ci va commencer un nouveau jeu, contrôler le jeu de ‘play’ à ‘play’. Celui-ci va aussi regarder si le jeu est fini (7 balles tombé d’une couleur) grâce à la méthode ‘isOver’.

Finalement, le ‘Main’ qui va lancer un nouveau jeu.