**PROJET : note de développement**

Version Java

1) Représentation de coordonnées

Bien. Un premier problème se pose en Java : l’inexistence des tuples. Il faut pourtant bien associer un système de coordonnée Y,X aux cases. Il faut donc créer sa propre version des tuples, qui fait ce qu’il faut : représenter un point dans un système 2D. D’où la création de la classe Coord. Il a fallu décider du format de Coord. Une classe propre ? Une classe interne ? Etant donné que Coord devra intervenir dans la gestion des Items (pions, etc) mais aussi des cases. Il fut décider de créer un package *tools* qui allait répertorié tous les outils créée pour servir le projet (ex : Coord)

2) Rules : package ou non ?

Rules est sensé contenir toute la logique de jeu. Package ou non ? En faire un package signifierais créer un objet par règle. Hors, ce qui est nécéssaire c’est de pouvoir appeler des méthodes qui représentent ces règles. Donc Rules ne sera pas un package mais bien un objet contenant toute les règles. Il sera intégré dans le package Game afin d’avoir accès à l’ensemble du jeu (plateau, murs, joueur, etc).

3) les directions :

Dans sa version de base, le jeu comporte 4 directions. Nous les avons définies en Java dans un Hashtable que nous avons déclaré constant dans la classe *Board* . Il est apparut néanmoins dans nos prototype que cette donnée est nécessaire à beaucoup d’autres endroits dans notre code. Mais plutôt que de définir la même chose à chaque fois, il est préférable de le définir une seule fois et de s’en servir. Par défaut, nous l’avons défini dans le board mais une classe *Global* qui contiendrait ça serait peut etre plus adapté. Question posée aux assistants. 

4) les déplacements

Comment gérer un déplacement ? En sachant que l’action va être indiquée par ActionHandler, où vérifier la validité du mouvement ?

* Downcasting transtypage MOAI to Pawn