



LO02 – Diagramme de classes et état fonctionnel

Version 1.0

GODEC Tristan, UYTTERHAEGEN Romain

Automne

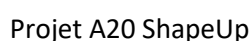
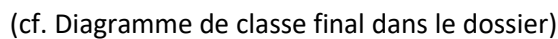


utt
UNIVERSITÉ DE TECHNOLOGIE
TROYES

Table des matières

LO02 – Diagramme de classes et état fonctionnel.....	1
Changements apportés à la structure du code	3
Premier diagramme de classes.....	3
Diagramme de classes actuel	3
Changements effectués.....	4
Etat fonctionnel du projet	5
Fonctionnalités disponibles.....	5
Améliorations possibles	5

(cf. Diagramme de classe initial dans le dossier)



Changements effectués

Entre le premier diagramme de classe et celui actuel, on peut remarquer différents changements majeurs :

- Il y a bien plus de méthodes dans la version finale. En commençant à développer l'application, il y a certaines méthodes qui nous ont paru nécessaire d'ajouter, et cela était prévisible. Il y a certains attributs rajoutés à certains endroits aussi.
- Le patron de conception Strategy a été modifié. Désormais, le joueur n'a plus d'objet héritant de Strategy, cela est uniquement valable pour les différents niveaux de difficulté des joueurs robots.
- Deux nouvelles classes « utilitaires » font leur apparition. La classe Turn permet de faire transiter dans un retour de méthode l'ensemble des informations d'un tour de jeu, on peut voir cette classe comme un tableau regroupant tout ce qu'il faut pour faire dérouler un tour. Il y a aussi la classe SortPlayerByPoint qui permet de trier aisément une liste de Player en fonction de leur nombre de points.
- Autre changement qui ne se remarque pas au début, mais Game est désormais aussi lié avec Board, afin d'avoir un Board de référence.

Etat fonctionnel du projet

Fonctionnalités disponibles

- Le cœur de l'application est fonctionnel, autant en vue console qu'avec l'interface graphique
- Les 3 modes de jeu et plateaux sont jouables.
- Le jeu répond aux attentes de base du cahier des charges

Améliorations possibles

- Il est possible d'implémenter des niveaux de difficulté pour les joueurs robots.
- On peut améliorer l'interface graphique avec un drag'n'drop par exemple.
- Le 3^{ème} mode de jeu peut être modifié de sorte que l'on choisisse nous-même les paramètres.
- On pourrait permettre de relancer une partie à la fin, tout en gardant les points précédents.
- Certaines parties algorithmiques de l'application pourraient être optimisées.