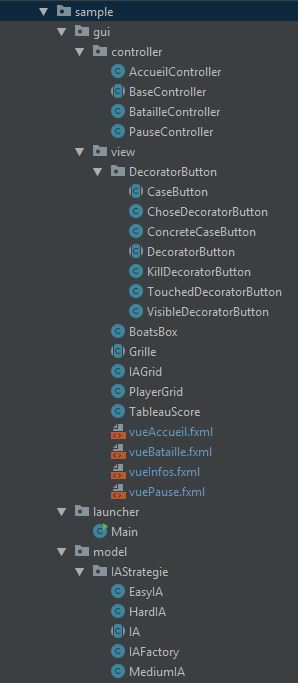
(1. sample)  
 1.1. gui  
 1.1.1. controller  
 1.1.1.1. BaseController  
 1.1.1.2. AccueilController  
 1.1.1.3. BatailleController  
 1.1.1.4. PauseController  
 1.1.2. view  
 1.1.2.1. DecoratorButton  
 1.1.2.1.1. CaseButton  
 1.1.2.1.2. ChoseDecoratorButton  
 1.1.2.1.3. ConcreteCaseButton  
 1.1.2.1.4. DecoratorButton  
 1.1.2.1.5. KillDecoratorButton  
 1.1.2.1.6. TouchedDecoratorButton  
 1.1.2.1.7. VisibleDecoratorButton  
 1.1.2.2. BoatsBox  
 1.1.2.3. Grille  
 1.1.2.4. IAGrid  
 1.1.2.5. PlayerGrid  
 1.1.2.6. TableauScore  
 1.1.2.7. vueAccueil.fxml  
 1.1.2.8. vueBataille.fxml  
 1.1.2.9. vueInfos.fxml  
 1.1.2.10. vuePause.fxml  
 1.2. launcher  
 1.2.1. Main  
 1.3. model  
 1.3.1. IAStrategie  
 1.3.1.1. EasyIA  
 1.3.1.2. HardIA  
 1.3.1.3. IA  
 1.3.1.4. IAFactory  
 1.3.1.5. MediumIA  
 1.3.2. Observer  
 1.3.2.1. IObserver  
 1.3.2.2. Partie  
 1.3.2.3. Subject  
 1.3.3. Difficulty  
 1.3.4. GrilleMdl  
 1.3.5. Point

Clément TORTI  
Florent BECOUZE  
G7  
Année 2018/2019

**Conception et programmation Objets Avancées**Université Clermont-AuvergneIUT Info 2A

**Documentation Code :  
Bataille-navale**



**(1. sample)**

C’est le dossier global du projet, dans lequel on a une structure MVC : dossier « gui » pour les packages « controller » et « view » et le package « model »

**1.1. gui (graphical user interface)**

C’est le dossier qui possède les packages « controller » et « view ». Dans le dossier, il y a tout ce qui concerne le côté graphique de l’application.

**1.1.1. controller**

C’est le package permettant de rassembler tous les contrôleurs des différentes vues.

**1.1.1.1. BaseController**

fd

**1.1.1.2. AccueilController**

hgfd

**1.1.1.3. BatailleController**

gf

**1.1.1.4. PauseController**

hgf

**1.1.2. view**

C’est le package permettant de rassembler tous ce qui est en rapport avec la vue (ce que voit l’utilisateur).

**1.1.2.1. DecoratorButton**

Ce dossier permet de gérer le design des boutons (ceux représentant les cases de la bataille) en fonction de leurs états, c’est-à-dire, s’ils représentent une case vide, une case avec un bateau dessous ou si le bateau est coulé.

**1.1.2.1.1. CaseButton**

ati

**1.1.2.1.2. ChoseDecoratorButton**

xcr

**1.1.2.1.3. ConcreteCaseButton**

bgfv

**1.1.2.1.4. DecoratorButton**

L ;,

**1.1.2.1.5. KillDecoratorButton**

N,

**1.1.2.1.6. TouchedDecoratorButton**

nb

**1.1.2.1.7. VisibleDecoratorButton**

nhtfgv

**1.1.2.2. BoatsBox**

njgv

**1.1.2.3. Grille**

Rg

**1.1.2.4. IAGrid**

N gh

**1.1.2.5. PlayerGrid**

Ngv

**1.1.2.6. TableauScore**

hng

**1.1.2.7. vueAccueil.fxml**

nh

**1.1.2.8. vueBataille.fxml**

,jhng

**1.1.2.9. vueInfos.fxml**

,k

**1.1.2.10. vuePause.fxml**

jgnfb

**1.2. launcher**

C’est le package permettant de rassembler la classe « Main », celle qui va être lancée au tout début du programme.

**1.2.1. Main**

N gf

**1.3. model**

C’est le package permettant de rassembler tous les classes en rapport avec ????

**1.3.1. IAStrategie**

Ce dossier contient l’ensemble des classes, gérant la difficulté de jeu de l’IA.

**1.3.1.1. EasyIA**

vcn

**1.3.1.2. HardIA**

gtref

**1.3.1.3. IA**

gte

**1.3.1.4. IAFactory**

gtre

**1.3.1.5. MediumIA**

rtefc

**1.3.2. Observer**

Ce dossier ???? contient les classes permettant de gérer le pattern « Observateur » ????

**1.3.2.1. IObserver**

fdc

**1.3.2.2. Partie**

asz

**1.3.2.3. Subject**

U^po

**1.3.3. Difficulty**

Mo^po

**1.3.4. GrilleMdl**

bnc

**1.3.5. Point**

rhgf