

# Mini projet de fin de cours

03.05.2017

## Introduction

L'objectif est de développer complètement un petit projet en commençant par l'analyse, la modélisation et en terminant par une implantation en Java. Afin de créer cette application graphique, il vous est demandé d'apprendre la programmation événementielle avec la librairie *Swing* qui fournit les éléments pour réaliser de telles applications.

- Ce mini projet peut être réalisé en binôme.
- Date de rendu : jeudi 1er juin 2017

## Cahier des charges

Nous vous demandons de réaliser un jeu de bataille navale. L'objectif est simplement de remplacer la version papier du jeu.

La bataille navale se joue à deux joueurs qui se parlent (pas de communication de machine à machine). Chaque joueur dispose de son ordinateur qui affiche une grille de taille 10x10 sur laquelle il positionne un porte avion (taille 1 case par 5 cases), 2 croiseurs (1 case par 4 cases) et 2 torpilleurs (1 case par 3 cases). Au début du jeu, chaque joueur positionne ses bateaux. Un bateau peut être placé horizontalement ou verticalement, mais pas en diagonale. Ensuite, chaque joueur dont c'est le tour choisit une des 100 cases sur laquelle il va « tirer » en annonçant la case choisie à son adversaire. L'adversaire vérifie si cette case correspond à l'un de ses bateaux. Si c'est le cas, ce bateau est endommagé. Un bateau est coulé s'il est touché en toutes ses cases. Un joueur qui a touché un bateau de l'adversaire rejoue. Le premier joueur à avoir coulé la flotte de l'adversaire a gagné. Le jeu devra disposer d'une interface graphique pour le placement des bateaux et pour le suivi des coups portés à l'adversaire.

Le jeu se passe en deux étapes :

- 1) Placer les bateaux dans le jeu. Ceci se fait en sélectionnant chaque type de bateau et en sélectionnant ensuite les cases dans lesquelles le bateau est placé. Le système s'assure que tous les bateaux sont placés avant le début de la partie et qu'ils ont la taille requise. La manière exacte de positionner les bateaux et de les afficher dans la grille est laissée à la discrétion du développeur.
- 2) « Tirer » dans les cases (c'est à dire les annoncer) de manière à chercher à toucher un bateau de l'adversaire. Quand un joueur annonce un coup à son adversaire, il marque la case correspondante dans son jeu afin de savoir quelles sont les cases déjà explorées. Quand un coup est réussi (un bateau de l'adversaire est touché), le joueur change le marquage de la case pour s'en souvenir. Quand l'adversaire annonce « coulé », les cases correspondant au bateau coulé sont marquées spécifiquement. Finalement les coups reçus dans ses propres bateaux sont également marqués. La manière d'identifier les

coups portés et reçus ainsi que les bateaux coulés chez son adversaire (forme, couleur, texte,...) est laissée à la discrétion du développeur

## Travail attendu

- Faites une analyse RDD du problème (rendu : cartes CRC pour chaque classe et diagramme de classe UML résultant)
- Modélisez le comportement des éléments critiques (non triviaux) à l'aide des diagrammes de séquence et d'états (rendu : modèles UML correspondants. Produisez au moins un modèle de chaque type)
- Implantez votre solution en Java (rendu : code source)

## Documentations

- Bataille navale [http://fr.wikipedia.org/wiki/Bataille\\_navale\\_\(jeu\)](http://fr.wikipedia.org/wiki/Bataille_navale_(jeu))
- Swing
  - Tutorial Oracle : <http://docs.oracle.com/javase/tutorial/uiswing/>
  - Tutoriel en français:  
<http://baptiste-wicht.developpez.com/tutoriels/java/swing/debutant/?page=fenetres#L3.3>