

# S1-01 Implémentation d'un besoin client

CHOFFAT Rémi - BESANÇON Marcelin - S1E

Nous nous sommes partagé le travail en faisant une classe chacun, l'une après l'autre. Nous nous sommes également partagé l'écriture des tests. Nous avons appris à utiliser GitHub pour nous faciliter la tâche de travail collaboratif.

Première partie : 5h environ

Deuxième partie : 9h environ

Nous avons terminé la version 2 avant la fin des 12h de SAÉ.

(Nous nous sommes basés sur la version 1 du sujet).

Difficultés rencontrées :

Faire le test pour qu'un paquet soit bien mélangé: Notre méthode *melangerPaquet()* utilise la méthode *piocherHasard()*. Elle crée un paquet temporaire dans lequel on ajoute des cartes tirées aléatoirement du paquet initial sur le dessus du paquet temporaire. Puis on remplace le paquet initial par le paquet temporaire. Cependant, il y a une probabilité que le paquet temporaire soit exactement le même que le paquet initial.

Pour pallier ce problème, il faudrait faire en sorte, lors du premier tour de boucle pour ajouter les cartes dans le nouveau paquet, de ne jamais prendre la dernière carte, et de pouvoir seulement l'ajouter après le premier ajout. Cela a pour effet de ne jamais avoir le même paquet car il y aura au minimum 1 carte changée.

Les méthodes s'appellent les unes les autres, donc si une a une erreur, on se retrouve à faire un cherche et trouve dans nos fichiers.

Le test de la méthode *toString()* est insupportable pour des histoires d'espaces...

Pour la classe Jeu, nous avons fait en sorte de bien faire communiquer toutes les classes entre elles et de ne pas créer de nouveau bug avec les différents contextes de jeu.

Nous avons utilisé les méthodes les plus évoluées pour *lancerJeu()* : tant que le jeu n'est pas fini (*etreFini()* = 0), on demande au joueur quelle carte il veut jouer sur quelle pile, puis on la joue si cela est possible.

Lors de cette SAÉ, nous avons appris principalement à résoudre des problèmes de manière efficace, et à optimiser nos programmes.