

*Projet réalisé par :
Fabio DE SOUSA LIMA
&
Capucine DAVEINE*

Rapport Projet POOIG :

Sommaire :

- I. Déroulement du projet
- II. Problèmes rencontrés
- III. Pistes d'amélioration

Introduction :

Ce jeu a été réalisé dans le cadre d'un projet de programmation de L2. Le sujet étant de programmer un jeu de société « Azul », jouable sur un terminal de commande mais aussi sur une interface graphique.

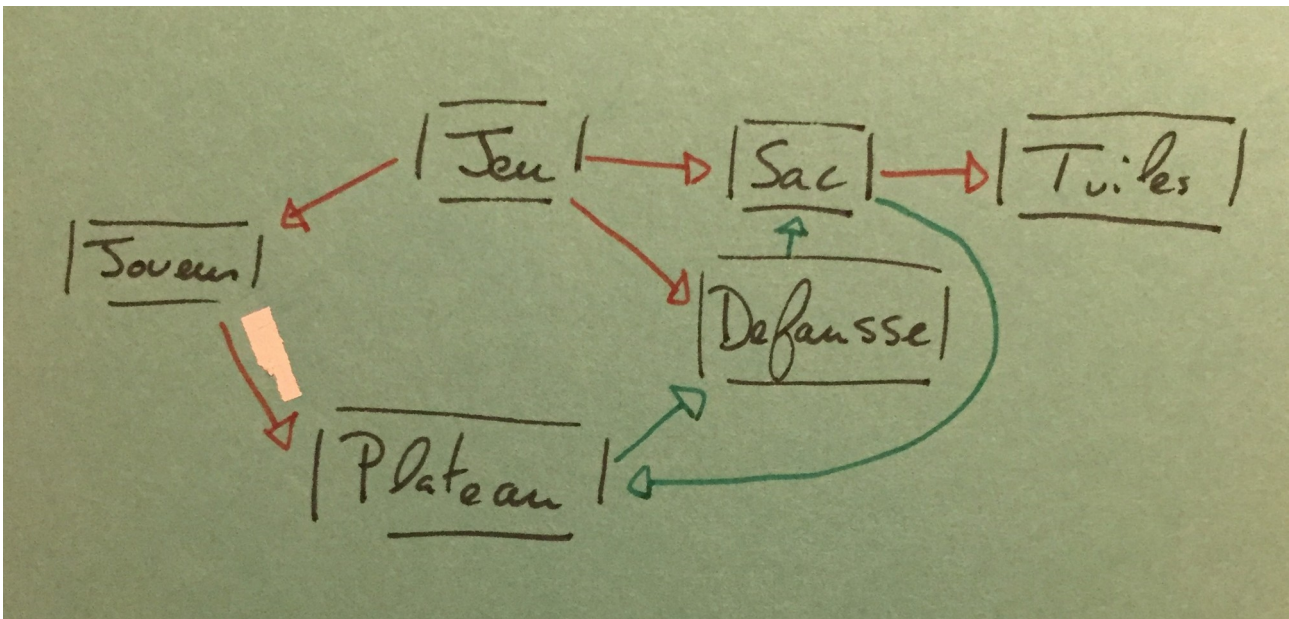
En binôme nous avons donc 2 mois et demi pour construire l'architecture du projet coder la version terminal et implémenter une interphase graphique à partir des connaissances acquises en cours.

Voici donc un document récapitulatif de l'évolution de notre projet. Un ReadMe est aussi disponible, il explique comment lancer le projet.

I. Déroulement du projet

Avant de commencer à programmer notre jeu nous nous sommes mis d'accord sur une architecture de base du projet (qui a bien sûr évolué au fur et à mesure en fonction des problèmes rencontrés), nous avons déterminé les classes essentielles au projet en fonction du sujet.

Nous avons une structure de projet qui ressemblait plus ou moins à cela :



Nous nous sommes réparti les premières class et la nous avons commencé à changer ce que nous avions prévu. Par exemple, au départ les tuiles avaient un int pour les représenter en fonction de leur couleur et des coordonnées dans le sac (pour les récupérer et donner une couleur en fonction d'une position), mais nous nous sommes rendu compte que les coordonnées ne caractérisaient pas vraiment une tuile et nous nous sommes tournés vers les énumérations beaucoup plus simples d'utilisation.

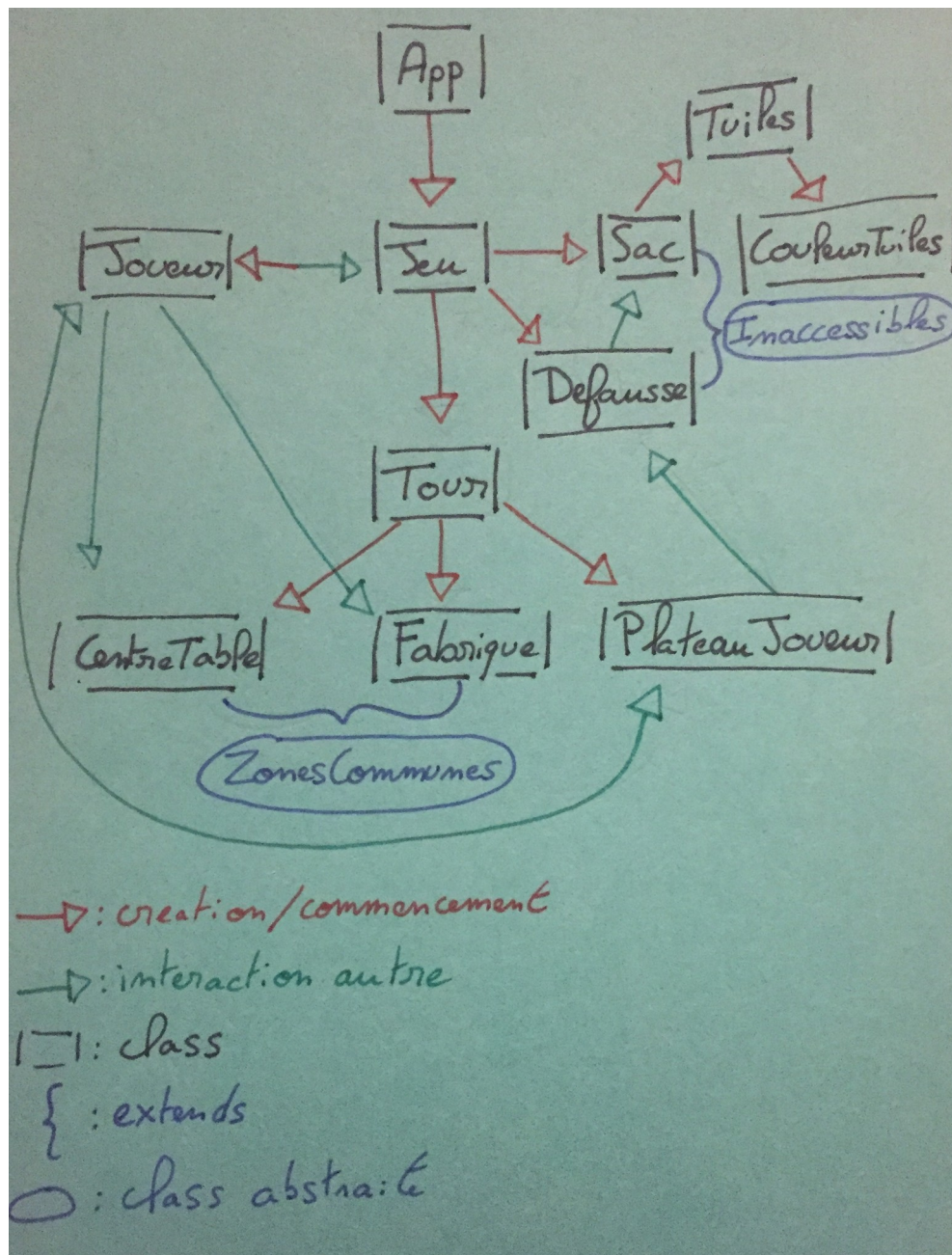
Nous avons également pensé à faire des class abstraites pour regrouper les class qui ont le même comportement comme Sac et Défausse qui ont la même structure et beaucoup de fonctions identiques il était donc plus simple de créer des class abstraites avec toutes ces fonctions et de faire des extends.

Une fois les class de base finis nous avons commencé à faire des class auxquelles nous n'avions pas pensé comme par exemple séparer toutes les parties du plateau en faisant :

- une grande partie avec la partie où les joueurs posent leurs tuiles (leur plateau personnel)
- une partie centre de la table où les tuiles vont lorsqu'elles ne sont pas prises
- une partie fabriques

En faisant cela, nous évitions d'avoir énormément de fonctions dans une seule class et de mieux se retrouver.

Une fois la partie terminal sans extension finie qui a cette structure :



(*dans le schéma le sac interagit avec les fabriques puisqu'on pioche des tuiles du sac pour les mettre dans les fabriques)

Nous avons décidé de prendre soit les interfaces graphiques soit les extensions et nous avons travaillé chacun de notre côté. Cette partie nous a pris beaucoup de temps et nous n'avions plus énormément de temps. Nous avons réussi à finir les interfaces graphiques fonctionnelles mais les extensions ne l'étaient pas encore. Nous nous sommes mis à deux dessus pour essayer de les régler mais malheureusement nous n'avions plus assez de temps et nous avons préféré ne pas rendre les extensions toute suite pour ne pas faire buguer tout le reste. Nous n'avions pas non plus assez de

temps pour faire l'interface graphique des extensions puisque celles-ci n'étaient pas fonctionnelles.

II. Problèmes rencontrés

Un des premiers problèmes que nous avons rencontrés a été comment représenter une tuile car avec juste un string ou juste un int ça n'était pas pratique car soit il fallait faire beaucoup de `.equals(String s)` soit pour l'affichage on affichait des int et ça n'était vraiment pas clair. Mais nous avons un peu cherché et nous avons trouvé les énumérations qui permettent de donner un int et un string a nos tuiles.

Un autre problème que nous avons eu a été avec les tuiles vides qui à la base étaient juste des null, mais en faisant une fonction pour les interfaces graphiques nous avons un problème avec un switch car on ne pouvait pas récupérer une couleur d'une case null donc c'était plus simple de mettre des tuiles vides qui retournent une couleur que l'on considère comme null.

Il y a eu aussi les interactions de la souris à gérer pour pouvoir déplacer des tuiles dans l'interface graphique, et que cela soit le plus lisible pour l'utilisateur.

Nous avons eu un autre problème au moment de faire les extensions car comme nous avons commencé par faire la version sans extension du projet la structure du projet n'était pas compatible avec les extensions, il y avait donc beaucoup de choses à changer et beaucoup de bugs, cela nous a mis très en retard

III. Pistes d'améliorations

Bien que ce projet soit fonctionnel il y a évidemment des façons de l'améliorer qui auraient pu être réalisées avec plus de temps. Comme par exemple pour l'extension du mur vide, il nous était demandé de faire attention à ce que le joueur ne soit pas bloqué non-intentionnellement. Pour cela nous affichons un message prévenant le joueur qu'en posant sa tuile à cet endroit il bloque une autre couleur et il peut alors renoncer, mais il est vrai que nous aurions put bloquer cette case pour que le joueur ne soit jamais coincé.

Nous aurions aussi pu faire un affichage plus beau, en mettant des images à la place des couleurs sur les tuiles ou en mettant la possibilité de changer le fond. Nous aurions aussi pu avoir un affichage moins brut que juste des carrés très basiques.

L'affichage aurait aussi pu être un peu plus clair pour la version terminal car on ne voit pas toujours ce qu'on fait ou ce qu'on a fait et pour jouer ça n'est pas très pratique parce que pour savoir ce qu'on a fait il faut remonter alors qu'on pourrait tout afficher tout le temps mais cela risque de surcharger et on risque de se perdre.

Nous n'avons également pas eu le temps de finir les extensions, car puisque nous faisons l'interface graphique et les extensions en même temps, nous n'avons pas eu le temps à la fin de travailler à deux sur les problèmes des extensions. Mais nous pensons pouvoir finir les extensions (sans interface graphique) pour la soutenance. Pour implémenter les Jokers, nous aurions créé une tuile Joker et lorsque le joueur la prend nous lui demandons s'il veut prendre plus de tuiles. Et pour les placers cela n'aurait rien changé, et pour implémenter les murs incolores nous n'aurions pas utilisé le modèle final et nous demanderions au joueur où il veut placer les tuiles sur le mur.

Conclusion :

Ce projet nous a permis d'appliquer les connaissances vues en cours ce semestre, même si nous n'avons pas tout utilisé. Nous avons également pu expérimenter les interfaces graphiques, tester des choses ce qui nous a appris pas mal de choses. On a pu voir qu'on pouvait passer d'une version terminal très peu ergonomique à une version avec une interface graphique en utilisant les bons outils. Mais nous avons également remarqué que nous gérons assez mal notre temps et que nous n'avons peut être pas assez de connaissances pour faire les interfaces graphiques des extensions.