

# Rapport Projet C++



## I – Introduction

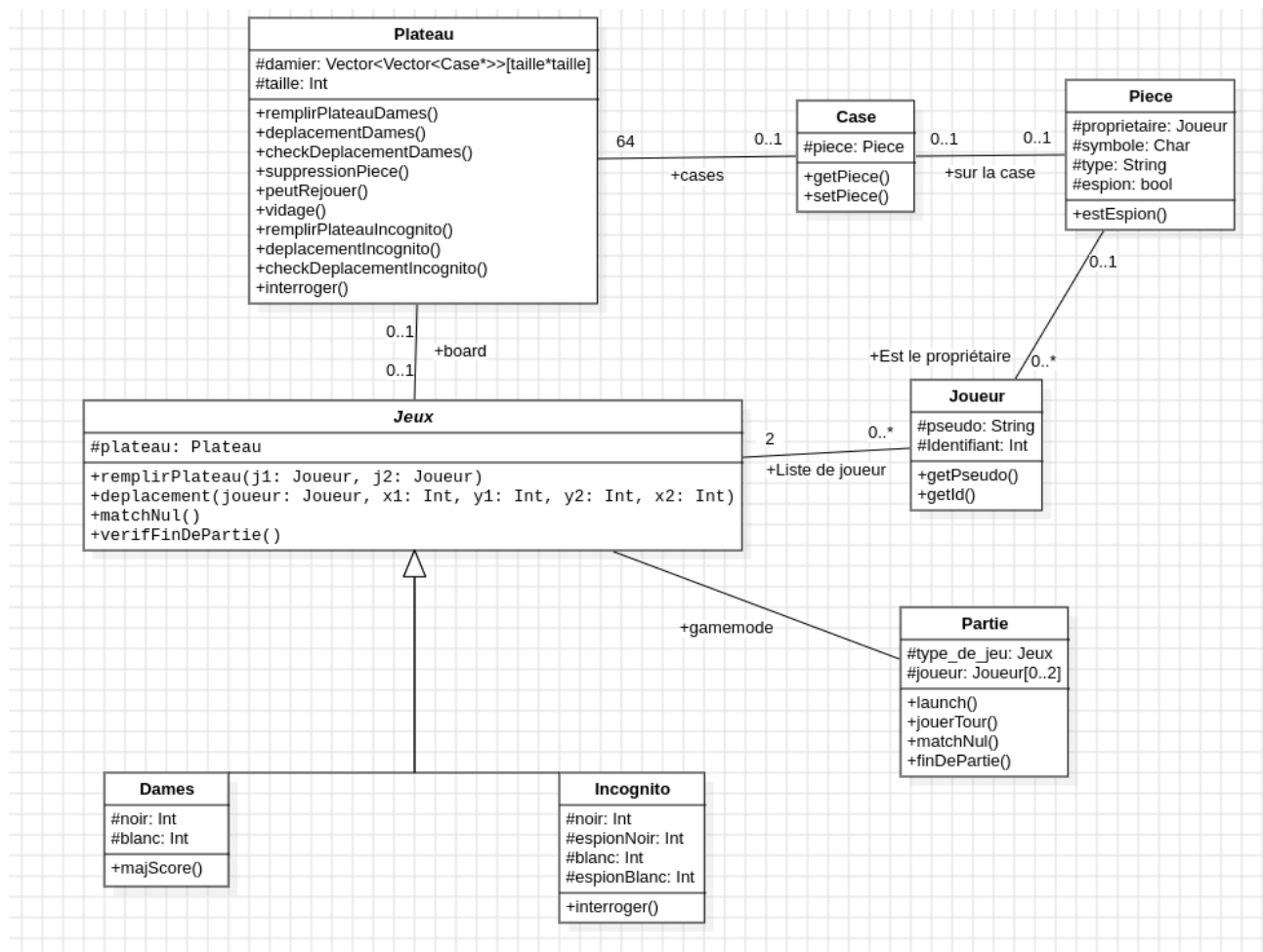
Nous tenions d'abord à nous excuser au sujet du projet, il n'est malheureusement loin d'être fini, seulement deux jeux sur les trois ont été codés, et il n'y a pas d'interface graphique. Nous avons manqué de temps, avec plusieurs projets en simultané notamment, nous n'avons pas bien mesuré le travail qu'il nous restait à faire, et nous avons fini par être dépassés. Toutes nos excuses.

Dans ce rapport, nous rajouterons une partie concernant ce que nous voulions faire, pour pas que cela soit trop léger, et pour que nous puissions en discuter pendant la soutenance.

## II – Ce qui a été traité

Les deux jeux dont nous nous sommes occupés sont le jeu de dames de base, et le jeu Incognito. Le jeu de dames était un choix assez évident pour nous, c'est un peu le jeu de « base » avec les échecs quand on imagine un jeu de plateau. Presque tous les aspects du jeu ont été implémentés, il manque le déplacement d'une dame qui enchaîne les prises, et l'amélioration du déplacement du pion quand il enchaîne les prises également.

Concernant le jeu Incognito, il nous intéressait d'avantage que son homologue Butin. Pour celui-ci, tout le contenu a été implémenté, il ne manque rien de particulier dans le concept du jeu. Il y a quand même des améliorations à faire, mais nous en reparlerons dans la partie IV.



Voici le diagramme de notre projet actuel. Dans la partie V, nous présenterons une idée de diagramme UML que nous aurions pu faire pour un projet en bon et du forme.

### III – Aspects significatifs

Concernant le jeu de dames, il n'y a pas grand-chose à dire malheureusement. Les déplacements généraux ont été assez compliqué à implémenter, surtout qu'il y a beaucoup de choses à prendre en compte (si c'est un pion, une dame, s'il peut enchaîner les coups, ou encore la suppression des pièces qui doit se faire uniquement à la fin du déplacement du tour). Pour Incognito, la fonction d'interrogation est le point marquant de ce jeu.

### IV – Problème connu

Pour le jeu de dames, les déplacements ne sont pas totalement implémentés, comme il a été dit précédemment. Il manque le déplacement d'une dame qui enchaîne les prises ; actuellement une dame ne peut jouer qu'une seule fois, puis elle doit s'arrêter.

Concernant le déplacement du pion, il ne peut pas choisir sur quelle case il va finir s'il doit enchaîner les prises : en effet, seul le premier déplacement compte, et si son premier coup est une prise, et qu'il peut enchaîner les prises, il va enchaîner les prises à chaque fois qu'il peut le faire. Si deux prises sont possibles, alors il va choisir la première prise qu'il peut faire.

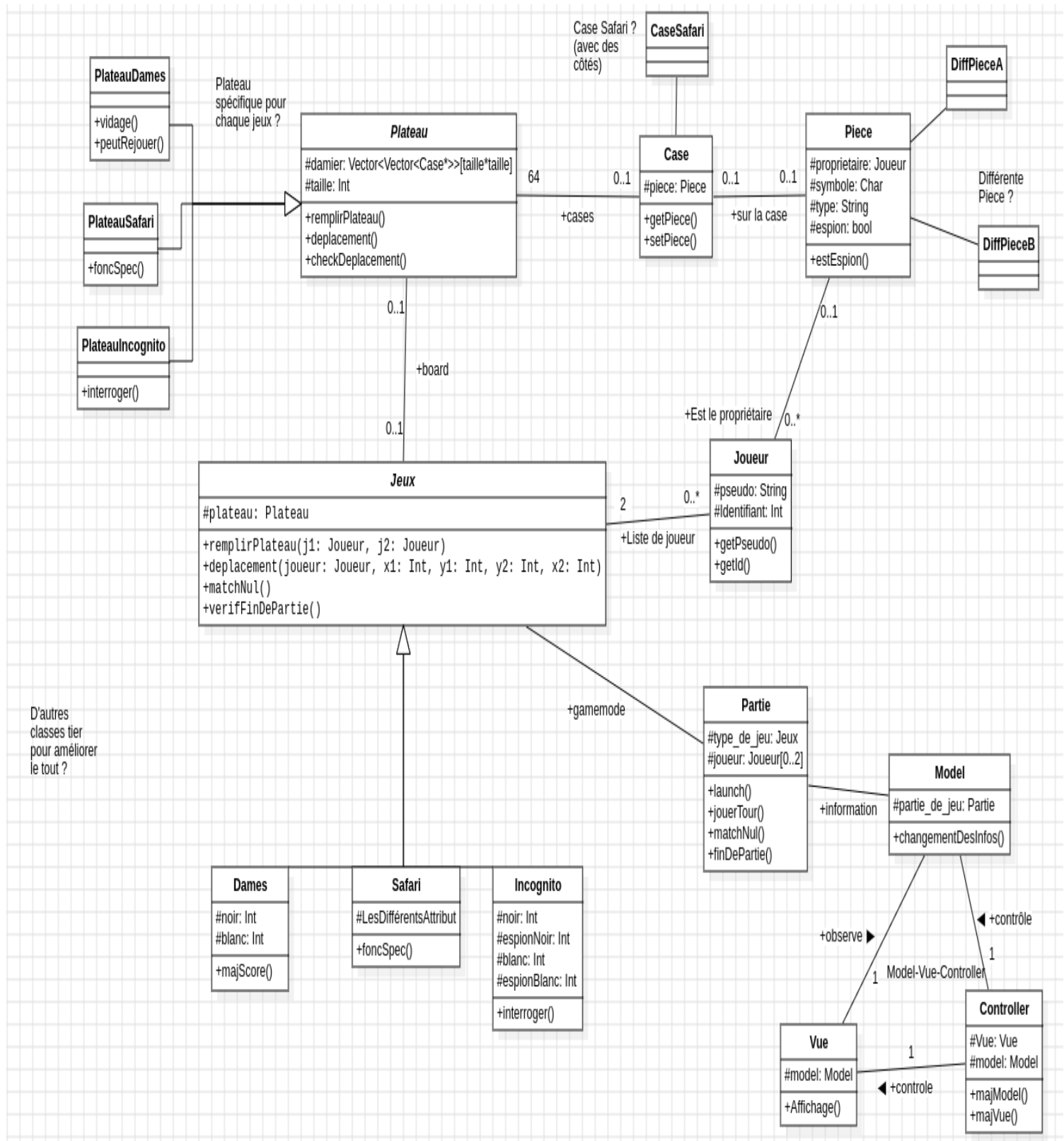
Pour résoudre ce problème, il aurait plutôt du refaire le principe même de la fonction de déplacement de pièces de dames : actuellement, le joueur choisi un déplacement possible pour une pièce (un pion en diagonale de 1 ou 2 de longueurs, une dame toute la diagonale s'il peut le faire) alors qu'il aurait plutôt du fallu calculer à l'avance où devrait atterrir la pièce. Une fonction récursive aurait été plus adapté, sûrement.

Pour Incognito, le problème réside essentiellement sur les cas à gérer à chaque fois : il y a beaucoup de cas à gérer, et il aurait fallu trouver un moyen plus propre d'accomplir ce que doit faire l'interrogation.

De manière générale, l'interface est « lisible » (ce n'est pas au niveau d'une interface graphique propre, bien sûr), mais nous voudrions rendre cela un peu plus lisible, notamment pour le jeu de dames, il est difficile de savoir pour un joueur si son pion est une dame.

Enfin, le code manque de « factorisation » : on aurait pu plusieurs fois factoriser le code, en ajoutant d'avantage d'héritage par exemple, c'est ce que nous voulions faire à la base après avoir développé les trois jeux.

### V – Ce que nous aurions voulu faire



Voici une idée de diagramme que nous imaginions pour le code. Bien sûr ce diagramme n'est pas une finalité : il aurait sûrement évolué dans un sens, en fonction des problèmes

rencontrés pendant l'implémentation.

Bien sûr, déjà nous aurions voulu finir le projet. Le prochaine objectif était de faire le jeu Safari, puis l'interface graphique, histoire de compléter le projet. Ensuite, nous aurions factorisé le projet, formaliser les concepts que nous aurions étudiés pour ce projet.

Il y a beaucoup d'aspect qui ont été utilisé pour un début, mais qui n'ont pas été modifiés par la suite. Par exemple, nous voulions utiliser les exceptions que nous avons rapidement abordées en cours cette année pour certains cas, par exemple la gestion des déplacements. En créant différent type d'exceptions, on aurait pu identifier pourquoi le joueur ne pouvait pas se déplacer là où il le souhaitait (dépassement du plateau, déplacement impossible si ce n'est pas en diagonale etc... chaque règle qui autorise ou non, et qui le vérifie, aurait pu être une exception. On aurait par la suite récupéré ces exceptions, et affiché au joueur ce qui n'allait pas).

De même, nous n'avons pas utilisé de enum, alors que cela aurait été beaucoup plus simple, notamment pour choisir quel type de pièce.

Une meilleur gestion des destructeurs n'auraient pas été négligeable non plus, de l'héritage, mais cela était prévu de faire quand nous aurions dû factoriser le code, de même que pour la généricité.