*Participants :*

* *Zackary SAADA*
* *Mohamed HAMOUTI*

*Groupe de TD : Ours*

# Echecs

# Déroulement du projet

*Tableau d’avancement : https://docs.google.com/spreadsheets/d/1OUcrdiWf3KS5toR-u\_YoVaG7gYSFpbO-0-FIea8\_ZmQ/edit?usp=sharing*

Dans le cadre de ce projet tutoré de jeu d’échecs, nous avons réalisé les travaux et avancements suivants.

Nous avons opté avec un autre groupe pour l’élaboration d’un plan dans lequel chacune des tâches ont été réparties au préalable. Nous avons fait ce choix suite à l’importance d’une bonne gestion de projet, que vous nous avez suggérée. Nous nous sommes vite rendus compte que malgré une bonne organisation, notre manque d'expérience dans ce genre de projets nous a causé des préjudices.

Afin de prévoir les erreurs et les difficultés que l’on pourrait rencontrer, nous avons commencé notre projet par l’élaboration d’ébauches en UML. On a donc constitué une première version d’un diagramme de classe et d’un diagramme de cas d’utilisation. Ceux-ci nous ont permis d’y voir un peu plus clair quant à la première version de notre code avec des bases solides sur nos connaissances, et sachant qu’on savait par quoi commencer et surtout vers quel objectif se diriger.

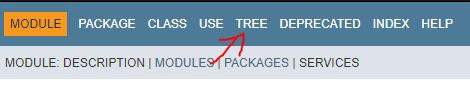
Les débuts étaient assez simples, nous avons commencé par la classe Pièce qui est la classe mère de Pion, Fou, Cavalier, etc. Ainsi, une fois ces classes mises de côté, nous pourrions commencer à nous documenter sur des notions plus problématiques. On a donc revu la répartition des tâches en se basant sur ce que l’on avait déjà réalisé.

Au fur et à mesure que l’on avançait, on se rendait compte que le travail des classes filles de Pièce prenait beaucoup plus de temps que prévu et on commença à regretter de ne pas s’y être pris en avance. Non pas par nonchalance, mais parce que les partiels et le début des cours en Interface Homme-Machine, nous ont énormément limités dans le temps qu’on pouvait attribuer au projet. Ceci explique notamment les énormes différences entre les dates prévues et effectuées dans le tableau d’avancement.

Finalement, nous nous sommes débrouillés pour vous proposer notre version et surtout notre vision de ce projet.

Au lieu de vous décrire chaque méthode et son comportement, nous vous redirigeons vers **notre propre Javadoc du projet**, dans laquelle tout sera présenté. (constructeurs et attributs compris) Quelques méthodes pas encore implémentées à l’écriture de ce texte seront signalées par un “TODO”.

Le lien: <https://mormox.github.io/JAVA-DOC-ECHECS-_-SAADA-HAMOUTI/> (cliquer sur “TREE” pour commencer la navigation)



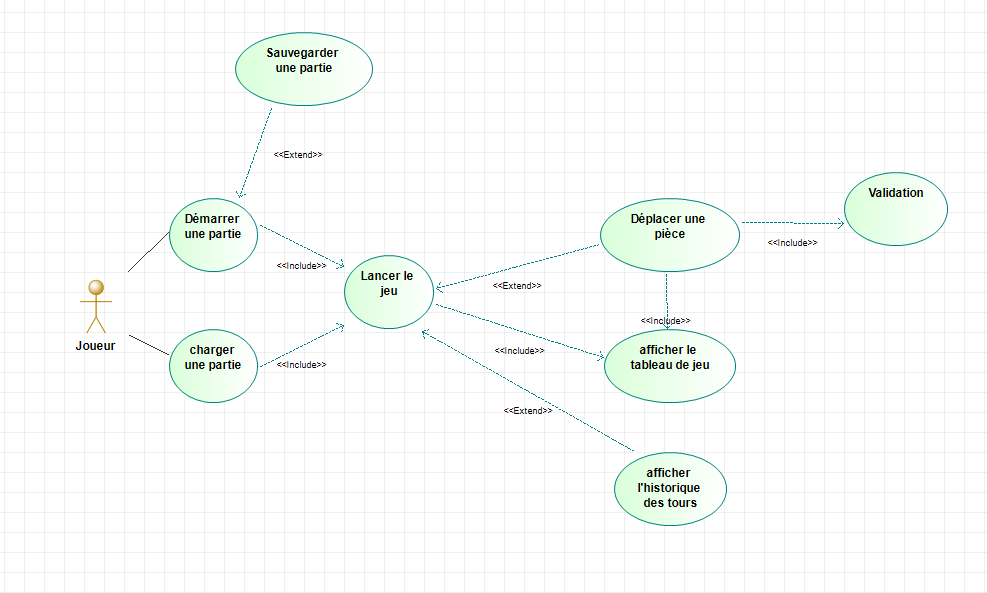
Vous pourrez consulter en annexe notre diagramme de cas d’utilisation, ainsi que les deux versions initiale et finale de notre diagramme de classe. Vous remarquerez qu’on a décidé de passer outre la classe Case. Celle-ci nous semblait être une perte de temps à coder, car n'ajoute que peu de fonctionnalités au projet, que nous avons pu coder directement dans un double tableau de Pieces. Nous nous en sommes très bien sortis avec cette décision, et trouvons qu’elle nous a été très bénéfique !

Nous espérons que le travail et le soin apporté au projet vous plaira, même si le rendu final n’est pas exactement à la hauteur de nos espérances. Nous voulions implémenter une interface, mais faute de temps cela sera impossible.

Merci de nous avoir lus !

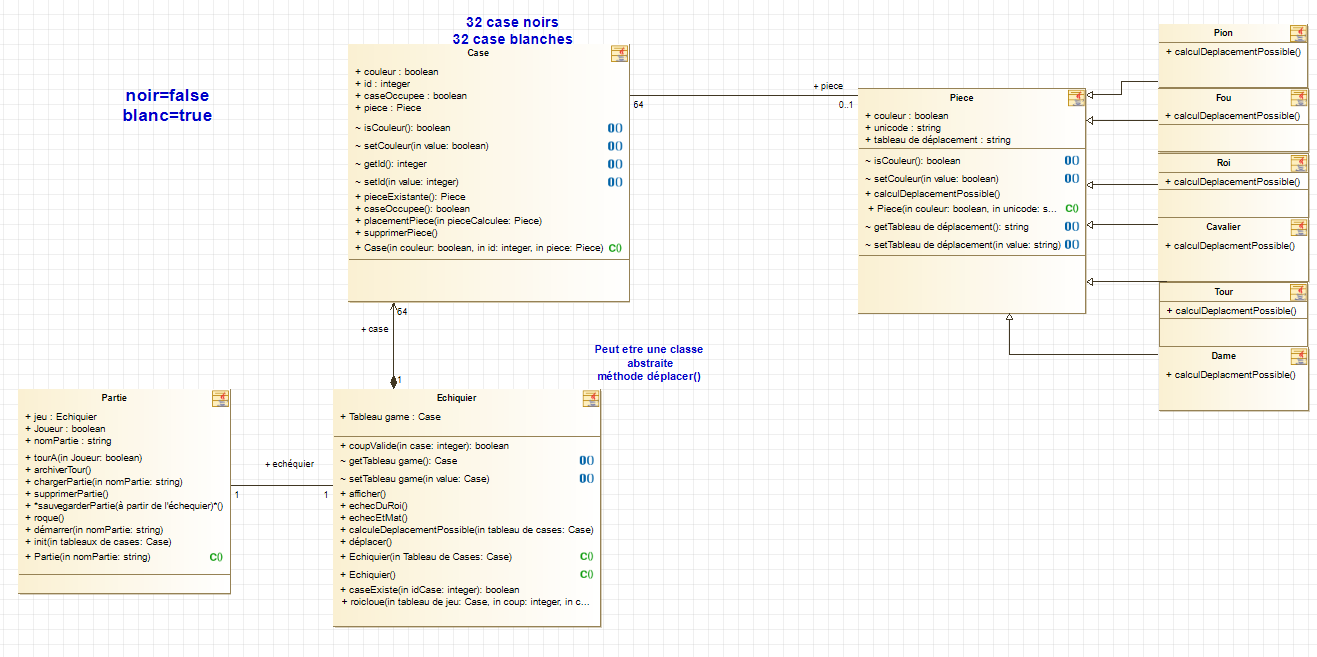
# Diagramme de cas d’utilisation

**Version 1 :**



# Diagramme de classes

**Version 1 :**



**Version 2 (beaucoup plus propre) :**

