

Olivier Cangrand  
Pierre Lethuillier

## **Rapport de projet de réseau : Morpion à l'aveugle**

### 1) Comment jouer ?

Lancer `main_propre.py` sur, le serveur de jeu est lancé sur le réseau.  
Pour les deux joueurs, il faut entrer la commande `nc localhost 7777` sur deux terminaux du réseau, la partie commence dès que deux joueurs se sont connectés.

### 2) Fonctionnalités :

Notre projet répond à l'attente principale du sujet, à savoir le morpion à l'aveugle en réseau entre deux clients, le jeu fonctionne et annonce le résultat aux joueurs à la fin de la partie, « You WIN », « You LOOSE » ou « DRAW ».

Nous avons essayé de gérer les nouveaux clients qui se connectent après les deux joueurs enfin de les faire devenir spectateurs mais nous avons rencontré un problème que nous n'avons pas réussi à résoudre.  
En effet, lors de la détection d'un nouveau client, nous vérifions si le joueur 1 et le joueur 2 ne sont pas vide, et dans ce cas-là nous ajoutons le client qui vient de se connecter à une liste de spectateur à qui nous enverront par la suite la grille complète au cours du jeu afin qu'il puisse suivre avec la vue des coups des deux joueurs. Cette partie fonctionne mais lors de l'arrivée d'un nouveau client le jeu se met à l'arrêt et c'est impossible de le reprendre. Nous n'avons malheureusement pas compris d'où vient le problème.

Nous avons également tenter de mettre en place un système afin de rejouer après la fin d'une partie mais ici aussi nous n'avons pas réussi à résoudre un problème qui empêche le bon fonctionnement du 'rejouer'.

A la fin de la partie, après avoir envoyer les résultats aux deux joueurs, nous demandons à un client (on aurait demandé aux deux si ça avait marché) si il veut rejouer en lui demandant de taper 'y' ou rejouer ou 'n' pour ne pas rejouer. Si il tape 'y' on reset les grilles ainsi que toutes les variables nécessaires et on rejoue. Le fait de remettre le jeu à son état initial fonctionne mais malheureusement lorsqu'on demande au premier joueur de rentrer la case qu'il veut jouer, la socket prend en compte la réponse qu'il avait donné avant, à savoir le 'y' et donc le jeu plante car ce n'est pas ce qu'il attend. Nous avons cherché beaucoup de chose pour contrer ce problème, comme essayer de mettre toute la partie rejouer dans une fonction à part, ou bien un moyen de vider les informations de la socket après la partie rejouer mais rien n'a fonctionné.

Nous sommes un peu déçus de ne proposer aucune fonctionnalités supplémentaires mais surtout frustrés car nous savons très bien qu'il suffit probablement de changer une petite chose pour que notre replay et nos clients en spectateurs puissent fonctionner.

### 3) Github

Nous aimerions rapidement parler de GitHub ici car comme vous avez peut-être pu le voir, les commit de somebugis (Pierre) sont marqués comme « committed with OCangrand ». C'est parce que lors du clonage du projet sur la session de Pierre, nous l'avons fait avec le compte Ocangrand (Olivier) et donc à chaque commit c'était automatiquement marqué comme étant un commit de

Olivier. Pierre a donc changé l’auteur de ses commits (avec `–author`) pour que ça s’affiche bien à son nom.