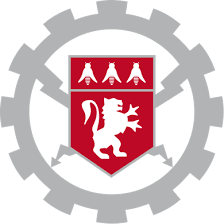
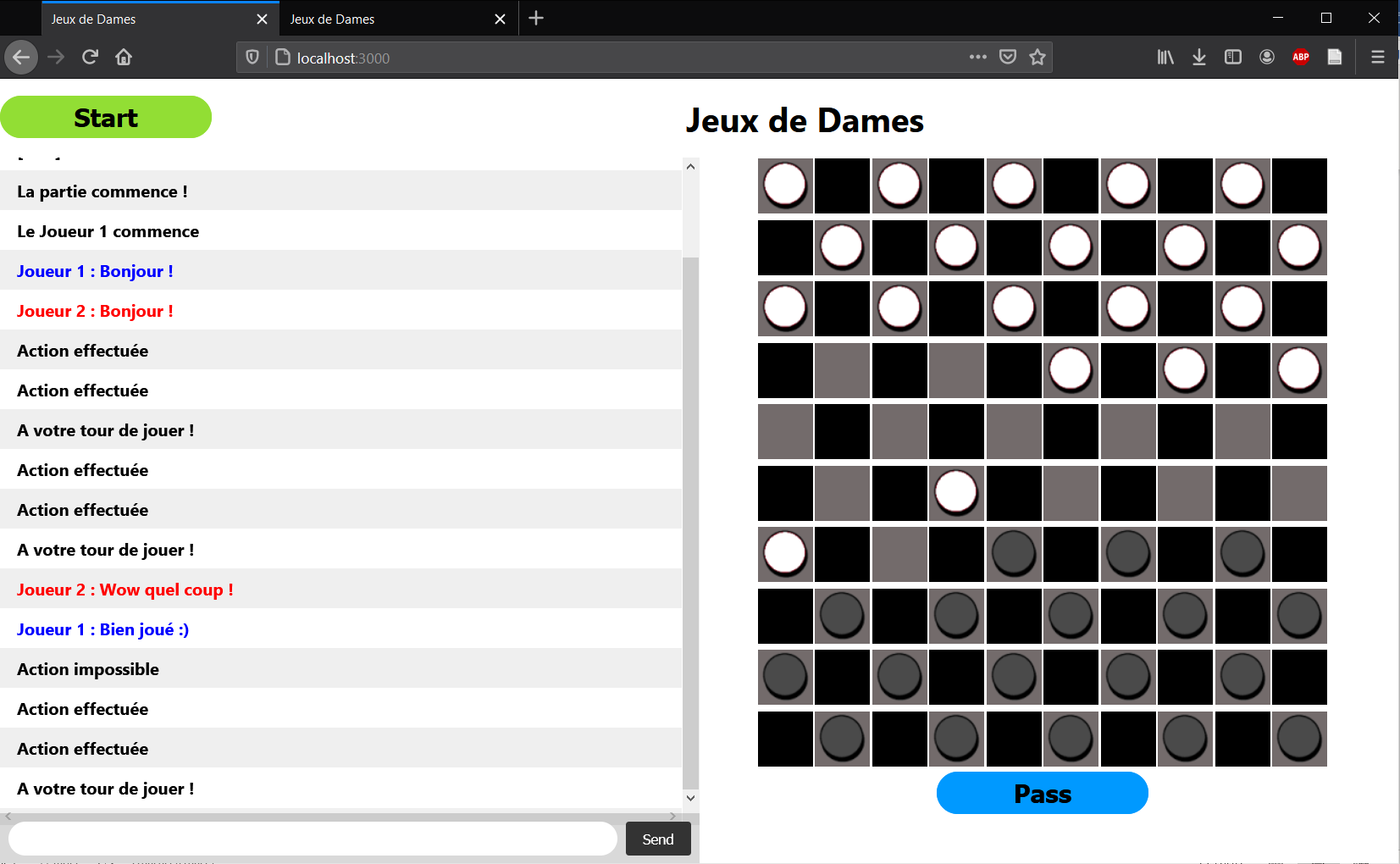
**Gaëtan Chanet**

**ELC-D3 - Applications Web**

**Projet Final**

**Jeu de Dame multijoueur**

**Présentation**



**Introduction**

Le présent projet consiste en la création d’un jeu de dames multijoueur, fonctionnel, reposant sur l’implémentation d’un serveur javascript fonctionnant sous Node.js. Ce serveur utilise lui-même les modules Express et Socket.io, lui permettant de répondre en asynchrone aux différentes requêtes des deux joueurs présents. L’application web ainsi créée permet aux deux joueurs de s’affronter lors d’une partie gérée par le serveur, l’interface est conçue de telle sorte que les joueurs puissent communiquer via un chat lors de leur partie.

**Cahier des charges**

Ce projet à été conçu selon une logique d’efficacité et d’adaptation. Le serveur et les interfaces doivent pouvoir gérer les différents comportements des joueurs de sorte à éviter au maximum le risque de bug, notamment à la déconnexion ou la reconnexion de ceux-ci. Parallèlement à cela, le serveur gère le jeu et le chat, dans lequel les messages des joueur ou les informations parties sont affichés. L’application doit donc garantir un jeu sans bug couplé à un système de chat fonctionnel.

La partie back-end doit ainsi pouvoir gérer en asynchrone les différentes communications entre les joueurs et celui-ci. Cela est permis par le module Socket.io de Node.js qui doit garantir un accès direct et en temps réel des informations du serveur. Chaque coup, action ou message est géré par le serveur et doit selon l’état de la partie renvoyer une réponse cohérente en conservant son fonctionnement. Pour pouvoir gérer plus simplement la partie, le serveur utilise l’implémentation d’un objet d’une classe Game, créée sur un fichier annexe, par lequel le cœur de la partie est géré.

Concernant la partie front-end, celle-ci doit sur le plan technique pouvoir communiquer de manière efficace avec le serveur, notamment via des requêtes socket envoyées au serveur pour les différentes actions du joueur. La page n’a donc jamais besoin d’être rechargée. Sur le plan graphique, l’interface utilisateur est découpée en une partie gauche, contenant la partie chat avec les messages et les informations partie, et une partie droite contenant l’interface interactive du damier. Les pions sont alors déplacés via un clic souris du pion sélectionné suivi d’un second clic indiquant la case cible pour du déplacement. Si un coup est impossible pour une quelconque raison, un message est affiché dans le chat indiquant au joueur pourquoi le coup n’est pas permis. La partie graphique contient également deux boutons permettant respectivement de commencer et de passer son tour au cours de la partie.

L’ensemble de ces deux parties doit en particulier pouvoir gérer les comportements non permis, les déconnexions et reconnexions des joueurs et afficher correctement les données de la partie en cours.

**Mode d’emploi**

Pour jouer une partie sur l’application développée, les deux joueurs doivent tout d’abord se connecter sur l’adresse du serveur (localhost:3000). Ensuite, ceux-ci auront affichés l’interface de l’application. Pour commencer la partie, les joueurs doivent tous deux avoir appuyé sur le bouton « Start », ce qui déclenchera le lancement de la partie avec un message dans leur chat.

Une fois la partie commencée, le joueur 1 commence (le joueur 1 est le premier joueur connecté et le suivant est le joueur 2) et effectue la première action en cliquant sur le pion devant être déplacé puis sur la case cible. Si le coup est valide, celui-ci est effectué et le joueur 2 peux continuer en faisant de même. Cependant, le jeu de dame permet en particulier de pouvoir manger plusieurs pions ennemis en un seul coup si cela est possible. Ainsi, lorsque d’un joueur mange un pion adverse, le tour n’est pas automatiquement passé et celui-ci doit appuyer sur le bouton passer son tour lorsque plus un pion ne peut être mangé. Ce bouton permet également de passer son tour si un des joueurs le veut.

Le jeu suit alors les règles habituelles du jeu de Dame, lorsqu’un des joueurs a gagné, la partie s’arrête automatiquement et le joueur gagnant est affiché dans le chat. Si un des joueur se déconnecte, il récupère les données du jeu lors de sans reconnexion et la partie est arrêtée tant que le joueur n’a pas réappuyé sur « Start ».

**Instructions d’installation**

Pour pouvoir fonctionner,