Cégep du Vieux Montréal

Rapport mi-mandat

Projet synthèse

**Réalisé par**

Gabriel Veilleux

Arslan Khaoua

Romeo Barraza

# Les tâches restant à faire

## Version Vanille

1. La condition de la fin du jeu : Le jeu est fini lorsque l’un des rois est capturé. Il faut vérifier à chaque coup si l’un des rois est capturé. Si c’est le cas, afficher un AlertDialog qui indiquera le joueur gagnant de la partie.
2. La promotion des pièces : Lorsqu’une pièce quelconque (sauf le roi) se trouve dans l’une des trois rangées opposé du joueur, elle peut promouvoir. La promotion est dans la plupart des cas optionnelle. Elle est obligatoire si la pièce arrive dans une case où elle ne possède plus de coups valides. Voici les pièces affectées :

a) Le pion : Sa promotion est obligatoire si elle arrive à la dernière rangée du côte opposé

b) La lance : Sa promotion suit la même logique de celle du pion.

c) Le chevalier : Sa promotion est obligatoire si elle arrive à la dernière ou à l’avant-dernière rangée du côté opposé

Lorsque une pièce arrive aux trois dernières rangées et que la promotion est optionnelle, afficher un AlertDialog qui montre au joueur deux images des deux pièces : l’image de la pièce non promue et une autre avec la pièce promue. Si le joueur choisit l’image de la pièce non promue, alors la pièce reste telle quelle, s’il choisit l’autre image la pièce est promue. Si la pièce à promouvoir est un pion, une lance ou chevalier, appliquer les règles décrites ci-haut.

1. Les conditions du parachutage des pièces capturées

Cela consiste à placer les pièces capturés dans l’échiquier. Jusqu’à maintenant, l’application peut techniquement parachuter mais sans suivre les règles du jeu. Les conditions à tenir en compte sont :

- Ne pas parachuter sur une pièce : Il faut voir si la case sélectionné est vide.

- Ne pas permettre la promotion dans le même coup que la pièce est parachutée : Lorsque une pièce est parachutée, ne jamais déclencher une promotion si elle se trouve dans les trois dernières rangées.

- Restreindre les cases pour certaines pieces : Un pièce peut être parachuté seulement s’il possède des coups valides dans la case oú elle se strouve. Voici les pièces affectés :

a) Le pion : Ne peut pas parachuter dans la dernière rangée du côte opposé du joueur ni dans la même colonne où se trouve un autre pion du joueur ni dans la même case où il peut faire échec et mat au roi adversaire.

b) La lance : Ne peut pas parachuter dans la dernière rangée du côte opposé du joueur.

c) Le chevalier : Ne peut pas parachuter dans les deux dernières rangées du côte opposé.

1. Les fonctions d’évaluation : pour que notre algorithme Minimax sois précis nous avons besoin de différente fonction qui vont évaluer diffèrent paramètre de la partie actuelle ce qui vas permettre à l’IA de mieux savoir qu’elle coups et meilleurs qu’un autre.

Pour l’instant nous avons 4 fonctions d’évaluation sur les 8 que l’on veut.

Les fonctions manquantes sont:

1) Good shape and bad shape : permet d’évaluer si les positions des pièces permettent une liberté de mouvement ou fait partie d’une formation d’attaquer ou de défense, si des pièces sont bloquées (ne peuvent pas bouger), si la position des pièces un peu d’impact sur le jeu

2) Distance to the kings: permet d’évaluer les pièces ennemies proche du roi, les pièces ennemies éloigner du roi, les pièces défensives proche du roi

3) Controlled squares : permet d’évaluer le contrôle de son propre camp, le contrôle du centre et les cases autours du roi

4) Pieces in hand : le type de pièces, potentiel de parachutage

1. L’historique de partie : quand une partie est fini celle-ci est sauvegarde et il est possible de revisionner les parties terminer nous permettant de reculer ou d’avancer coups par coups
2. Le Savepoint : permet de sauvegarder l’état actuelle d’une partie en cours pour pouvoir la continuer plus tard. Il faut surtout tenir en compte les nombres de pièces capturés pour chaque joueur et les positions des pièces dans l’échiquier.
3. Stratégies d’évaluation : Interface java qui comprend un nom et une méthode qui combine les stratégies d’évaluations d’échiquier. L’évaluation du minimax vas ensuite se servir de la classe désirer pour effectuer l’évaluation pour introduire un biais à l’IA.
4. Choix de difficulté : Proposition d’un profile de difficulté de l’IA à l’utilisateur avant le début d’une partie.
5. Parachutage pour le Minimax : Finir l’implémentation dans le générateur de déplacements pour que le Minimax puisse parachuter des pièces sur l’échiquier si possibles.

## Version Chocolat

1. Garder l’historique des coups : Une fois qu’une partie est finie, enregistrer les coups joués et le nom des deux joueurs. La donnée gardant les coups est un string qui représentent le déplacement des pièces. Dans un écran à part, l’échiquier et deux boutons seront affichés. Les boutons permettent de reculer et avancer un coup.
2. Effets sonores : Ajouter des effets sonores lorsque certaines actions sont effectuées lors d’une partie.
   * L’utilisateur ou Minimax joue un coup ;
   * Une pièce est capturée ;
   * Une pièce est parachutée ;
   * Lorsqu’une pièce est promue ;
   * Lorsque la partie est gagnée ;
   * Lorsque la partie est terminée ;
3. Thème de l’application : Introduire des thèmes a l’application pour changer l’apparence des pièces, l’échiquier et la couleur de fond.