

WinEchek

Dossier préliminaire

Mathis DELOGE, Antoine PETOT, Ange PICARD
Arthur CARCHI, Lucas FOUGEROUSE, Vincent DERECLLENNE

1 Introduction

1.1 Besoins

WinEchek est une application Windows qui permettra de jouer aux échecs sur un ordinateur. Les parties pourront être dispensées de différentes manières. La première sera de jouer à deux joueurs humains sur le même ordinateur. La seconde permettra à deux joueurs humains de jouer en réseau (local) sur la même partie. Enfin, la dernière proposera à un joueur seul de dispenser une partie contre l'ordinateur qui sera doté d'une intelligence artificielle.

1.2 Objectif

Le but du projet n'est pas forcément d'implémenter le plus de fonctionnalités possibles, ni même de proposer quelque chose d'innovant. En effet, ce type d'application étant assez difficile à réaliser, les projets existants seront forcément plus intéressants que notre réalisation finale. Notre objectif est donc mettre en pratique, à travers la réalisation d'une application complexe, les différents domaines abordés dans notre formation (IHM, CO, POO, Réseau) mais également de découvrir les bases de la programmation d'intelligences artificielles. Pour résumer, nous réalisons cette application plus dans un but pédagogique que pour combler un besoin (les solutions existantes sont très performantes, et la plupart proposent déjà nos fonctionnalités).

1.3 Fonctionnalités principales

Jouer une partie

- Jouer en réseau
 - Rejoindre une partie
 - Moyenne Rechercher une partie
 - Moyenne Héberger une partie
- Jouer contre une IA
 - Basse Sélectionner le niveau de difficulté
- Jouer sur la même machine
 - Haute Afficher quel joueur doit jouer
- Gérer la partie
 - Moyenne Annuler un coup
 - Moyenne Rejouer un coup
 - Moyenne Afficher l'historique
 - Moyenne Afficher les pièces prises
 - Moyenne Afficher le nombre de tous
 - Basse Afficher le timer
 - Basse Sauvegarder une partie
 - Basse Charger une partie
 - Basse Jouer sans la souris
- Jouer un tour
 - Haute Déplacer une pièce
 - Moyenne Afficher les déplacements possibles
 - Haute Promouvoir une pièce
 - Haute Prendre une pièce
 - Haute Roquer

2 Existant

2.1 Moteurs

Les moteurs d'échecs sont le cœur des jeux d'échecs. Ce sont eux qui gèrent toutes les contraintes des parties (comme vérifier si les déplacements sont valides). Ils simulent adversaire sur une partie grâce à un module d'intelligence artificielle intégré.

2.1.1 StockFish

<https://stockfishchess.org>

Stockfish est un programme d'échecs multiprocesseur open source développé par Tord Romstad, Marco Costalba, Joona Kiiski et Gary Linscott. Ce programme est libre et gratuit et est le meilleur logiciel d'échecs non-commercial au monde depuis mai 2014. Sur l'ensemble des logiciels, il est considéré comme un des meilleurs avec Komodo 9.3 et Houdini 4.

2.1.2 Komodo

<https://komodochess.com>

Komodo est un programme d'échecs commercial et non-libre créé par Don Dailey, Mark Lefler et Larry Kaufman en 2007. Il a été considéré comme le meilleur programme d'échecs en 2016.

2.1.3 GNU Chess

<https://www.gnu.org/software/chess>

GNU Chess est un logiciel libre de jeu d'échecs, sous les termes de la licence publique générale GNU, maintenu par la collaboration de développeurs. Ne disposant que d'une saisie des coups en ligne de commande, il peut être considéré comme un moteur d'échecs. Il est souvent utilisé avec un environnement graphique comme XBoard ou GlChess pour la 3D.

2.2 Superordinateurs

2.2.1 Deep Blue

https://fr.wikipedia.org/wiki/Deep_Blue

Deep Blue est un superordinateur conçu par IBM destiné à jouer aux échecs. Il a battu Garry Kasparov, un des meilleurs joueurs d'échecs au monde, en 1997. Ce fut un événement remarquable dans le monde de l'intelligence artificielle, car dès lors, aucun ordinateur n'avait réussi à battre un joueur humain aux échecs, et on pensait que ce ne serait pas possible avant un long moment.

2.2.2 Hydra

[https://fr.wikipedia.org/wiki/Hydra_\(ordinateur_d%27%C3%A9checs\)](https://fr.wikipedia.org/wiki/Hydra_(ordinateur_d%27%C3%A9checs))

Conçu en 2002, Hydra est un superordinateur destiné à dominer le monde des ordinateurs d'échecs et d'obtenir une victoire probante contre les humains en utilisant la force brute. Il était en 2005 l'ordinateur le plus fort en confrontation en temps réel avec un classement ELO évalué à plus de 3000. L'appellation "Hydra" représente la partie matérielle mais aussi logicielle.

2.2.3 Belle

[https://fr.wikipedia.org/wiki/Belle_\(ordinateur_d%27%C3%A9checs\)](https://fr.wikipedia.org/wiki/Belle_(ordinateur_d%27%C3%A9checs))

Belle est un programme d'échec ainsi que l'ordinateur sur lequel il tournait. Conçu dans les années 1970 et 1980, il avait pour seul but de jouer aux échecs. Belle est ainsi le programme le plus fort de son époque (aux échecs) avec un score ELO de 2250 et atteint le niveau de maître international en 1983. Belle pouvait grâce à son matériel dédié calculer environ 160 000 positions par seconde.

3 Solutions similaires

3.1 Jeux en ligne

3.1.1 Lichess

<https://fr.lichess.org>

Lichess est un jeu d'échecs par navigateur qui permet de faire des parties contre d'autres joueurs réels, une fois connecté sur le site, ou bien contre une IA avec différents niveaux de difficulté allant de 1 à 10. Le site propose aussi des tutoriels proposant une initiation aux échecs et différents entraînements en fonction de son niveau de maîtrise.

3.1.2 ShredderChess

<https://www.shredderchess.net>

Ce jeu d'échec en ligne ne propose que quelques fonctionnalités. Parmi celles-ci, nous avons la possibilité de démarrer une nouvelle partie avec les pions blancs ou avec les pions noirs, annuler un coup et pour finir choisir le niveau de l'ordinateur qui va jouer contre nous (Easy, Medium, Hard). Par rapport à la graphique, c'est une représentation très simpliste, en 2D avec une coloration des cases en bleu pour savoir à chaque tour quelles sont nos possibilités de mouvement.

3.1.3 Flash Chess

<http://www.stratozor.com/echeecs/jeu-echeecs-flash.php>

C'est un jeu d'échec en ligne basé sur la technologie flash. Développé dans les années 2000 afin de démontrer la puissance d'ActionScript, un langage de programmation utilisé pour des applications clientes. Parmi ses fonctionnalités, il ne permet uniquement d'afficher l'historique des coups et les mouvements possibles en cliquant sur une pièce. Il intègre également la possibilité de redémarrer une partie.

3.1.4 Flash Chess III

<http://www.lesjeuxgratuits.fr/jouer.php?n=73>

Ce jeu d'échec est la version améliorée de Flash Chess. En effet, il arbore un design tout en 3D et quelques fonctionnalités supplémentaires. Il propose de recommencer, sauvegarder ou reprendre une partie, annuler un coup et est également disponible en téléchargement pour application de bureau : SparkChess.

3.1.5 SparkChess en ligne

<https://www.sparkchess.com>

SparkChess en ligne est un jeu d'échec très complet sur navigateur. Tout d'abord, il propose un choix d'Intelligence Artificielle en fonction du niveau désiré avec une idée de mode histoire (certains personnages sont à débloqués au fur et à mesure du jeu). Dans cette version en ligne, ce jeu d'échec propose un historique des coups, un compteur de temps, la possibilité d'annuler ou coup, mais permet également de sauvegarder l'état d'une partie et d'en charger une.

3.2 Application de bureau

3.2.1 SparkChess hors ligne

<https://www.sparkchess.com/download.html>

Une version payante de SparkChess est disponible et permet de jouer hors ligne contre son ordinateur, mais également en réseau avec un chat intégré à l'application. Ce jeu d'échec très complet propose notamment des fonctions d'analyse de jeu à un moment donné, éditeur de jeu avec le format FEN mais également l'affichage des coups possibles ou même exporter le plateau en image PNG.

3.2.2 WJChess 3d

<https://fr.jeffprod.com/wjchess>

WJChess 3d est jeu d'échec pour pc fonctionnant sous Windows. Il inclut de nombreuses options et fonctionnalités et à la particularité d'avoir une partie graphique assez élaborée et modifiable à volonté. Ce logiciel ne permet qu'un mode 2 joueurs ou contre l'ordinateur, il intègre les possibilités de sauvegarder, reprendre une partie ou même créer une position grâce à l'importation / exportation des positions au format FEN. De plus, il respecte toutes les règles des échecs et propose plusieurs niveaux de difficulté contre l'IA avec un temps de recherche ou une profondeur fixe pour l'intelligence artificielle.

4 Technologies

4.1 Préambule

Pour le type d'application, nous avons opté pour une application de bureau. Le cycle de développement ainsi que les technologies nous correspondant plus qu'avec une application web. Pour des raisons de budget et de flexibilité, nous avons fait le choix d'utiliser une architecture client-serveur / client plutôt qu'une architecture client / serveur / client, cela nous permet de ne pas avoir à matérialisé un serveur et nous donnera l'occasion de découvrir l'utilisation de cette architecture.

4.2 Langage

Pour ce qui est du langage utilisé pour l'implémentation, nous avons évalué trois options, le C++, le C# et Java.

Le principal atout du C++ est la performance, en effet celui-ci permet de descendre plus bas niveau que le C# ou le Java. D'autre part, la bibliothèque standard est moins haut niveau. Les outils fournis en comparaison au Java/C# ne permettent pas un développement rapide ceux-ci étant plus performant, mais moins simple à l'utilisation ce qui dans le cadre de notre projet est peu pertinent, nous ne cherchons pas à réaliser un programme performant. De plus, il nous aurait fallu utiliser un framework comme Qt pour réaliser la graphique, et l'environnement de développement associé (Qt creator) est quelque peu contraignant.

Le Java quant à lui, présente l'avantage d'être plus haut niveau que le C++, d'avoir plusieurs librairies graphiques intégrées. Suite à la réalisation de plusieurs projets en Java (POO, Graphe, Modélisation), nous sommes familiarisés avec sont utilisation, il nous aurait été simple de l'utiliser. L'environnement de développement que nous utilisons (IntelliJ Idea) pour le développement Java est complet et moderne et moins contraignant que QtCreator. Par ailleurs, le fait que Java tourne sur une machine virtuelle permet une portabilité du programme sous différentes plateformes jugé inutile dans le cadre de ce projet.

Notre choix se portera sur le C#. Celui-ci dispose d'un environnement de développement puissant et complet (Visual Studio). Le framework utilisé avec le C# (.NET) est un framework complet et haut niveau. Il permet par exemple une abstraction sur des points qu'il aurait été plus long de développer en C++ comme par exemple la mise en réseau du programme. La création de l'interface graphique est aussi plus simple qu'avec ses deux concurrents, la bibliothèque graphique (Windows Presentation Foundation) fournie avec le framework simplifie par exemple la réalisation de fenêtre. Et bien que jugé peu pertinente dans le cadre de ce projet, pour ce qui est de la performance le C# se situe entre le C++ et le Java.

Le choix du C# par rapport au Java s'est donc fait sur la flexibilité du Framework graphique fourni par C#, mais c'est également l'opportunité d'apprendre plus en détail ce langage, très utilisé dans le monde professionnel.

En conclusion, le C# nous permet de simplifier un bon nombre de points qu'il est peu intéressant de développer dans le cadre de ce projet. Ce gain de temps nous permettra donc de travailler plus des points intéressants du projet, la qualité et la robustesse, et peu être d'arriver plus vite à la partie final du projet : la réalisation de l'intelligence artificielle.

5 Annexe

5.1 Webographie

- <http://www.ffothello.org/informatique/algorithmes>
- <http://wannabe.guru.org/scott/hobbies/chess>
- <http://www.chessopolis.com/computer-chess/#info>
- <http://www.tckerrigan.com/Chess/TSCP>
- <https://chessprogramming.wikispaces.com>
- <https://fr.jeffprod.com/blog/2014/comment-programmer-un-jeu-dechecs.html>
- <https://github.com/Tazeg/JePyChess>
- <http://codes-sources.commentcamarche.net/source/50090-chess-game-core-librairie-jeu-d-echec-en-c>
- http://imagecomputing.net/damien.rohmer/data/previous_website/documents/teaching/13_0fall_cpe/3eti_software_development_c/documents_generaux/02_presentation_projet.pdf
- <http://khayyam.developpez.com/articles/algo/genetic>
- http://www.gamedev.net/page/resources/_/technical/artificial-intelligence/chess-programming-part-i-gett
- <http://www.frayn.net/beowulf/theory.html>
- <http://www.chessbin.com/>
- <http://people.csail.mit.edu/plaat/mtdf.html>
- <http://www.top-5000.nl/authors/rebel/chess840.htm>
- https://en.wikipedia.org/wiki/Computer_chess