



SORBONNE UNIVERSITÉ / CFA DES SCIENCES

Projet tuteuré Quoridor

tuteur :

Devamadushan THEVENIN
Tamer BOUKHROUM
Yoan LE NEVEZ
Rémi RICHETIN

Chargé de TD :

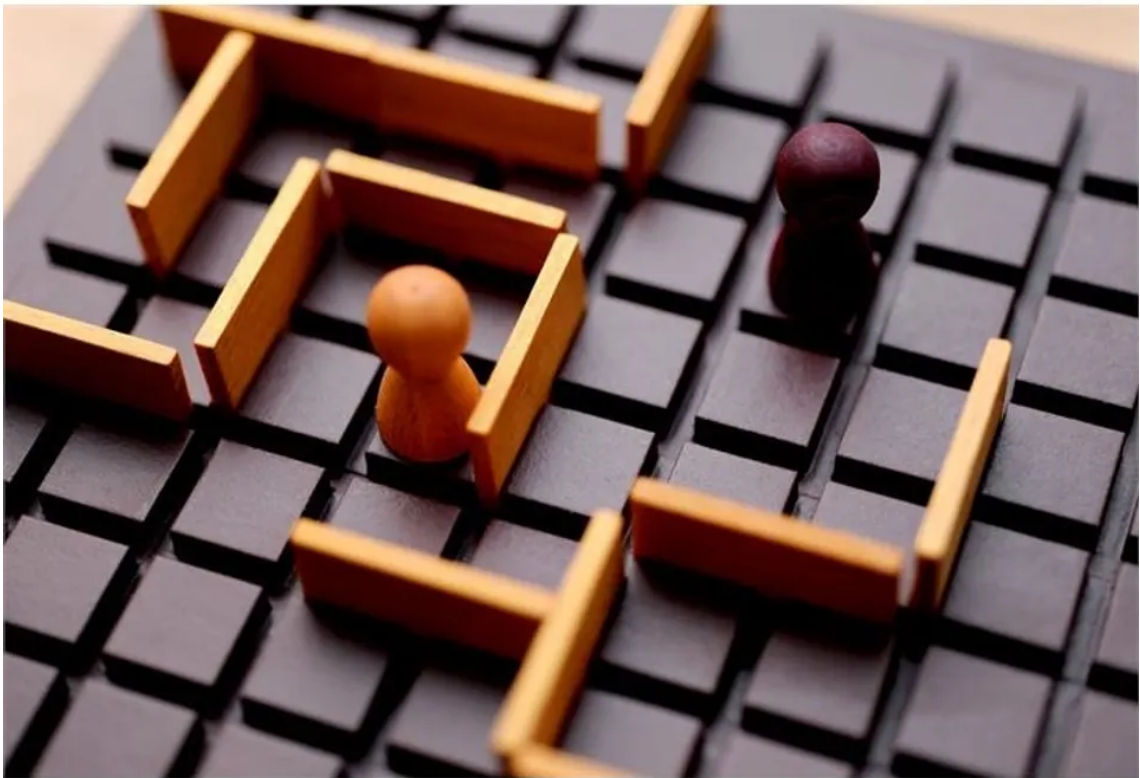
Sébastien TIXEUIL

Résumé

Le projet s'inscrit dans le cadre de l'UE projet tuteuré LU3IN117 visant à développer un logiciel ludique et stratégique inspiré du jeu de plateau **Quoridor**. Ce projet offre l'opportunité de combiner plusieurs compétences techniques, notamment la conception d'interfaces graphiques modernes, et l'implémentation d'une intelligence artificielle avancée.

Chapitre 1

Présentation du projet



1.1 But du Jeu

Le but fondamental du jeu **Quoridor** est de traverser le plateau en déplaçant son pion depuis sa position de départ jusqu'à atteindre la ligne opposée. Chaque joueur commence la partie positionné sur une ligne spécifique (par exemple, l'un en haut et l'autre en bas dans une partie à deux joueurs) et doit trouver le chemin le plus rapide et stratégique pour parvenir à l'objectif.

Chaque tour, le joueur peut choisir de déplacer son pion d'une case adjacente (horizontalement ou verticalement) pour progresser vers la ligne d'arrivée. Dans certaines situations, lorsqu'un adversaire bloque directement le chemin, le pion peut effectuer un saut pour contourner l'obstacle.

En plus de déplacer son pion, le joueur dispose d'un nombre limité de murs qu'il peut placer sur le plateau. Ces murs servent à créer des barrières qui ralentissent la progression de l'adversaire. Le challenge consiste à bien doser entre l'avancement personnel et le blocage de l'itinéraire adverse, sans compromettre son propre parcours.

La limitation du nombre de murs impose une gestion stratégique de ces ressources. Placer un mur au mauvais moment ou dans une mauvaise position peut se retourner contre le joueur, car il pourrait se retrouver lui-même bloqué.

À chaque coup, le joueur doit analyser le plateau, anticiper les mouvements adverses et envisager plusieurs itinéraires possibles pour atteindre la ligne opposée. La capacité à prévoir plusieurs coups à l'avance est cruciale pour se dégager des impasses.

L'état du plateau évolue constamment en raison du placement des murs. Ainsi, même une stratégie soigneusement planifiée doit pouvoir être adaptée en fonction des actions de l'adversaire. L'anticipation et la réactivité sont des atouts majeurs pour réussir.

Une règle essentielle du jeu est que, même avec le placement de murs, il doit toujours rester au moins un chemin viable pour chaque joueur. Cette contrainte garantit que le jeu reste dynamique et accessible, tout en forçant les joueurs à optimiser l'utilisation de leurs murs.

Synthèse

En résumé, le but du jeu **Quoridor** est de parvenir le premier à atteindre la ligne opposée du plateau, tout en utilisant intelligemment le déplacement du pion et le placement des murs pour contrer son adversaire. Chaque partie est une véritable épreuve de réflexion et de tactique.

1.2 Options retenues

fin de concevoir le jeu du Quoridor nous avons choisis deux options : .

Une interface graphique moderne

L'interface utilisateur est développée en **JavaFX**, une technologie adaptée aux applications graphiques interactives en Java. Ce choix est motivé par plusieurs facteurs :

- **Expérience et maîtrise de l'équipe** : Tous les membres du groupe ont déjà utilisé Java dont JavaFX, ce qui réduit la courbe d'apprentissage et optimise le temps de développement.
- **Ergonomie et accessibilité** : L'objectif est d'offrir une interface fluide, intuitive et agréable à utiliser, où les joueurs peuvent interagir aisément avec le plateau et les options de jeu.
- **Flexibilité et modularité** : JavaFX permet une séparation claire entre la logique et l'affichage grâce à l'utilisation de fichiers FXML, facilitant ainsi la maintenance et l'amélioration future de l'interface.

Jouer contre un ordinateur intelligent

L'un des axes majeurs du projet est l'intégration d'une **intelligence artificielle** pour offrir un mode de jeu contre l'ordinateur. L'IA sera développée en **Python**.

- **Richesse des bibliothèques IA** : Python est reconnu pour sa vaste gamme de bibliothèques spécialisées en intelligence artificielle et en optimisation de décisions, facilitant l'implémentation d'un adversaire performant.
- **Flexibilité et rapidité de développement** : Grâce à sa syntaxe claire et concise, Python permet de tester rapidement différentes approches stratégiques sans nécessiter une lourde implémentation.
- **Expérience et expertise de l'équipe** : Plusieurs membres du groupe maîtrisent déjà Python et ses outils d'IA, ce qui permet une intégration efficace de l'intelligence artificielle sans difficulté technique majeure.
- **Adaptabilité et évolutivité** : L'utilisation de Python pour l'IA permet de futurs développements basés sur l'apprentissage automatique, offrant ainsi la possibilité d'adapter l'adversaire virtuel au niveau des joueurs en fonction de leurs performances.

1.2.1 Architecture et Technologies adoptées

Nous avons conçu une architecture modulaire permettant une séparation claire des différentes couches du projet :

Backend : moteur de jeu en Java Le cœur du jeu repose sur un backend développé en **Java**, qui gère :

- La logique du jeu et l'application des règles définies
- La gestion des mouvements et des positions des entités (joueurs et murs)
- La validation et la vérification des placements de murs

Ce backend assure un fonctionnement fiable du jeu en vérifiant en temps réel la validité des actions effectuées par le joueur ou l'IA.

Frontend : JavaFX pour l'affichage L'affichage du jeu est géré par JavaFX permettant d'implémenter l'interface et d'interagir avec la logique du jeu en backend.

Gestion des modules et dépendances Le projet sera organisé en modules et utilisera **Maven** pour la gestion des dépendances (définies dans `pom.xml`).

1.2.2 Perspectives d'amélioration

Une amélioration envisageable serait le déploiement via **Docker** pour simplifier la distribution et la gestion des versions de l'application.

En **résumé**, nous avons opté pour une combinaison de **Java** pour le backend et la logique de jeu car tous les membres ont déjà utilisé ce langage, **Python** l'intelligence artificielle, et **JavaFX** pour une interface moderne et réactive.

1.3 Le groupe

Notre équipe se compose de plusieurs membres aux compétences complémentaires. Bien que les noms et rôles définitifs soient à confirmer, nous envisageons la répartition suivante :

Développement de l'intelligence artificielle(Python)

Devamadushan THEV R NCH N

Tàmer BOU K RROUM

Développement de l'interface graphique (JavaFX)

Yoan LE NEVEZ

Rémi R J R TN M

1.3.1 Mini-CV

Voici nos minis CV avec les projets et expériences pertinentes vis à vis du projet ce trouve ci-dessous.

Note : Les rôles pourront être ajustés en fonction de l'évolution du projet.

Rémi Rajaratnam

Apprenti développeur
informatique

✉ rajaratnam.remi@gmail.com

☎ 06 64 41 92 56

🌐 <https://github.com/Imrejar>

Compétences

Langages

Java, Python, C++, Javascript,
HTML, CSS, PL/SQL, OCAML,
SHELL

Outils de projet

Git, Jira, Vscodé, IntelliJ

Management de Projet

SCRUM, méthode Agile

Formations

Licence 3 informatique DANT

Sorbonne Université/CFA des
Sciences

Paris

- Alternance
- Introduction aux Data Sciences
- Nouvelles technologies du Web
- Programmation Java Avancée

Licence 1 et 2 informatique

Sorbonne Université

Paris

Bac Scientifique SI ISN

Moissy-Cramayel

Expérience Professionnelle

Développeur Back-end - Alternance

Covéa - GMF

Sep 2024 – Sep 2025 | Levallois Perret, France

- **Analyse** technique des outils de déploiement et gestion de variables pour les marques MMA, GMF, et MAAF.
- **Proposition** d'uniformisation des pratiques via CoVariable pour gérer les variables d'environnement de manière cohérente.
- **Amélioration** de l'outil interne Covariable en Java pour optimiser la gestion de Variables

Projets

Robot

Python

- Conception et développement d'un système robotique.
- Modélisation et simulation du robot.
- Création de jeux de simulation et défis d'évitement d'obstacles.
- Implémentation des fonctionnalités sur le robot physique.

Sites Web

HTML/CSS/JS

- Développement de sites web

Reconnaissance d'images

Python

- Développement d'un système de reconnaissance d'articles vestimentaires avec TensorFlow.
- Utilisation de modèles de deep learning pour une identification précise

Football et chaine de Markov

Python

- Collecte, nettoyage et modélisation des données de matchs de football.
- Utilisation de chaînes de Markov pour prédire les scores futurs.
- Validation du modèle par validation croisée.

Logiciel gestion de version

C++

- Conception sécurisée d'un système de gestion de versions.
- Cryptage des données pour la confidentialité et l'intégrité.
- Implémentation de fonctionnalités pour une gestion efficace.

Analyse de données

Python

- Analyse approfondie des données avec Pandas et Matplotlib.
- Création de visualisations claires pour des insights rapides.