Projet long - Jeu de société Cluedo
Rapport de l'itération 2 - Groupe IJ01

Agathe Perrin, Antonin Litschgy, Mickaël Song, Maëlis Marchand,
Thierry Xu, Tom Bonetto, Antoine Dalle-Fratte

Département Sciences du Numérique - Première année
2021 - 2022

Table des matières

Introduction

L’objectif de notre projet est de réaliser un jeu de société type Cluedo qui se joue sur ordinateur. Un utilisateur aura la possibilité de jouer contre des ordinateurs présentant différentes stratégies. Les principales fonctionnalités du jeu original seront retrouvées, avec notamment : le déplacement d’un pion sur le plateau de jeu, la formulation d’hypothèses et d’accusations, l’accès à une grille pour noter ses indices au fil de la partie, etc.

Ce rapport présente ce que nous avons réalisé lors de la deuxième itération du projet qui s’est déroulée du 9 au 23 avril.

Organisation

Pour cette deuxième itération, nous avons adopté une nouvelle organisation. Chaque membre de l'équipe s'est donné des objectifs de mi-itération, à réaliser lors de la première semaine de l'itération, puis des objectifs pour la seconde semaine. Cela nous a permis de mieux nous répartir le travail, aussi bien dans l'équipe que dans le temps.

Travail réalisé au niveau du code

Voici les principaux objectifs que nous avons réalisé au niveau du code :

- Travail sur l'implémentation des stratégies ;
- Menu du choix du personnage qui s'affiche en début de partie ;

[objectifs]

Nous avons donc réparti ces différents objectifs au sein de l'équipe.

Détails de la réalisation

1. Travail sur l'implémentation des stratégies

La programmation des différentes stratégies (humaine, naïve et experte) a été poursuivie, notamment les fonctions *montrerCarte*. Cette méthode choisit la carte montrée par un joueur lorsqu'un autre joueur formule une hypothèse. Par exemple, si un joueur naïf possède 2 cartes de l'hypothèse, il en montrera une au hasard, tandis qu'un joueur expert montrera en priorité une carte que le joueur qui formule l'hypothèse aurait déjà vu pour le désavantager.
Nous avons également ajouté une classe **StrategiePassive**, qui sera la "stratégie" adoptée par un joueur qui ne joue pas. En effet, si le nombre de joueurs est inférieur à 6, un personnage qui n'est pas pris par un joueur doit quand même pouvoir se déplacer dans une salle quand un joueur formule une hypothèse qui l'implique.
Il reste encore du travail dans la programmation des stratégies. La conception objet est peut-être à revoir : pour éviter les redondances, il serait peut-être plus judicieux de faire une classe Stratégie, avec d'autres classes qui en héritent, plutôt qu'une interface Stratégie.

2. Menu du choix du personnage

Le menu du choix du personnage - qui s'affiche en début de partie quand le joueur humain doit choisir son personnage - a été commencé. Pour l'instant, les 6 personnages du Cluedo (Rose, Pervenche, Orchidée, Olive, Moutarde, Violet) s'affichent et chacun est associé à un bouton. Quand le joueur cliquera sur le bouton du personnage qu'il souhaite, alors ce personnage lui sera attribué. Pour l'instant, les actions des boutons ne sont pas terminées :

un bouton cliqué affiche dans le terminal le nom du personnage en question. Cela est évidemment à changer par la suite.

3. Création de la classe Arbitre et de sa classe de test.

La classe arbitre permet aux joueurs de jouer tour à tour. Elle possède un constructeur qui définit une liste des joueurs qui vont jouer ainsi que des méthodes permettant d'avoir le joueur courant, de passer au joueur suivant et d'arbitrer la partie.

4. Création de la classe Carnet et de sa classe de test.

La classe carnet permet d'écrire des symboles dans le carnet, elle sera utilisée par les joueurs pour noter des informations sur la partie. Plusieurs symboles sont possibles. C'est en fait un tableau à deux dimensions de symboles. Initialement tous les symboles sont vide. Le joueur peut cocher une case avec la méthode cocherCase prenant en paramètre les coordonnées de la case et le symbole qu'il veut y mettre. Il peut aussi passer au symbole suivant (par exemple en cliquant on passe du premier symbole au deuxième, au troisième,... jusqu'à revenir au premier. On peut également vider une case, une ligne, une colonne ou la grille entière. Tous les tests réalisés sur cette classe sont fonctionnels.

5. Travail sur la grille de jeu

La grille de jeu a été avancée par rapport à la dernière itération. Il y a eu l'ajout de plusieurs icônes de notation d'indices. Pour cela, nous avons dessiné les différentes cases, pixel par pixel, grâce au logiciel GIMP. Nous avons ajouté au joueur la possibilité d'avoir des cases blanches, des checks, des ronds, des points d'interrogations et autres. Pour implémenter dans la grille d'indice le changement d'icône nous avons créé une méthode checkBoxListener qui prend en paramètre un Action event afin qu'à chaque clique sur la case, le motif de cette dernière change. Il est changé dans l'ordre suivant :

case blanche	→	motif check
motif check	→	motif point d'interrogation
motif point d'interrogation	→	motif rond
motif rond	→	motif autre
motif autre	→	case blanche

6. Affichage du plateau

L'interface graphique du plateau est maintenant implémentée. Pour l'instant, elle est encore basique et permet uniquement de différencier une salle d'un couloir. Celle-ci est basée sur une matrice composée de caractères (M pour mur, S pour salle, C pour couloir, P pour porte) qui représente le type des cases. De plus, l'interface gère l'affichage d'un pion sur une case selon la présence du joueur sur celle-ci.

7. Plateau de Jeu et Joueur

On peut maintenant ajouter des joueurs sur des cases, vérifier si une case est déjà occupée ou non. De plus, si un joueur est ajouté sur une case de type Salle alors il sera placé automatiquement dans un emplacement libre de la zone dédiée de la salle. Des tests ont été réalisés pour vérifier certaines fonctions et certains aspects de l'implémentation du plateau. Du côté Joueur, il possède maintenant un attribut Position qui permet de lui donner des coordonnées, les fonctions pour récupérer cette position ou la modifier ont été également implémentées.

8. Fenêtre d'actions

La fenêtre d'actions est composée des dés et des différents boutons permettant d'émettre une hypothèse, accusation ou pour terminer le tour du joueur. Pour l'instant, celle-ci n'est que graphique et aucune action n'est associée aux boutons.