

Claim

Dossier de Validation IHM



Table des matières

1. Introduction.....	3
2. Affichage du plateau de jeu.....	
a. Affichage du plateau.....	
i. Affichage des cartes et du plateau de jeu	
ii. Affichage du type d'une carte en fonction de l'état du jeu	
iii. Mise des cartes à la bonne échelle	
b. Gestion des interactions avec le plateau.....	
i. Gestion des cliques sur le plateau	
ii. Affichage des cartes jouables	
iii. Affichage des cartes joué et pioché	
c. Fonctionnalités supplémentaires.....	
i. Affichage de la pioche, défausse et pile de score en option	
ii. Animation de distribution des cartes et temporisation	
iii. Affichage de la main du joueur adverse	
3. Interface Graphique.....	
a. L'information disponible au joueur.....	
i. Les scores	
ii. Le joueur courant	
iii. L'IA qui réfléchit	
b. Les boutons de l'interface.....	
i. Menu	
ii. Nouvelle Partie	
iii. Règles de jeu	
c. Le menu.....	
i. Partie Customisable	
ii. Bouton Annuler et Refaire	
iii. Bouton Sauver et Charger une partie	
d. Fin de partie.....	
4. Inspirations et choix pour l'évolution de l'affichage.....	
a. Références prises	
b. Tests utilisateurs	
c. Evolutions	
5. Conclusion.....	

Introduction

Nous avons, à la suite de plusieurs retours et idées au sein groupe, construit l'interface des interactions humains-machines pour permettre aux utilisateurs de notre programme de pouvoir facilement prendre en main et effectuer une ou plusieurs partie de ce jeu de plateau que nous avons adapté sur ordinateur à l'aide du langage de programmation java et de la bibliothèque graphique Swing.

Il est important de noter que l'interface utilisateur de ce jeu n'est designé uniquement pour le combat contre une IA, la partie joueur contre joueur qui marche sur le point technique n'est pas spécialement jouable.

Affichage du plateau de jeu

Affichage du plateau

Affichage des cartes et du plateau de jeu

Les cartes du Premier joueur, celui qui va commencer la partie sont toujours dessiner en bas au centre de l'écran, elles sont trié par faction et par valeur dans la main de gauche à droite. Les cartes du second joueur se trouve en face de la main du joueur en haut au centre de l'écran. Au centre du plateau se trouve deux emplacements gris qui indique la où sera posé la carte joué par le joueur lors d'un clic sur sa carte, en bas pour le premier joueur et en haut pour l'autre joueur.

Sur la plateau à gauche se trouve deux emplacements rouge pour la pile de score, l'un sur l'autre, la pile de score du premier joueur est en bas et celle du joueur 2 est en haut, Sur la droite juste à droite à la même hauteur que les mains se situe en bleu les piles de carte de la seconde main qui vont constituer la main de la phase 2, en bas celle de premier joueur et en bas celle de second. Durant la phase 1 une carte au centre légèrement à gauche sera affiché prise la de pioche qui indique la carte à gagné pour la main de la phase 2. A gauche se trouve la pioche en vert et l'emplacement de la défausse en violet. Pour la phase 2, la pioche et la défausse ne sont plus nécessaires et donc plus affichées.

Affichage du type d'une carte en fonction de l'état du jeu

Pour afficher les cartes de jeu dans le niveau Graphique nous avons décider de récupérer depuis le moteur le type Card qui contient un int et un string qui correspondent à la valeur et à au nom de la faction de la carte et de transformer en tableau de couple d'entier où la première valeur est la valeur et la seconde et une autre valeur qui correspond à la faction :

- 1 -> Goblin
- 2 -> Dwarf
- 3 -> Knight
- 4 -> Doppelganger
- 5 -> Undead

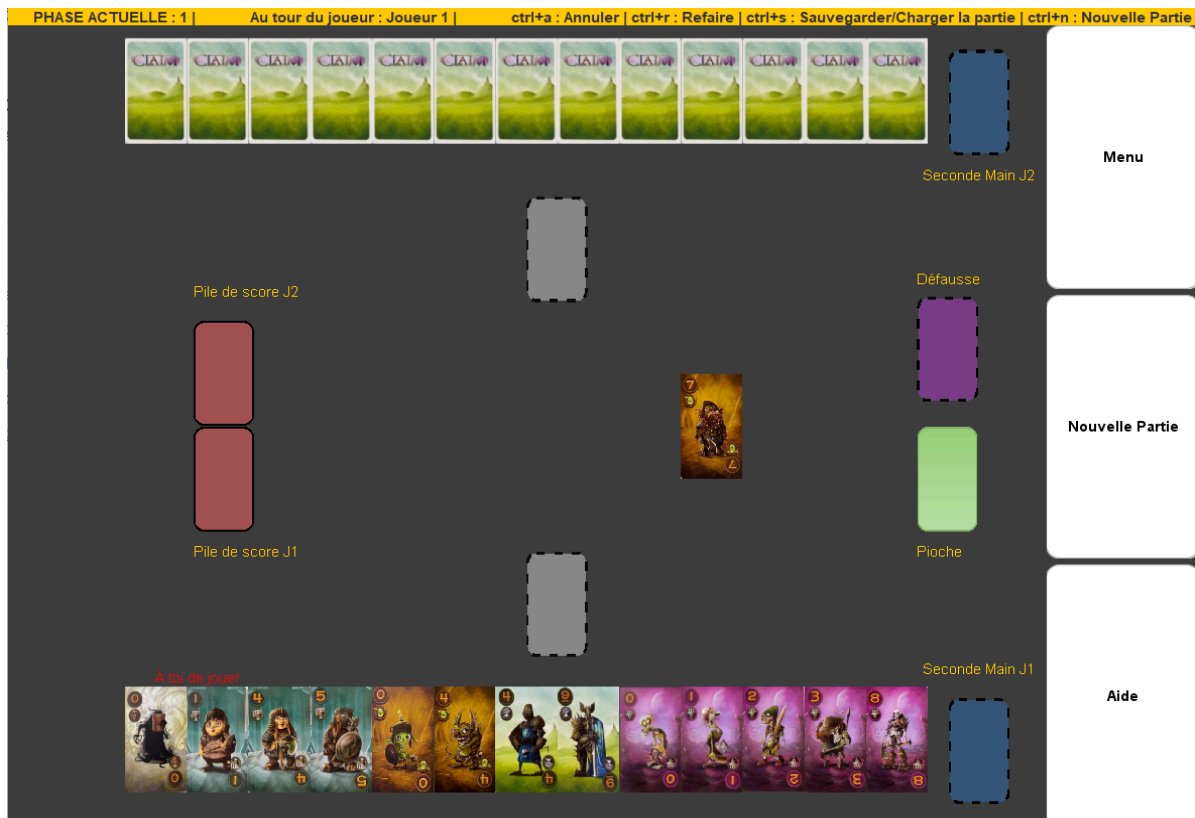
Exemple le couple d'entier (6, 4) correspond à Doppelganger de valeur de 6 :



Mise des cartes à la bonne échelle

Pour la gestion de la carte selon la taille du plateau, on met à jour la taille des cartes en fonction de la largeur et la hauteur actuelle de la fenêtre, en utilisant des multiplication et division pour trouver la taille et l'emplacement des cartes selon la taille de fenêtre.

On trouve la largeur de la fenêtre avec `getWidth()` et sa hauteur avec `getHeight()`. Pour la redimension de la fenêtre si on redimensionne, certaines dimension rendent la formes des cartes rectangulaire, peu visible, pour éviter cela si nous avons du temps nous aurions restreint le redimensionnement en liant hauteur et largeur de façon à ce que tout changement de la hauteur change aussi la largeur pour garder le même ratio entre ces deux valeurs, après avoir choisie une bonne dimension de base.



Gestion des interactions avec le plateau Gestion des cliques sur le plateau

Affichage des cartes jouables

Pour les cartes lorsqu'on joue en second certains règles nous forcent à jouer certaines carte de notre main, pour faciliter et aider l'utilisateur, les cartes qui ne sont pas jouables à un moment donnée sont grisées pour éviter que l'utilisateur reste bloqué en essayant de cliquer sur une carte qui ne peut être joué.

Affichage des cartes gagnantes et perdantes

Pour aider l'utilisateur à jouer en second lorsqu'il doit suivre le pli, sans qu'il est à ce souvenir de toutes les règles pour savoir si la carte qu'il va jouer va le faire gagner ou perdre le pli, on dessiner une bordure verte ou rouge autour des cartes jouables, une bordure verte pour une carte qui va gagné et une bordure rouge pour les cartes qui vont perdre.



Réception du clic par un utilisateur des cartes jouées

Les cartes qui veulent être jouées par l'utilisateur sont récupérées lors d'un clic sur cette carte. On récupère la position avec la classe `AdaptateurSouris` qui va appeler `MousePressed(MouseEvent e)` et `e.getX()` et `e.getY()`. Puis on convertit la position (x, y) du clic en indice de position de la carte dans la main pour l'envoyer au contrôleur médiateur qui va effectuer le tour de jeu avec cette carte et mettre à jour le moteur en utilisant la fonction `clicSouris(int indiceCarte)` qui va appeler `joueTour(int indiceCarte)`.

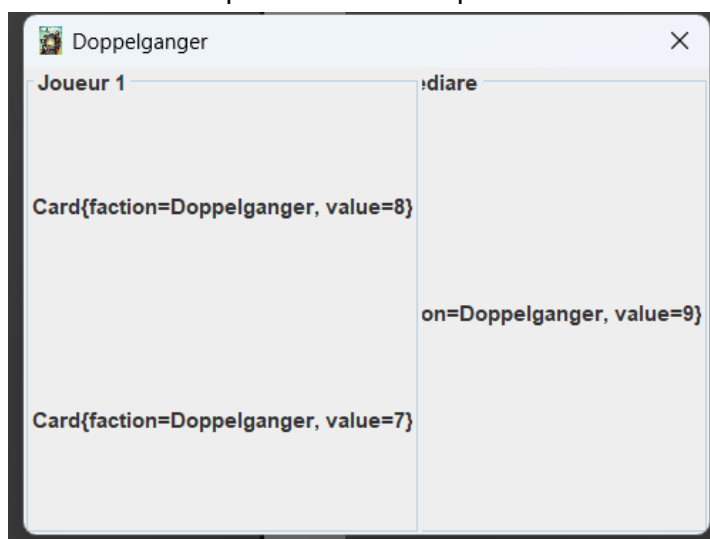
Si l'utilisateur a cliqué sur une carte non jouable ou en dehors de sa main, aucune action n'est effectuée. Les cartes jouées par chaque joueur sont stockées à chaque tour de jeu dans des variables `Card carteJoueur1`, `carteJoueur2` dans le moteur et sont ensuite récupérées pour les afficher au centre du plateau.

Si nous avions plus de temps nous aurions aimé implémenter un drag and drop sur les cartes que l'on veut jouer en déplaçant l'image de la carte sur la table sur la position de la souris quand le clic est maintenu et si le clic est relâché quand la souris a largement dépassé la hauteur des autres cartes de la main.

Fonction supplémentaires

Affichage de la pioche, défausse et pile de score en option

Dans les options de jeu nous avons un toggle qui détermine si nous affichons des informations supplémentaires pour aider le joueur. Un affichage qui affiche ou non le contenu de la pile de score des deux joueurs par faction en cliquant sur une icône de faction dans le rectangle de la pile de score à droite qui va ouvrir une fenêtre avec la liste des cartes de cette faction. Par manque de temps nous n'avons pas implémenté cette fonctionnalité pour la défausse et la pile de score en option mais nous l'aurions affichée de manière similaire.



Animation de distribution des cartes et temporisation

Pour que les joueurs comprennent bien les actions qui viennent d'être effectuées. Au moment de la bataille à chaque pli une fois que les deux cartes ont été jouées, on effectue une pause suivit d'une animation qui va déplacer les cartes de leurs position actuelle vers leurs destinations.

Exemple :

Dans la phase 1 avec :

- Un undead 7 en premier et
- Un knight 2 en second,
- Un un dwarf 8 à gagner au centre.
- Le undead 7 va dans la pile de score du premier joueur,
- Le knight 2 va dans la défausse,
- Le dwarf 8 va dans la pile de la seconde main du premier joueur,
- Une carte de la pioche est ajouté à la seconde main du second joueur
- Enfin une carte nouvelle carte est prise de la pioche est mise au centre en carte à gagner pour le prochain pli si la phase 1 n'est pas terminée.



Les animations peuvent être longue surtout une fois que les joueurs connaissent à l'avance le résultat du pli une fois les règles bien comprises. Si nous avions eu le temps nous aurions ajouté une option dans le menu pour choisir entre 3 type d'animations : "Normales", "Rapides", et "Pas d'animation". Qui modifierai les valeurs du nombre d'itération et de pause des animations.

La durée totale de l'animation est contrôlée par une variable `dureePause` dans le contrôleur, qui peut varier entre la phase 1 et la phase 2 (car la phase 2 nécessite beaucoup moins d'animations). Pour chaque étape de l'animation, il existe une variable nommée `latence`, car chaque animation démarre simultanément, et cette latence permet de contrôler l'ordre dans lequel chaque étape de l'animation se déroule. La temporisation entre les tours est contrôlée par un timer, et l'utilisation de fonctions de rappel assure l'ordre correct d'exécution du code et du timer.

La transition entre les deux phases est réalisée en affichant un message au milieu du jeu à la fin de la phase un pour informer le joueur qu'il va passer à la phase deux.



Affichage de la main du joueur adverse

Dans les options nous avons ajouté une fonctionnalité pour choisir d'affiché tout le temps la main et les cartes du second joueurs pour obtenir des informations supplémentaire pour

jouer. L'utilisateur peut choisir d'activer ou non cette option pour s'aider dans la partie. Cette option est aussi utile pour montrer si les coups de l'IA sont stratégiques ou non.



Interface Graphique

L'information disponible au joueur

La phase actuelle et les raccourcis clavier

Dans l'interface principale (InterfaceGraphique) au dessus du plateau de jeu se trouve une barre de fond jaune qui à gauche indique la phase actuelle de jeu, suivi du joueur courant puis des raccourcis clavier avec l'action qui leur est associée.

Au tour du joueur 1

Quand l'utilisateur doit jouer, par exemple au début de la partie ou après que l'IA est joué. Un texte en rouge à dessus à gauche de sa main s'affiche pour lui indiquer qu'il peut jouer.

Les boutons de jeu

A droite du plateau de jeu se trouve 3 boutons qui vont s'activer lors d'un clic :

- "Ouvrir Menu" : qui va ouvrir une nouvelle fenêtre avec un Menu et plusieurs boutons et options.
- "Nouvelle Partie" : qui va lancer rapidement une nouvelle partie avec les mêmes settings que la partie actuelle.
- "Aide" : qui va ouvrir un pop avec les règles du jeu.

Le menu

Bouton Annuler et Refaire

Les boutons annuler et refaire qui sur un clic vont annuler le dernier coup joué jusqu'au début du tour du joueur 1 ou refaire la dernière action annuler aussi jusqu'au début du tour du joueur 1.

Partie Customisable

Le bouton partie customisable va ouvrir une fenêtre InterfacePartieCustom et fermer la fenêtre du jeu. Dans cette nouvelle fenêtre le joueur peut choisir de créer une nouvelle partie

en choisissant le nom des joueurs et s'il veut affronter un autre Humain où une IA et peut choisir son niveau de difficulté.

Si il choisit d'affronter une IA le nom de l'IA sera fixe.

Bouton Sauver et Charger une partie

Ce bouton ouvrent une petit fenêtre avec un champ de texte qui attend le nom du fichier de sauvegarde d'une partie suivit de 2 boutons :

- Sauvegarder : qui va sauvegarder un fichier de la partie actuelle avec le nom donné dans le champ de texte
- Charger : qui va charger une partie à partir du fichier qui la nom du nom donné dans le champ de texte

Check Box Toggles

Dans le menu se trouve en bas 2 check box toggle :

- La première : "Afficher la main de l'adversaire" qui quand elle est selectionné va remplacer les cartes face caché de la main adverse par des cartes visibles.
- La seconde : "Afficher les cartes et infos de la pile de score" qui va afficher un petit tableau à gauche des emplacements de piles de scores avec pour chaque faction si on gagne cette faction en vert, en rouge si on perd et en gris si aucune carte de cette faction n'est présente dans la pile. Avec un rectangle et un chiffre qui indique la plus haute valeur de cette faction dans la pile de score.

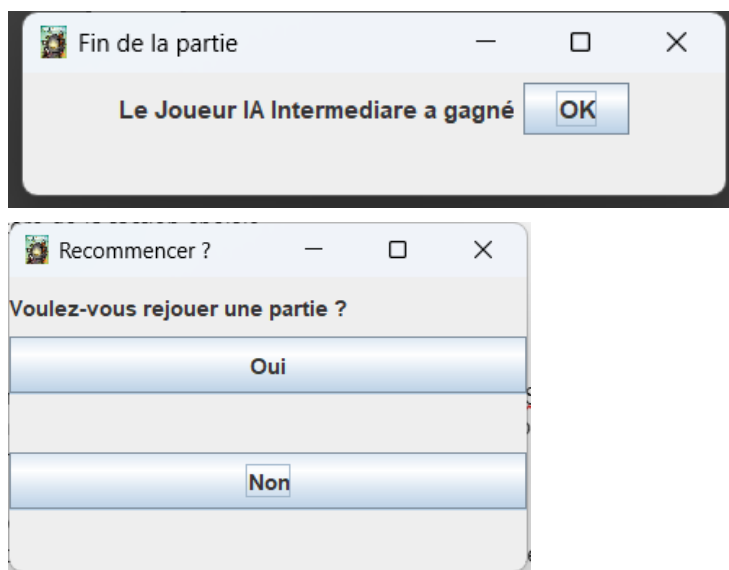
Si on clique sur les icônes de la pile de score, un pop-up s'ouvre et indique les cartes dans la pile de score de la faction choisie.

Fin de partie

Quand une partie se termine, une condition est check dans la boucle javaSwing : qui va regarder si l'animation est fini puis va rendre visible une fenêtre pour le gagner puis une fenêtre pour recommencer la partie.

Si cette condition est vraie, on attend la fin de l'animation, alors une fenêtre s'affichera avec le nom du vainqueur du jeu et un bouton 'OK'. Si l'utilisateur clique sur 'OK', une fenêtre s'ouvrira avec un composant 'ComposantRecommencerPartie' qui propose deux boutons :

- 'Oui', pour relancer une nouvelle partie rapide avec les mêmes paramètres que la partie actuelle.
- 'Non', pour masquer cette fenêtre avec `setVisible(false)` et permettre à l'utilisateur de voir le plateau final du jeu.



Inspiration et choix pour l'évolution de l'affichage

Références prises

Nous nous sommes inspirés de l'affichage de jeu de bataille à pli comme la bataille, la belote, le rami, le tarot, le bridge, poker ou même l'interface de jeu de carte plus récent qui se joue sur table et qui ont été adapté sur ordinateur comme Yugioh, Magic the gathering ou pokemon.

Certains de ces jeux sont disponible en jeu informatique avec une interface utilisateur simple avec des animations et positions de cartes. En regardant et en jouant à ces jeux nous avons pris des décisions quant à la façon dont on agence notre plateau de jeu et le cartes et la façon dont les animations devraient se jouer.

Tests utilisateurs

Nous avons pris plusieurs tests utilisateurs qui nous aidés pour prendre des choix pour modifier notre jeu. Voici une synthèse de nos retour trié par âge et par genre :

Les gens plus jeune comprenez assez vite le fonctionnement du jeu via le feedforward et feedback parfois même en quelque tours, au contraire les personnes plus âgée ne pouvait pas du tout comprendre le jeu sans les règles. Les différents emplacements des cartes était compris par certains joueurs sans texte juste grâce aux animations mais les joueurs ne connaissant pas les jeux de cartes n'arrivait pas à deviner la défausse et la pioche.

Nous avons des retours sur la lisibilité du plateau, la compréhension des différents éléments du plateau et sur l'application des règles. Nous avons essayé de répondre à ces retours en modifiant et en faisant évoluer notre interface pour aider les utilisateurs. Nous avons par exemples ajouté beaucoup de texte en rouge et jaune sur le niveau graphique

pour aider les gens avec un texte plus accessible, un bouton d'aide qui donne une direction pour jouer beaucoup plus court que les règles.

Evolutions

A partir des retours utilisateurs nous avons fait évoluer notre jeu avec plusieurs versions de l'affichage de l'interface utilisateur :

- Nous avons eu plusieurs versions de l'interface

Sans textes

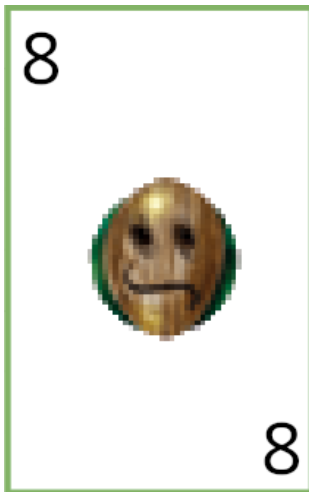


Avec des textes



L'ajout des textes pour indiquer les emplacements était très demandé par les retours, et si nous avions eu le temps nous aurions mis une options pour désactiver les textes sur le niveauGraphiques du plateau, qui serai activé par défaut.

Nous avons commencé avec des designs de cartes très basique:
Des cartes très basique avec un fond blanc, un contour, un symbole et deux chiffres :



Ce design à ensuite été changé suite au retour par le design des cartes du jeu officiels.



Si nous avions eu le temps nous aurions mis un tutoriel d'aide pour les règles, plus digestif que le texte des règles.

Conclusion

En Conclusion nous avons fait notre plateau en s'inspirant beaucoup de jeu pré-existant pour créer notre interface. Beaucoup de fonctionnalité que nous aurions aimé ajouté mais par manque de temps et d'effectif nous nous sommes concentré sur ce qui nous sembler plus important d'abord en laissant pour plus tard ce qui nous a semblé le moins important. Les retours nous ont aussi aidé à nous orienté vers ce qui était le plus important d'abord..