



Dossier de Conception P32

Jeu des paires

Par Diego Rodriguez



Table des matières

| | |
|---|---|
| I) Sujet..... | 3 |
| II) Fonctionnement de l'application | 3 |
| a) Comment jouer..... | 3 |
| b) Menu | 4 |
| III) Hiérarchie des composants de la vue initiale | 6 |
| IV) Contrôleur de la vue | 7 |
| a) Diagramme états-transitions..... | 7 |
| b) Traitements | 7 |
| c) Traitements externes | 8 |

I) Sujet

Ce projet propose une implémentation d'un jeu de mémoire visuelle et plus précisément un jeu de paires. L'objectif est de retrouver les paires correspondantes dans un tableau qui contient des images ou des couleurs. L'idée de ce sujet m'est venue lorsque je jouais aux cartes avec mon colocataire. Nous avons joué au jeu des paires et je me suis dit que le coder en Swing pourrait être une bonne initiative. Les utilisateurs pourront ainsi s'amuser à retrouver des paires sans forcément avoir un paquet de cartes sous la main. Les joueurs pourront ainsi améliorer leur mémoire visuelle et découvrir leur potentiel. La mémoire est une faculté qui doit toujours se travailler et c'est pourquoi je pense qu'il est important de jouer à ce style de jouer pour l'entretenir et faire travailler l'élasticité du cerveau.

II) Fonctionnement de l'application

a) Comment jouer

Une fois l'application lancée, la page qui s'affiche permet au joueur de directement commencer la partie. Une description détaillée des composants graphiques sera faite ci-dessous. Le joueur fera face à une fenêtre avec une grille de bouton qui n'affichent rien. Il devra cliquer sur les boutons afin de révéler l'image qui se cache derrière ceux-ci. Il remarquera qu'après 2 clics sur 2 boutons différents, il ne pourra plus cliquer sur d'autres boutons. En effet, le joueur devra cliquer sur le bouton « Valider » afin de valider sa sélection. Ce « blocage » forcé est utile car il force le joueur à ne pas se tromper lors de ses choix pour trouver les bonnes paires. Une fois le bouton « Valider » saisi, si les deux boutons correspondent à une paire (c'est-à-dire que l'image qu'ils contiennent est la même) alors le contenu des boutons se figera et ils ne seront plus cliquable. Il restera donc N-1 paires à trouver.

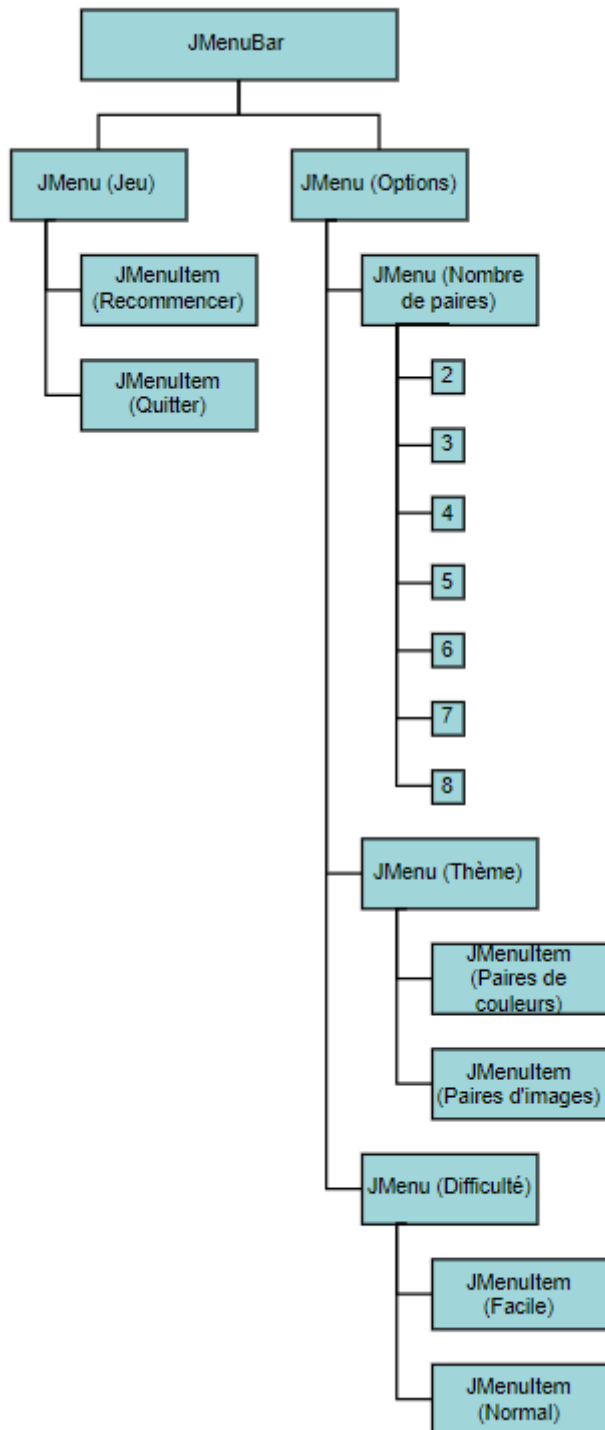
Si les deux boutons choisis ne correspondent pas à une paire lors du clic sur le bouton « Valider », les deux boutons se remettront à leur état initial, les images ne seront plus visibles et les boutons redeviendront cliquables. Bien sûr, les images cachées restent toujours à la même position entre chaque coup du joueur. Ainsi se suit une suite d'action afin que le joueur trouve toutes les paires du tableau. Si le joueur arrive jusqu'au bout, un message de victoire s'affichera en indiquant au joueur le nombre de coups qu'il lui a fallu pour trouver toutes les paires.

Durant la partie, le nombre de coups que le joueur a actuellement fait est affiché en bas à gauche. En bas à droite, le joueur peut appuyer sur le bouton « Recommencer » pour recommencer la partie. Les paires seront mélangées et tous les boutons reviendront à leur état initial. Bien sûr, le joueur peut quitter l'application en cliquant sur la croix en haut à droite, il peut agrandir la fenêtre en cliquant sur le rectangle en haut à droite et peut aussi réduire l'application en cliquant sur le tiret en haut à droite.

b) Menu

Le jeu contient aussi un menu qui permet de gérer la partie et de modifier des options du jeu afin de permettre à l'utilisateur de se sentir un peu plus libre dans le jeu et de modifier la partie en fonction de ses préférences.

Voici ci-dessous l'arbre de décomposition hiérarchique du menu.



Dans ce graphique, seul les JMenuItems déclenchent un événement. Voici une description de chaque événement.

Recommencer => Permet de relancer une nouvelle partie en réinitialisant la fenêtre actuelle. Similaire au bouton « Recommencer » en bas à droite de la fenêtre.

Quitter => Permet de quitter l'application

2,3,4,5,6,7,8 => Permet de définir le nombre de paires à trouver. Par exemple, si l'utilisateur sélectionne 2, alors il n'y aura que 2 paires, soit 4 boutons et la taille de la fenêtre s'adaptera par rapport à cela. Le processus est le même pour les autres chiffres. De base, le nombre de paires est de 8.

Paires de couleurs => Le contenu des boutons correspond à des couleurs. Il faudra donc trouver des paires de couleurs.

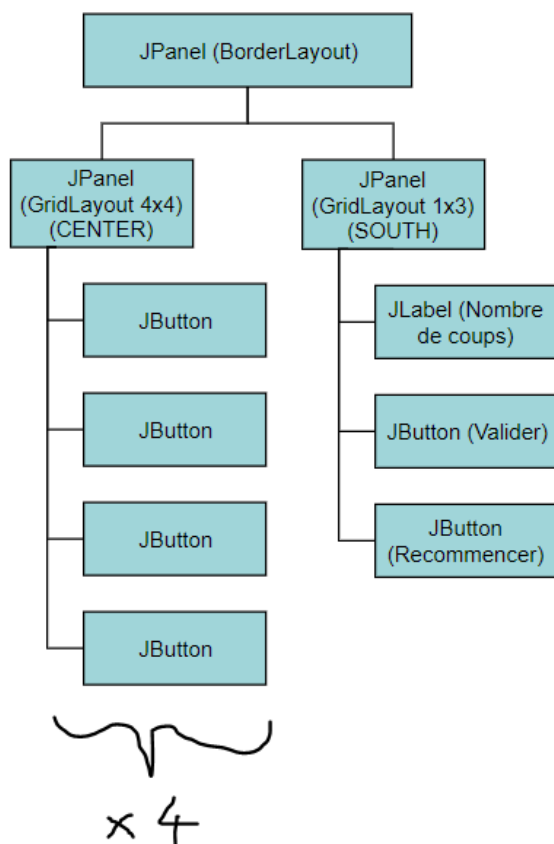
Paires d'images => Le contenu des boutons correspond à des images. Il faudra donc trouver des paires d'images. Cette option est l'option par défaut.

Facile => Le difficulté du jeu sera définie à facile. Le joueur n'aura pas à valider tous les deux coups. Ce mode permet de finir le jeu beaucoup plus vite et permet aux enseignants de tester l'application plus rapidement.

Normal => La difficulté du jeu sera définie à normale. Le jeu sera en mode normal. C'est la difficulté par défaut. Le joueur devra valider tous les deux coups.

III) Hiérarchie des composants de la vue initiale

Ci-dessous est présent l'arbre de décomposition hiérarchique des composants de la vue initiale, c'est-à-dire la vue qui apparaît lors du lancement de l'application. Les éventuelles modifications graphiques de la vue seront précisées ci-après.

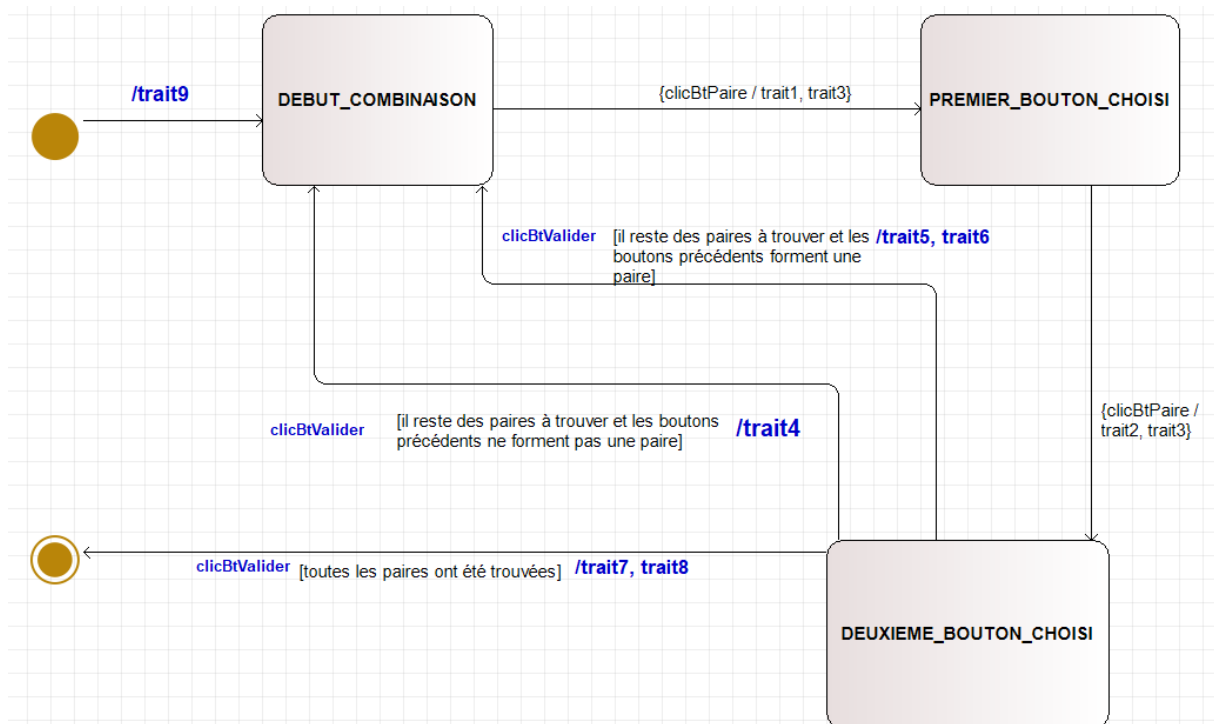


En fonction du nombre de paires que choisit l'utilisateur dans les options du menu, la taille de la fenêtre va s'adapter. Plus particulièrement, seulement le `JPanel CENTER` va être modifié. Étant donné que le nombre de boutons sera réduit, le `GridLayout` le sera aussi. Voici un petit tableau qui décrit ce `GridLayout` en fonction du nombre de paires.

| Nombre de paires | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|------------------|-----|-----|-----|-----|-----|-----|-----|
| GridLayout | 2x2 | 3x2 | 2x4 | 2x5 | 3x4 | 2x7 | 4x4 |

IV) Contrôleur de la vue

a) Diagramme états-transitions



Je n'ai pas indiqué en condition dans le diagramme que pour le bouton « Valider », il faut toujours être passé par les deux états « PREMIER_BOUTON_CHOISI » et « DEUXIEME_BOUTON_CHOISI ». Je ne l'ai pas précisé car c'est une condition inhérente au suivi logique du diagramme, mais je préfère le préciser ici pour éviter toute incompréhension.

Les ronds correspondant à l'état initial et l'état final sont inversés par rapport à ce que l'on a vu en cours. Modelio oblige à mettre le rond plein en état initial et inversement pour l'état final. De plus à cause d'un problème sur Modelio, vous remarquerez que j'ai écrit certains boutons et traitement en noir, et d'autres en bleus. Veuillez m'excuser pour cette différence mais je n'ai pas pu rester homogène sur tout le schéma car Modelio mettait automatiquement des crochets gênants. Pour les flèches des traitements 1,2 et 3, je n'ai pas eu ce problème.

b) Traitements

Trait1 : Enregistrer le premier choix de l'utilisateur, désactiver le bouton et afficher son contenu.

Trait2 : Enregistrer le deuxième choix de l'utilisateur, désactiver le bouton et afficher son contenu.

Trait3 : Enregistrer le bruitage du clic sur un bouton.

Trait4 : Réactiver les deux boutons précédents et cacher leur contenu.

Trait5 : Désactiver définitivement les deux boutons et afficher leur contenu.

Trait6 : Activer le bruitage de paire correcte.

Trait7 : Activer le bruitage de victoire.

Trait8 : Afficher la victoire du joueur en le félicitant et indiquer le nombre de coups lui a fallu pour finir le jeu.

Trait9 : Activer une musique de 3 minutes à chaque fois que la vue est créée.

c) Traitements externes

Certains traitements ne s'appliquent pas en fonction des états. Ils sont détaillés ici.

Traitement réinitialiser : réinitialiser la vue lorsque le joueur clique sur le bouton « Recommencer » en bas à droite ou dans le menu.

Traitement mode facile : enclencher le mode facile si le joueur le sélectionne dans le menu. Une fois dans ce mode de difficulté, le retour à l'état « DEBUT_COMBINAISON » peut être retrouvé en cliquant sur un troisième bouton ou en cliquant sur le bouton « Valider ». En mode normal (le mode par défaut du diagramme ci-dessus), l'état « DEBUT_COMBINAISON » est retrouvé seulement en cliquant sur le bouton « Valider ».

Traitement du type de tableau : changer le type de tableau (couleurs ou images) en fonction de ce que choisit le joueur dans le menu.