

## **RAPPORT DU PROJET JeuMemory**

**Nom** : BARRY

**Prénom** : Saliou

**Groupe** : IE2-21a

## I. Hiérarchie de classe qui a été mise en place avec les classes « Action », « Transfert » et « Bataille »

La hiérarchie de ces trois classes est en fait une forme d'héritage dont la classe mère est la classe **Action** et les classes **Transfert** et **bataille** héritent de la classe mère et l'utilité de cet héritage est qu'en fait on peut faire dériver une nouvelle classe d'une classe existante et ainsi en récupérer les attributs et méthodes, sans avoir à la réécrire totalement, on évite ainsi la redondance du code, il permet également de raccourcir le temps de mis au point du code déjà utilisé

## II. Déroulement d'une action de « Transfert »

L'ouverture de la boîte de dialogue « transfertDlg » se fait lorsqu'un joueur courant obtient tous les personnages de la famille « rare » ou « communs », cette boîte de dialogue reçoit de la fenêtre principale des informations sur l'indice du joueur courant et la liste des joueurs ; la méthode **initCombo()** est appelé dès l'ouverture de la boîte, elle permet de remplir une liste déroulante nommée ComboJoueurs (de type JComboBox) avec le pseudo de tous les joueurs.

Une fois qu'un joueur est sélectionné dans la liste déroulante, le gestionnaire de la JComboBox vérifie s'il ne s'agit pas du joueur courant et fait appel à deux méthodes **initPanneau()** et **affichePanneau()** elles servent respectivement à créer autant de boutons que de personnages possédés par le joueur sélectionné sur le Panneau gauche nommé *PanneauG* et afficher les photos des personnages sur les boutons créés.

Le clic sur un des boutons de personnage du Panneau gauche déclenche un gestionnaire d'évènement `private void boutonActionPerformed(ActionEvent evt){ ....}` qui permet d'encadrer en rouge les personnages de la même famille et affiche les informations de ces personnages sur une zone d'édition nommée « Infos ».

L'action de transfert est réalisé lors du clic sur le bouton Transfert et le déclenchement de ce gestionnaire crée une instance de la classe transfert et fait appel à sa méthode **execute()** qui permet de retirer les personnages du joueurs du joueur sélectionné et de les ajouter au joueur courant, ensuite il fait appel à deux méthodes **creerPanneau()** qui récrée les boutons sur le panneau gauche (*PanneauG*) pour le joueur sélectionné et droit (*PanneauD*) pour le joueur courant en fonction du nombre de personnage possédé par chacun et **dessinePanneau()** qui affiche les photos des personnages sur les boutons créés.

A la fin de l'action, la boîte de dialogue renvoie l'information sur le déroulement du transfert à travers l'accessoire **getDeroulement()**.

### III. Déroulement d'une action de « Bataille »

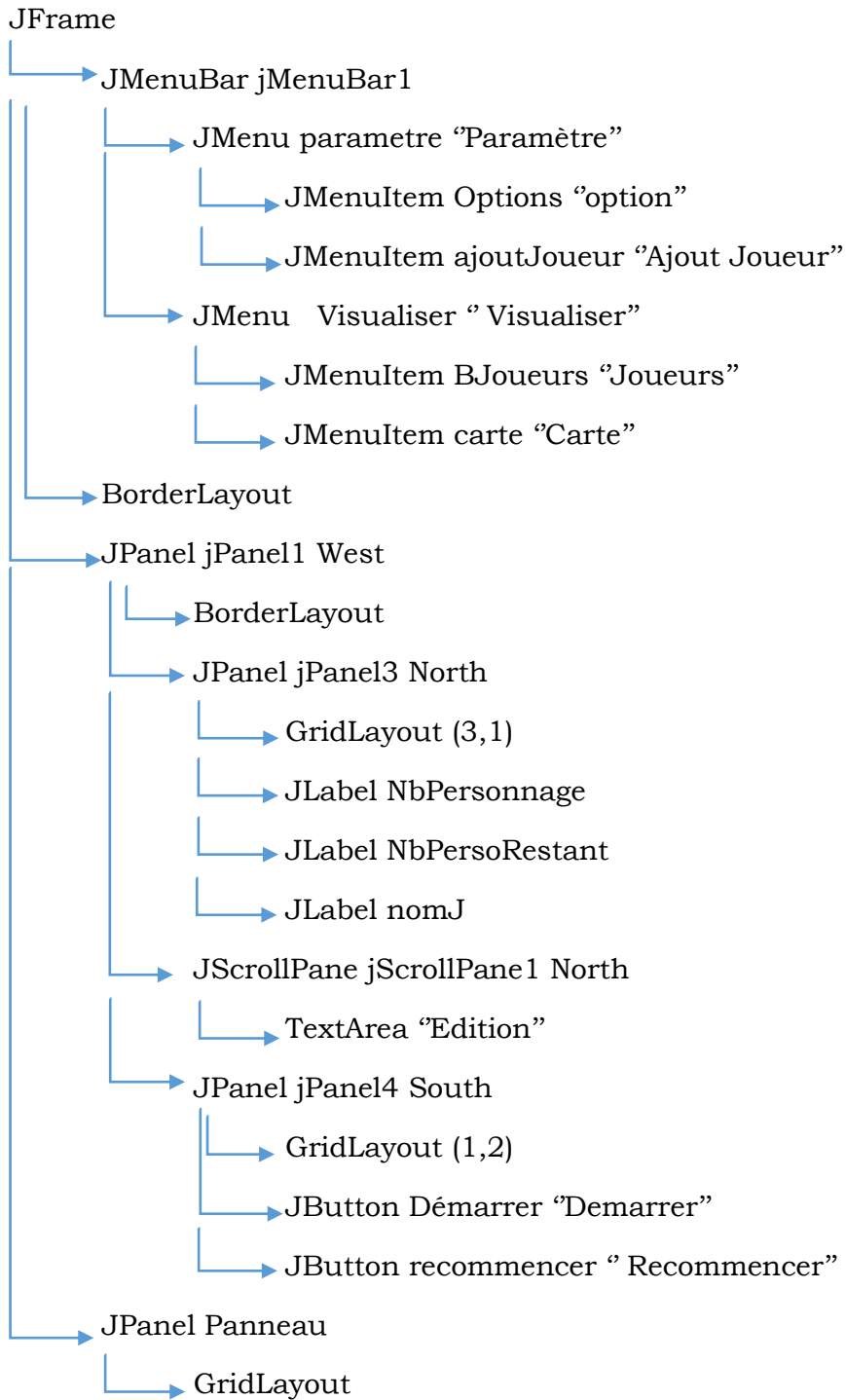
La fenêtre « BatailleDlg » s'ouvre lorsqu'un joueur courant obtient tous les personnages de la famille légendaire ou épiques, cette boîte de dialogue reçoit de la fenêtre principale des informations sur l'indice du joueur courant et la liste des joueurs. La méthode `initListeJ()` est appelé dès l'ouverture de la boîte, elle permet de remplir une liste nommée `liste` (de type `Jlist`) avec le pseudo de tous les joueurs.

Une fois qu'un joueur est sélectionné dans la liste, le gestionnaire d'évènement associé à la sélection d'un joueur dans la liste (`listeMouseClicked`) vérifie s'il ne s'agit pas du joueur courant et crée une instance de la Bataille, ensuite les informations sur les paquets du joueur courant et le joueur sélectionné sont affichées respectivement sur `InfosCarte1` et `InfosCarte2` qui sont de type `JTextArea`.

La bataille commence lors du clic sur le bouton Démarrer et le déclenchement de ce gestionnaire récupère le premier personnage de chacun des deux joueurs et l'affiche respectivement sur `Carte1` et `Carte2` qui sont de type `JButton` et crée une instance de la classe `Bataille` et fait appel à sa méthode `execute()` qui permet réaliser la comparaison des deux premières cartes respectives du paquet des joueurs. Elle renvoie la valeur 1 si le joueur courant gagne, la valeur 2 si l'adversaire gagne, la valeur 0 en cas d'égalité

A la fin de l'action, la boîte de dialogue renvoie l'information sur l'indicateur boolean à travers l'accessor `getOk()` et sur le déroulement du transfert à travers l'accessor `getDeroulement()`.

#### IV. Arborescence de la fenêtre principale « JeuMemory »



#### V. Explication sur les attributs et le constructeur

La classe « JeuMemory » comporte plusieurs attributs :

- Un attribut **persos** de type **LesPersonnages**, c'est une liste qui contient les personnages en fonction du niveau du jeu choisi
- Un attribut **joueurs** de type **LesJoueurs**, c'est une liste qui contient tous les joueurs sélectionnés ou ajoutés
- Un attribut **monJeu** de type **Jeu** qui permet le déroulement du jeu en accédant à ses méthodes depuis la classe principale
- Des attributs **l1,c1,l2,c2** de type **int**, ces attributs permettent de gérer l'indice des cartes sur le panneau
- Un attribut **indC** de type **int** qui permet de gérer le tour des joueurs, cet attribut représente l'indice du joueur courant
- Des attributs **bouton1** et **bouton2** qui permettent de récupérer respectivement le name du premier bouton cliqué c'est-à-dire le premier personnage retourné et name du deuxième bouton.

Dans le constructeur j'ai initialisé l'attribut **joueurs** et **persos** en créant des nouvelles instances dont le constructeur de l'attribut **persos** prend en paramètre 4 (niveau du jeu par défaut) et j'ai initialisé les attributs **l1, c1, l2, c2** à -1 pour dire qu'aucune carte n'est retournée au lancement du jeu et j'ai aussi initialisé l'indice du joueur à zéro et j'ai invalidé le bouton **recommencer**, les sous options **carte** et **Joueurs**. Aucune méthode est appelée dans le constructeur.

## **VI. Explication des différentes étapes du déroulement du jeu**

### **a- Les actions à réaliser avant le démarrage du jeu**

Avant le démarrage du jeu il est nécessaire de sélectionner au moins deux joueurs et le niveau de jeu souhaité via la boîte de dialogue « InitDlg », les joueurs et le niveau sélectionnés sont renvoyés à travers les accesseurs de cette boîte de dialogue mais il est possible aussi par la boîte de dialogue « SaisieJoueurDlg » de créer un nouveau joueur par la saisie des informations correspondant à ce joueur et c'est également par l'accesseur de cette classe que le joueur créé est récupéré dans la fenêtre principale, ce sont les actions à gérer avant le clic sur le bouton Démarrer.

### **b- Les actions à effectuer ou possible pendant le jeu**

Pendant le déroulement du jeu plusieurs actions est possible d'être faites, le joueur courant a le droit de retourner deux cartes et si les personnages se trouvant sur ces deux cartes sont les mêmes alors elles sont rendues invalides sinon les cartes sont retournées et le tour passe au joueur suivant, il est possible aussi d'ouvrir la boîte de dialogue « SaisiePersonnage » et de visionner les photos des personnages que le joueur courant a déjà gagnés au

cours du jeu, cette boîte de dialogue reçoit le joueur courant à travers son constructeur et il est également possible de visualiser les informations des joueurs grâce à la fenêtre de dialogue « VisuJoueursDlg » qui affiche les informations relatives à chaque joueur lorsqu'on clique sur sa photo : son pseudo, sa famille préférée, les personnages en sa possession, ainsi que son score (somme des valeurs des personnages acquis).

Lorsqu'au cours du jeu le joueur courant réussit à obtenir tous les personnages de la famille « rares » ou « communs », Une boîte de dialogue «TransfertDlg» s'ouvre. Elle propose au joueur courant de sélectionner un joueur dans une liste déroulante contenant le pseudo de tous les joueurs et de transférer tous les personnages d'une famille du joueur sélectionné.

Egalement lorsque le joueur courant réussit à obtenir tous les personnages de la famille « légendaires » ou « épiques », Une boîte de dialogue « BatailleDlg » s'ouvre. Elle propose au joueur courant de sélectionner un joueur dans une liste « ListeJ » qui contient le pseudo de tous les joueurs, le joueur courant à la possibilité d'engager une bataille ou de l'annuler

### **c- Les actions liées à l'arrêt du jeu**

Une fois qu'un joueur réussit à obtenir tous les personnages de sa famille préférée ou lorsqu'il n'y a plus de carte à retourner le jeu s'arrête et indique le gagnant dans la zone de texte.