

**Rapport intermédiaire sur le projet à rendre pour le vendredi 1er avril 2022**

**Le programme a été réalisé en binôme avec Hoarau Tristan**

**1) JeuMemory**

Les attributs de la classe JeuMemory sont :

- joueurs de la classe LesJoueurs qui nous permettra de mettre l'ensemble des joueur de la partie dans une seule variable.
- persos de la classe LesPersonnages qui nous permettra de mettre l'ensemble des personnages de la partie dans une seule variable

Explications du constructeur :

```
public JeuMemory(){ //Pas de paramètres
```

```
// initialisation des composants sur la JFrame  
    initComponents();
```

```
// créer une nouvelle instance de la classe LesPersonnages qui elle, crée une ArrayList vide  
    this.persos = new LesPersonnages();
```

```
// créer une nouvelle instance de la classe LesJoueurs qui elle, crée une ArrayList vide  
    this.joueurs = new LesJoueurs();  
}
```

Tous les attributs de la classe sont bien initialisés.

## 2) SaisieJoueurDlg

- a) Le rôle de la boîte de dialogue SaisieJoueurDlg est de créer un nouveau joueur.
- b) et c) Elle possède 4 variables globales qui sont j pour stocker le joueur qui va être créé, la variable booléenne ok pour savoir si le joueur a été créé ou non, photo de type ImageIcon qui va nous permettre d'attribuer une photo au joueur créé et la variable lp de la classe LesPersonnages qui va nous permettre d'initialiser la JList des noms des familles.

Le constructeur de SaisieJoueurDlg prend 3 paramètres qui vont nous permettre par la suite de mettre SaisieJoueurDlg en communication avec la classe principale JeuMemory :

Frame parent : le parent qui sera JeuMemory, on mettra this au moment de l'appel dans JeuMemory

Boolean modal : capture le focus de la fenêtre jusqu'à ce qu'elle soit fermée ou non

LesPersonnages lp : la classe SaisieJoueurDlg reçoit une liste de personnages pour pouvoir initialiser la JList des noms des familles des personnages.

Donc SaisieJoueurDlg reçoit comme information une liste de personnages et peut renvoyer le joueur créé avec l'accessor getJoueur() mais aussi le booléen ok avec l'accessor getOk() qui comme dit précédemment nous permettra de savoir si le joueur a bien été créé.

Pour Initialiser les noms des familles dans la JList ListeFamilles il faut cliquer sur le sous-menu Ajouter joueur du menu Paramètres dans JeuMemory pour que l'interface de SaisieJoueurDlg s'ouvre, à ce moment-là dans le constructeur de cette interface on a une méthode initListeFamilles() qui va ajouter un par un les noms des familles des personnages déjà présents dans le jeu à la JList ListeFamilles et cela sans répétition de nom de famille.

Illustration d'échange d'informations entre SaisieJoueurDlg et jeuMemory :

Ce code a été écrit dans la classe jeuMemory :

```
private void AjoutJoueurActionPerformed(java.awt.event.ActionEvent evt) {  
    SaisieJoueurDlg sjd = new SaisieJoueurDlg(this,true,this.persos); //appel du constructeur de la JDialog  
    sjd.setVisible(true); //affichage de la JDialog  
    if(sjd.getOk()) //si le booléen renvoyé par la JDialog a la valeur true c'est-à-dire qu'un joueur a été créé  
    {  
        this.joueurs.ajouteJoueur(sjd.getJoueur()); //on ajoute le joueur renvoyé par la JDialog à la liste des joueurs  
    }  
}
```

c) Arborescence de l'interface de la JDialog SaisieJoueurDlg :

JDialog SaisieJoueurDlg

BorderLayout

JPanel jpanel1 North

FlowLayout

JLabel jLabel1 « Créez votre joueur »

JPanel jpanel2 West

GridLayout(3,1)

JPanel jpanel4

BorderLayout

JLabel jLabel6 «Donnez votre pseudo :» North

JScrollPane jScrollPane3 Center

JTextArea Pseudo

JPanel jpanel5

BorderLayout

JLabel jLabel5 «Quelle est votre famille préférée des personnages :» North

JScrollPane jScrollPane4 Center

JList ListeFamilles

JPanel jpanel6

BorderLayout

JLabel jLabel7 «Choisissez une photo :» North

JButton Parcourir Center

JPanel jpanel2 East

BorderLayout

JButton Photo Center

JPanel jpanel2 South

GridLayout(1,2)

JButton Annuler « Annuler »

JButton Valider « Valider »

d) Les événements gérés par la boîte de dialogue SaisieJoueurDlg :

Le Bouton Parcourir : **private void ParcourirActionPerformed(java.awt.event.ActionEvent evt)**

```
//On crée une variable
```

```
JFileChooser jf= new JFileChooser();
```

```
// Si un fichier a été choisi
```

```
if (jf.showOpenDialog(this)== JFileChooser.APPROVE_OPTION)
```

```
{
```

```
    //On récupère le chemin d'accès à la photo
```

```
    String path = jf.getSelectedFile().getPath();
```

```
    //On récupère l'image dans l'ordi
```

```
    Image img=Toolkit.getDefaultToolkit().getImage(path);
```

```
    //On redimensionne la photo pour qu'elle ne prenne ni plus ni moins de place que le bouton Photo
```

```
    img=img.getScaledInstance(Photo.getWidth()-10, Photo.getHeight()-10, Image.SCALE_DEFAULT);
```

```
    //On construit l'ImageIcon à partir de l'image sélectionnée
```

```
    photo = new ImageIcon(img);
```

```
    //On met l'image sur le bouton
```

```
    Photo.setIcon(photo);
```

```
}
```

Le bouton Valider : **private void ValiderActionPerformed(java.awt.event.ActionEvent evt)**

```
if(!this.Pseudo.getText().equals("")) //on vérifie qu'un pseudo a bien été saisi
{
    if(this.ListeFamilles.getSelectedIndex() != -1) //on vérifie qu'une valeur a bien été sélectionnée
    dans la liste
    {
        //si les conditions précédentes sont réalisées, on crée un nouveau joueur avec le pseudo et la
        famille sélectionnée
        this.j = new Joueur(this.Pseudo.getText(),this.ListeFamilles.getSelectedValue());
        if(this.photo != null) //si une photo a bien été sélectionnée
        {
            j.setPhoto(photo); //on applique cette photo au joueur créé
            this.setVisible(false); //on ferme la JDialog
            this.dispose(); //fermeture de l'instance associée
            this.ok=true; //on met le booléen à true, pour spécifier qu'un joueur a bien été créé
        }
    }
    else
    {
        //si aucune famille n'a été sélectionnée, on applique au joueur la première famille de la liste
        j.setFamillePref(this.ListeFamilles.getModel().getElementAt(0));
    }
}
else
{
    //si aucun pseudo n'a été rentré, on met un message d'erreur dans la JTextArea
    this.Pseudo.setText("Vous n'avez pas saisi de pseudo !!!");
}
```

Le bouton annuler : **private void AnnulerActionPerformed(java.awt.event.ActionEvent evt)**

//si on clique sur le bouton Annuler de la JDialog, celle-ci se ferme et le booléen est mis à false, pour indiquer qu'aucun joueur n'a été créé

```
this.ok=false;  
this.setVisible(false);  
this.dispose();
```

### **3) VisuPersonnagesDlg**

a) Le rôle de la boîte de dialogue VisuPersonnagesDlg est de permettre la visualisation des personnages gagnés par un joueur.

b) ET e)

VisuPersonnagesDlg possède une variable globale joueur de la classe Joueur, cette variable globale nous permettra de visualiser les personnages gagnés par ce joueur en question.

Le constructeur de VisuPersonnagesDlg prend 3 paramètres qui vont nous permettre par la suite de mettre VisuPersonnagesDlg en communication avec la classe principale JeuMemory :  
Frame parent : le parent qui sera jeuMemory, on mettra this au moment de l'appel  
Boolean modal : capture le focus de la fenêtre jusqu'à ce qu'elle soit fermée ou non  
Joueur joueur : ce paramètre va nous permettre de pouvoir visualiser les personnages gagnés par ce joueur.

Donc VisuPersonnagesDlg reçoit comme information un joueur et n'a besoin de renvoyer aucune information à la classe principale JeuMemory car toutes les informations seront visibles son l'interface.

Pour La visualisation des personnages d'un joueur il faut cliquer sur le sous-menu Cartes du menu Visualiser dans JeuMemory pour que l'interface de VisuPersonnagesDlg s'ouvre, à ce moment là dans le constructeur de cette interface on a une méthode initPanneau() qui va nous permettre d'afficher tous les personnages gagnés par le joueur et cela se passe par l'accesseur getPaquet() de la classe joueur.

Avec la méthode initPanneau(), nous créons dynamiquement des boutons pour chaque personnage dans le paquet du joueur et l'ajoutons au trombinoscope.

Illustration d'échange d'informations entre VisuPersonnagesDlg et jeuMemory :

Ce code a été écrit dans la classe jeuMemory :

```
private void CartesActionPerformed(java.awt.event.ActionEvent evt) {  
    Joueur j=new Joueur("FanMemory", "commun"); //création d'un joueur test  
    j.initPaquetTest(); //on lui affecte un paquet test  
    VisuPersonnagesDlg vpd = new VisuPersonnagesDlg(this,true,j); //appel du constructeur de la JDialog  
    vpd.setVisible(true); //affichage de la JDialog  
}
```

c) Arborescence de l'interface de la JDialog VisuPersonnagesDlg:

JDialog VisuPersonnagesDlg

BorderLayout

JLabel Joueur «Cartes du joueur :» North

JPanel trombinoscope Center

JLabel Score « Score : » South

JButton Afficher « Afficher » South

d) Les événements gérés par la boîte de dialogue VisuPersonnagesDlg :

Le bouton afficher : **private void AfficherActionPerformed(java.awt.event.ActionEvent evt) :**

```
//on parcourt la liste des personnages que possède le joueur
for(int i=0;i<this.joueur.getPaquet().getTaille();i++)
{
    //on récupère le bouton d'indice i dans le JPanel
    JButton jb = (JButton)trombinoscope.getComponent(i);

    //on affecte à ce bouton la photo du personnage correspondant
    this.joueur.getPaquet().getPerso(i).setImgBouton(jb);
}
```

**FIN**