

TP3

L'objectif de ce TP est de renforcer les acquis en matière de gestion de composants et d'écriture des gestionnaires d'événements. Mais ce TP est surtout l'occasion de voir qu'un manque d'architecture peut vite aboutir à du code dans lequel les composants sont tous interdépendants. Le remplacement d'un composant ou l'ajout d'un nouveau composant peut alors devenir lourd et le risque d'oublier une mise à jour augmente très rapidement.

Exercice – album photo

Nous proposons de réaliser un album photo, comme illustré à la figure 1. L'application initialise l'album à partir de photos (`images.zip`) situées dans un répertoire prédéfini et dispose d'un menu permettant d'ajouter d'autres photos. Une liste à gauche permet de sélectionner le nom de la photo à afficher dans le panneau central. Il est également possible de sélectionner la photo à afficher en cliquant sur l'une des icônes du bandeau bas. De plus, l'utilisateur peut passer d'une photo à la précédente ou à la suivante en cliquant sur les boutons correspondants en haut. Enfin, un curseur à droite permet de redimensionner la photo couramment affichée. Le noyau applicatif de cette application est fourni. Il est constitué des classes `Photo.java` qui mémorise non seulement l'image correspondant à la photo mais aussi un facteur de zoom, et `Album.java` dont l'état est une liste de `Photo`. La javadoc de ces classes est également fournie (`doc.zip`).

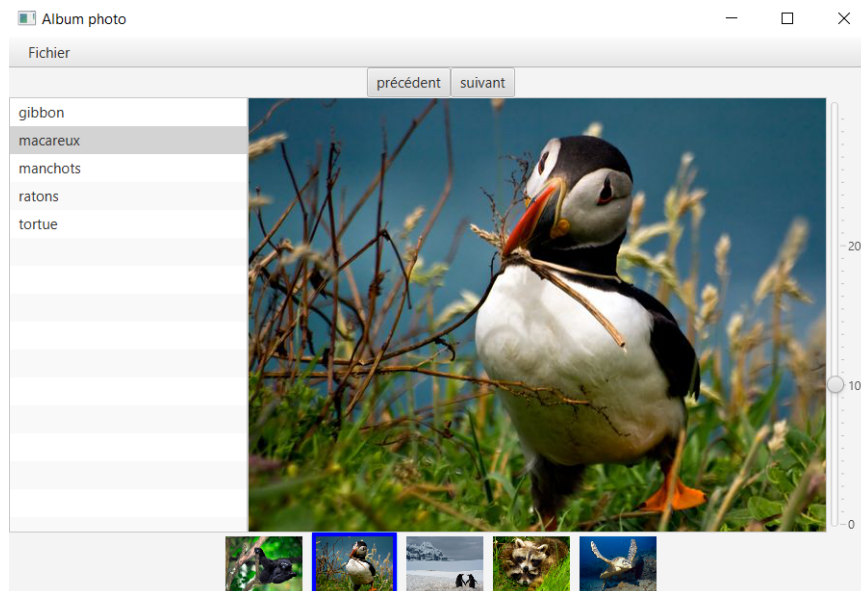


FIGURE 1 – Album photo

1. Créez une classe `Visionneuse` correspondant à la fenêtre principale de l'application.
2. Écrivez une méthode `creerCentre()` qui crée le panneau central de taille préférée 600×450 contenant la photo courante (une `ImageView`), alignée à gauche. Le panneau est ensuite ajouté au conteneur principal de la fenêtre.

3. Le curseur à droite permet de modifier le facteur zoom de la photo courante.
 - (a) Écrivez une méthode `creerSlider()` qui crée et paramètre le curseur (un `Slider`). Consultez la documentation de cette classe pour afficher et espacer les tirets principaux de 100 et les petits de 10, ainsi que pour afficher les valeurs des unités. Le curseur est ensuite ajouté au conteneur principal de la fenêtre.
 - (b) Ajoutez un gestionnaire d'événements au curseur qui met à jour le facteur zoom de la photo courante et la redimensionne en conséquence dans le panneau central.
4. La liste à gauche permet de sélectionner une seule photo à partir de son nom, qui devient alors la photo courante affichée dans le panneau central.
 - (a) Écrivez une méthode `creerListe()` qui crée la liste (une `ListView<String>`) à gauche de l'interface, initialisée avec les noms des photos de l'album. Utilisez la méthode `FXCollections.observableArrayList` pour initialiser le contenu de la liste à sa création et la méthode d'instance `getSelectionModel()` pour agir sur la sélection des items. La liste est ensuite ajoutée au conteneur principal de la fenêtre.
 - (b) Ajoutez un gestionnaire d'événements à la liste qui (1) met à jour le modèle en faisant de la photo choisie la photo courante, (2) affiche la photo sélectionnée dans le panneau central et (3) met à jour le curseur afin qu'il corresponde au facteur de zoom de la photo sélectionnée.
5. Le bandeau bas est constitué de boutons.
 - (a) Écrivez une méthode `creerBandeauBas()` qui crée un bandeau (un `Pane`) composé d'autant de boutons qu'il y a de photos dans l'album. Le panneau est ensuite ajouté au conteneur principal de la fenêtre. L'image de ces boutons peut être obtenue en utilisant la méthode `getIcône()` de la classe `Photo`. Les boutons n'ont pas de bordure.
 - (b) Ajoutez à chacun de ces boutons un gestionnaire d'événements qui (1) met à jour le modèle en faisant de la photo choisie la photo courante, (2) rajoute une bordure bleue de 4 pixels autour de l'icône sélectionnée (les autres icônes n'ont pas de bordure), (3) affiche la photo sélectionnée dans le panneau central, (4) met à jour le curseur afin qu'il corresponde au facteur de zoom de la photo sélectionnée et (5) met à jour la sélection de la liste pour qu'elle corresponde à la photo sélectionnée.
 - (c) Modifiez le gestionnaire d'événements de la liste de sorte à ce que l'icône sélectionnée dans le bandeau bas (celle avec une bordure bleue) corresponde à la sélection de la liste.
6. Le bandeau haut est constitué de deux boutons.
 - (a) Écrivez une méthode `creerBandeauHaut()` qui crée un panneau composé de deux boutons «précédent» et «suivant». Les boutons sont ensuite ajoutés au conteneur principal de la fenêtre.
 - (b) Ajoutez aux boutons «précédent» et «suivant» un gestionnaire d'événements qui (1) met à jour le modèle en faisant de la photo précédente/suivante (*i.e.*, d'index immédiatement inférieur/supérieur) la photo courante, (2) rajoute une bordure bleue de 4 pixels autour de l'icône de la photo courante (les autres icônes ne doivent pas avoir de bordure), (3) affiche la photo sélectionnée dans le panneau central, (4) met à jour le curseur afin qu'il corresponde au facteur de zoom de la photo sélectionnée, (5) met à jour la sélection de la liste pour qu'elle corresponde à la photo sélectionnée et (6) ajuste la disponibilité des boutons «précédent» et «suivant» selon que la photo courante soit la première (pas de précédente), la dernière (pas de suivante) ou au milieu d'autres photos.
 - (c) Modifiez les gestionnaires d'événements de la liste et des boutons du bandeau bas pour qu'ils mettent à jour la disponibilité des boutons «précédent» et «suivant» en fonction de la position de la photo courante dans l'album.

7. L'application dispose d'un menu *Fichier* qui comporte deux items (séparés par un séparateur) : l'un pour ajouter une photo à l'album et l'autre pour quitter l'application.
 - (a) Écrivez une méthode `creerMenu()` qui crée une bar de menu (une `MenuBar`), composé d'un `Menu Fichier` constitué d'un `MenuItem Ajouter une photo`, d'un séparateur (un `SeparatorMenuItem`) et d'un `MenuItem Quitter`. La bar de menu est ensuite ajoutée au conteneur principal de la fenêtre.
 - (b) Ajoutez un gestionnaire d'événements à l'item de menu *Quitter* afin qu'il ferme l'application (appel à `System.exit(0)`).
 - (c) Ajoutez un gestionnaire d'événements à l'item de menu *Ajouter une photo* afin qu'il
 - (1) ouvre une boîte de dialogue (un `FileChooser`) permettant de choisir une photo,
 - (2) ajoute la photo sélectionnée dans le modèle et
 - (3) mette à jour tous les autres composants impactés par cet ajout.