

Evaluation Desktop

Principe

On va récupérer le même jar que pour l'évaluation JSP, mais ce sera beaucoup plus simple. Pensez à prendre la dernière version du jar.

Éléments extérieurs

Pour pouvoir faire ce projet, un jar va vous être fourni : *pseudoerrors.jar*.

N'oubliez pas de :

- Créer un répertoire *lib* dans votre projet
- Copier ce jar dans *lib*
- Faire le lien vers ce jar dans le projet Eclipse

Ce jar contient 2 classes :

- ErrorItem qui correspond à chaque élément de la liste. Cette classe possède les attributs suivants :
 - long id
 - String type
 - String label
 - int level
 - String commentaire
 - int status qui peut prendre différentes valeurs :
 - STATUS_UNREAD (non lu)
 - STATUS_READ (lu)
 - STATUS_COMM (commenté)
 - STATUS_PROCESSED (traité)
 - Pour accéder à ces valeurs, on fait, par exemple, ErrorItem.STATUS_UNREAD
- ErrorDao qui est un pseudo-DAO avec 3 méthodes :
 - List<ErrorItem> list()
 - void delete(long id)
 - ErrorItem getById(long id)

A noter : Initialement toutes les erreurs renvoyées par le list() sont en status UNREAD. De plus les id des ErrorItem dans la liste ne sont pas consécutifs (on n'a **pas** id=1, 2, 3, etc)

Obligations

Pour accéder au DAO il **FAUT** utiliser un service.

Remarques

- Inutile de passer par un DTO, on peut travailler directement avec ErrorItem (le toString() est bon)
- Toute la documentation nécessaire se trouve dans le Pdf sur Swing.

Unique Frame

On affiche une Frame qui contient une JList qui affiche les ErrorItem.

Cette Frame contient aussi un bouton Supprimer qui permet de détruire un élément de la JList après popup de confirmation.

Si vous avez le temps : faire en sorte que la liste puisse scroller verticalement en utilisant un JScrollPane

Si vous avez encore plus de temps : faire en sorte qu'une popup de confirmation s'affiche quand vous fermez la fenêtre (en cliquant sur la croix en haut à droite). Pour cette option, cherchez par vous-même, parce que ce n'est pas dans le Pdf sur Swing.