

TP Java EE - 3

Consignes :

Le code que vous utiliserez pour ce TP est celui que vous avez vous même produit lors du/des TPs précédents.

Vous êtes libres d'utiliser l'environnement de développement que vous désirez.

Dans la série de TP, vous allez réaliser un livre d'or en partant d'un squelette pré-rempli. L'issue de cet ensemble de TP donnera lieu à une note, donc soyez sérieux lors de la réalisation. Les sujets sont conçus de façon à ce que peu de travail intermédiaire soit nécessaire.

Note sur le code produit :

Cet énoncé ne donne pas de nom aux classes et aux actions de façon volontaire pour que vous soyez libre de faire comme bon vous semble.

Cependant, lorsque vous programmez, pensez à la relecture future de ce que vous produisez, que ce soit pour vous ou pour un autre. Ainsi, aérez votre code, nommez correctement vos variables, méthodes et classes.

Slides du cours :

<http://bit.ly/2018-JAVAEE-SLIDES>

Partie 1 : Découverte des API REST

Démarrer votre application avec la même commande maven que pour les TPs précédent, puis rendez-vous sur l'url <http://localhost:8080/tpjavaee/api-docs/>

Voici une documentation d'api directement branchée à notre code métier.

Cette interface nous permet de consulter et faire des appels REST pour interagir avec notre application.

Actuellement uniquement la création et la récupération de *User* sont branchés.

Vous pouvez déjà faire plusieurs essais avec cette interface pour les tester.

Partie 2 : Nouveau service REST

Nous allons créer une nouvelle méthode d'API permettant de récupérer la liste des *GoldenBookEntry* et la documenter pour qu'elle apparaisse dans cette interface (vous pouvez prendre pour référence la classe *UserWebService*).

- Créez une nouvelle classe *GoldenBookEntryWebService* dans le package `com.codetroopers.eput.ws`
- Cette classe doit étendre la classe `javax.ws.rs.core.Application`
- Injectez le *GoldenBookEntryService* en attribut de la classe

- Créez une méthode qui retourne une liste de *GoldenBookEntry* et dont le corps sera un appel *goldenBookEntryService.loadGoldenBookEntries()*
- Ajoutez l'annotation `javax.ws.rs.GET` sur cette méthode
- Ajouter les 3 annotations suivantes sur la classe
 - `@ApplicationPath("ws")`
 - `@Path("/entries")`
 - `@Produces(MediaType.APPLICATION_JSON)`
- Redémarrez votre serveur, une nouvelle méthode REST est maintenant disponible

Partie 3 :

Jugeant l'interface actuelle peu conviviale, une équipe s'est chargée de la réécriture de celle-ci en utilisant des technologies de rendu côté client (Vue.js). Le résultat de ce travail est mis à disposition à l'adresse suivante :

<https://drive.google.com/open?id=1ADLbIIEg2fIFa7GRMIIBQMWWHwN-1D0>

Récupérez l'archive et décompressez-la pour l'intégrer à votre projet. Les fichiers de celle-ci doivent être copiés dans `src/main/webapp`.

Redémarrez votre serveur et allez sur <http://localhost:8080/tpjavaee/index.html> vous devriez voir la nouvelle interface en action. Pour l'instant rien ne s'affiche et des erreurs javascript trahissent le manque de développement de votre côté.

Partie 4 :

L'équipe en charge de cette refonte visuelle s'est basée sur l'utilisation exclusive de webservices pour communiquer avec l'application existante, cependant, bien que le code métier existe, les webservices pas encore.

La documentation de leurs attentes est présente dans le fichier `API.adoc` (ou à l'adresse <https://drive.google.com/file/d/1s01dyyhy33NnESL7seYRtgijWpYDM4E>) présent dans l'archive. Implémentez les webservices nécessaires en vous référant à la façon dont vous l'avez fait dans la partie 2.

Pensez à utiliser la documentation d'API expliquée dans la partie 1 également lors de vos développements.

Partie 5 :

Aspect critique, vous consignerez dans le document `TODO.md` à la racine du projet vos remarques quant à la structure actuelle du projet et ses éventuelles lacunes ainsi que quelques perspectives d'amélioration.

Partie 6 : Travail terminé

Fin des TPs, vous allez réaliser une archive du travail réalisé et la remettre à l'enseignant.

1. Nettoyez le répertoire de travail via la commande *mvn clean* afin de supprimer les fichiers créés lors de la compilations des sources.
2. Créez une archive de l'ensemble des sources (zip ou rar) (il s'agit du dossier contenant le fichier pom.xml, Readme.md, TODO.md et surtout le repertoire src). Pour information l'archive doit faire environ 30Mo, et en aucun cas plus de 50Mo...
3. Nommer cette archive avec année, votre (ou vos si vous travaillez en binôme) numéro d'étudiant et nom, prénom (ex :
2018_4561208_Hallyday_Johnny_4981378_Mitchell_Eddie)
4. Uploadez l'archive à l'adresse <https://www.dropbox.com/request/Ry9mT00aY3dYMHFwleMA> et allez voir l'enseignant pour valider le bon upload. Vous recevrez un mail vous confirmant le bon upload. La page d'upload sera automatiquement close à l'issue du TP.