

## TP

### TP MVVM - Zk Framework

L'objectif de ce deuxième TP est d'appréhender le pattern MVVM (Model-View-ViewModel) à l'aide du framework Zk de Zkoss (<http://www.zkoss.org>). Le TP va consister à implémenter une table interactive et un camembert, en utilisant le Zk Framework.

#### Exercice 1.1 (Installation et configuration)

1. Le Zk framework nécessite la mise en place d'un serveur web (eg. Tomcat) et de l'IDE Eclipse JEE. Suivez des instructions disponibles ici : [https://www.zkoss.org/wiki/ZK\\_Studio\\_Essentials/Installation](https://www.zkoss.org/wiki/ZK_Studio_Essentials/Installation)
2. L'installation de Zk se réalise ensuite via le Marketplace Eclipse (Help>Eclipse Marketplace) et cherchez `zk studio`. Les instructions d'installation sont détaillées dans l'URL ci-dessus.
3. Testez votre configuration avec un exemple comme décrit ici : [https://www.zkoss.org/wiki/ZK\\_Installation\\_Guide/Quick\\_Start/Create\\_and\\_Run\\_Your\\_First\\_ZK\\_Application\\_with\\_Eclipse\\_and\\_ZK\\_Studio](https://www.zkoss.org/wiki/ZK_Installation_Guide/Quick_Start/Create_and_Run_Your_First_ZK_Application_with_Eclipse_and_ZK_Studio). Le projet peut également être créé avec Maven [https://www.zkoss.org/wiki/ZK\\_Installation\\_Guide/Quick\\_Start/Create\\_and\\_Run\\_Your\\_First\\_ZK\\_Application\\_with\\_Eclipse\\_and\\_Maven](https://www.zkoss.org/wiki/ZK_Installation_Guide/Quick_Start/Create_and_Run_Your_First_ZK_Application_with_Eclipse_and_Maven).

#### Exercice 1.2 (Table interactive)

Commencez par réaliser la table interactive (interactive au sens où les champs sont modifiables par l'utilisateur).

1. créer une classe JAVA qui s'appelle `Budget` et qui représente votre modèle de données, avec un champ `intitule` (string), un champ `description` (string), et un champ `valeur` (float).
2. implémentez le composant graphique de la table en ZUL (le langage XML de description d'interfaces de Zk), à l'aide du composant `<grid>`. Inspirez vous de l'exemple ci-après : [https://www.zkoss.org/wiki/ZK\\_Getting\\_Started/Tutorial](https://www.zkoss.org/wiki/ZK_Getting_Started/Tutorial) (partie MVVM)
3. implémentez votre `ViewModel` et utilisez des mécanismes de binding bi-directionnel pour faire des modifications des données du `Model` à partir de la `View`.
4. ajouter dans le ZUL trois champs (intitulé, description, valeur) et un bouton de validation pour permettre à l'utilisateur de saisir une nouvelle entrée dans le tableau. On vérifiera que les champs intitulé et description ne sont pas vides avant d'ajouter les données dans le modèle.

#### Exercice 1.3 (Camembert interactif)

Réalisez ensuite le Camembert interactif

1. Utilisez le composant Chart/Pie décrit ici pour représenter graphiquement votre camembert [https://www.zkoss.org/zkchartsdemo/pie\\_basic](https://www.zkoss.org/zkchartsdemo/pie_basic) en liant la représentation graphique aux données du modèle
2. Vérifiez le bon fonctionnement de l'application (à chaque modification de donnée sur la table correspond un changement dans le camembert).