
TP 3 : Plugin de Ressources et Spring Security

M2 MBDS
20/11/2013



Twitter

Présentation Générale

L'objectif de ce TP est d'ajouter de nouvelles fonctionnalités à votre site via les deux plugins :

- Ressources : <http://grails.org/plugin/ressources>
- Spring Security : <http://grails.org/plugin/spring-security-core>

Objectif

1/ Utilisation du Plugin de Resource

Le plugin "Resource" devrait déjà être installé dans votre projet, vérifiez dans le module de plugins de votre projet.

Vérifiez dans le fichier BuildConfig.groovy et dans le module de Plugins de votre projet qu'il est bien présent.

Créez un nouveau fichier CSS dans le dossier web-app/css pour appliquer de nouveaux styles à votre site.

Créez un nouveau fichier JS dans le dossier web-app/js pour regrouper tout le Javascript dont vous pourriez éventuellement avoir besoin.

Référez ces deux fichiers dans le fichier de config "ApplicationResources.groovy" en créant un nouveau module "custom" qui contiendra :

- Vos deux fichiers nouvellement créés
- JQuery pour pouvoir utiliser quelques fonctions comme les appels Ajax.

N'oubliez pas de référencer votre module nouvellement créé dans la page via la méthode :

```
<r:require module="custom"/>
```

Une fois ceci fait, les ressources définies seront accessible dans les pages qui contiendront cette annotation.

Ajoutez du contenu à vos fichiers CSS et JS nouvellement créés et vérifiez que vos ajouts sont bien pris en compte.

Maintenant que tout fonctionne, appliquez quelques styles à votre page pour améliorer le style et la lisibilité.

Point de validation : Appelez-moi pour me montrer la configuration, les fichiers créés et les styles appliqués à vos pages

2/ Utilisation du Plugin Spring Security

Le plugin "Spring Security" va devoir être installé, pour ceci, allez dans le fichier "BuildConfig.groovy" et ajoutez dans la partie

plugin la ligne :

```
runtime ":spring-security-core:1.2.7.3"
```

Compilez votre projet pour lancer le téléchargement.

Vérifiez ensuite la présence du plugin dans le module de plugins.

Une fois ceci fait, exécutez la commande :

```
s2-quickstart DOMAIN_CLASS_PACKAGE USER_CLASS_NAME ROLE_CLASS_NAME
```

Remplacez :

DOMAIN_CLASS_NAME par le package de base défini au tout début du projet

USER_CLASS_NAME par le nom de votre nouvelle classe "Utilisateur" qui sera utilisé par Spring Security pour la gestion des utilisateurs (ne mettez pas le même nom que la classe d'utilisateur que vous avez déjà)

ROLE_CLASS_NAME par le nom de la classe qui gèrera les rôles de sécurité.

Trois classes vont être créées :

- User
- Role
- UserRole (une association des deux précédentes classes)

Inspectez le fichier "Config.groovy" et constatez les modifications faites par le plugin.

Ajoutez à la classe User nouvellement créée les attributs que vous aviez définis dans votre classe Utilisateur et remplacez l'ancienne par la nouvelle dans votre projet. Vous devrez effectuer des modifications dans tout votre projet afin de modifier toutes les références vers l'ancienne classe. N'oubliez pas de modifier le BootStrap en conséquence.

Modifiez votre projet de façon à obliger un utilisateur à s'identifier avant de pouvoir consulter son compte et modifier ses informations.

Point de validation : Appelez-moi pour me montrer l'intégration de la sécurité que vous avez appliqué

Rendu

Si vous avez des questions, adressez-moi un mail à l'adresse greg.galli@tokidev.fr, je ferai de mon mieux pour revenir vers vous le plus rapidement possible.