Partie 3 – Moteur de recherche RESTFul

Nous allons étudier les différentes étapes du développement d'une application Spring Boot. Ces étapes étant dépendantes, il est indispensable que chaque partie soit entièrement réalisée avant de passer à la suivante

1 – Création de la classe du domaine Partnership

Un partenariat décrit le partenariat réalisé par une entreprise sur un projet. Un projet est donc proposé par une entreprise mais peut faire l'objet de plusieurs partenariats, c'est à dire de plusieurs entreprises partenaires intervenant sur le projet de différentes manières.

1. Créez la classe du domaine «ourbusinessproject.Partnership » et la classe de test associée « PartnershipTest ».
2. Modifiez le contenu du fichier « PartnershipTest .java» de telle sorte qu'il soit identique aux contenus disponibles ici : https://gist.github.com/FranckSilvestre/b16351c08429e0b54486db7484614f31
3. Modifiez le contenu de la classe Partnership pour que les tests de la classe PartnershipTest passent.
   // GIT \\ fix 3.1 classe Partnership

2 - Création du service PartnershipService

1. Créez la classe «ourbusinessproject.PartnershipService » et la classe de test associée « PartnershipServiceIntegrationTest ».
2. Modifiez le contenu de la classe de test pour qu'elle soit identique au contenu disponible ici : https://gist.github.com/FranckSilvestre/0271d99713422874a4c6f42ccc16b0ce
3. Modifiez la classe « PartnershipService » pour que les tests passent.
   // GIT \\ fix 3.2.3 classe PartnershipService

Afin de disposer de quelques Partnership au lancement de l'application nous allons mettre en place la création de ces partnerships en nous appuyant sur les classes Bootstrap et InitialisationService introduites dans la partie 2.

4. Modifiez le contenu des fichiers BootstrapTest.java et PartnershipServiceIntegrationTest.java pour qu'ils soient identiques aux fichiers disponibles ici :
   https://gist.github.com/FranckSilvestre/8d596100c05fb091df0b3730a5f8fcc3
   https://gist.github.com/FranckSilvestre/b4c6636627477bb4967e03a7333c06d0
5. Lancez les tests et constatez que certains tests ne passent plus.
6. Modifiez votre projet de telle sorte que les tests passent de nouveau.
   // GIT \\ fix 3.2.6 initialisation OK

3. Création du RESTful contrôleur

Cet exercice vise la mise en place un Web service permettant d'ajouter et de supprimer des partenariats.

1. Créez la classe PartnershipController. Annotez la classe de l'annotation

Ce(tte) oeuvre est mise à disposition selon les termes de la Licence Creative Commons Paternité - Pas d'Utilisation Commerciale 3.0 France.

org.springframework.web.bind.annotation.RestController.
Créez les deux classes de tests PartnershipControllerTest et PartnershipControllerIntegrationTest. Modifiez les contenus de ces deux classes pour qu'ils correspondent aux contenus disponibles en ligne :
https://gist.github.com/FranckSilvestre/b3c70a792a47d8f950fe0c652157b605
https://gist.github.com/FranckSilvestre/cfbf22696ba1b5690bb4267dd11563d0

2. Lancez les tests et constatez que certains tests ne passent plus.
3. Modifiez votre projet de telle sorte que les tests passent de nouveau.
   // GIT \\ fix 3.3 REST controller OK

4. Dans les arcanes du mapping objet/relationnel...

1. Dans la méthode qui teste l'ajout d'un partnership, pour vérifier que le partnership obtenu est lié à la bonne entreprise, le code utilisé est par le suivant :
   `assertThat(fetchedPartnership.getEnterprise().getId(), is(partnerEnterprise.getId()));`
   Pourquoi ne pas avoir écrit directement
   `assertThat(fetchedPartnership.getEnterprise(), is(partnerEnterprise));?`
   Comment faire pour pouvoir écrire une telle égalité ?
   La réponse se trouve quelque part par là :
   https://access.redhat.com/documentation/en-US/JBoss_Enterprise_Application_Platform/4.3/html/Hibernate_Reference_Guide/Persistent_Classes-Implementing_equals_and_hashCode.html
2. Expliquez en 10 lignes maximum ce que vous avez retenu de votre investigation.
   // GIT \\ fix 3.4 arcanes ORM

5 - Le moteur de recherche sur les Partnerships

Nous allons à présent mettre en place un moteur de recherche sur les Partnerships.
On souhaite mettre en place un Web service REST qui permet de retrouver une liste de partenariats correspondant aux critères de recherche suivants :

- titre du projet : paramètre de requête «project_title »
- nom de l'entreprise : paramètre de requête « enterprise_name ».

Si aucun paramètre n'est fourni dans la requête, on remonte tous les partenariats.
Si un des deux paramètres est présent dans la requête, la recherche filtre suivant l'unique paramètre présent. Par exemple, si seul le titre du projet est spécifié, la requête doit remonter tous les partenariats concernant le projet correspondant.
Enfin le web service doit supporter la spécification de l'ordre de tri des résultats et la spécification de la pagination.
Concevez et mettez en œuvre la solution qui vous semble la plus optimale.
// GIT \\ fix 3.5 search engine

5 - Application cliente pour OurBusinessProject

Créez une application Angular ou Vue.js cliente de l'application OurBusinessProject.
// GIT \\ fix 3.6 rest client

Ce(tte) oeuvre est mise à disposition selon les termes de la Licence Creative Commons Paternité - Pas d'Utilisation Commerciale 3.0 France.