TD n°5

SpringBoot

Notions abordées

•
Back-end :
1. Routes
2. Controllers
3. Models
4. JPA/Hibernate

Application Gestion de scripts (partie 1)

Vous appartenez à la DSI et utilisez régulièrement des scripts permettant d'automatiser des tâches, qu'ils soient créés par vos soins, par vos collègues ou trouvés sur Internet. Vous êtes également amenés à créer/modifier des fichiers de configuration.

Après quelques années d'expérience, vous en avez créé quelques-uns... que vous ne voudriez pas perdre, compte tenu du travail effectué.

Vos collègues du service sont dans le même cas que vous.

On vous confie la mission de créer une application web qui permettra de recenser et de sauvegarder les scripts (de tous genres) créés par les différents intervenants.

Résumé

Outils à utiliser	• Spring-boot
Principales fonctionnalités	◦ Gestion des scripts (Ajout, modification, suppression), ◦ Listage

Règles de gestion (partielles)

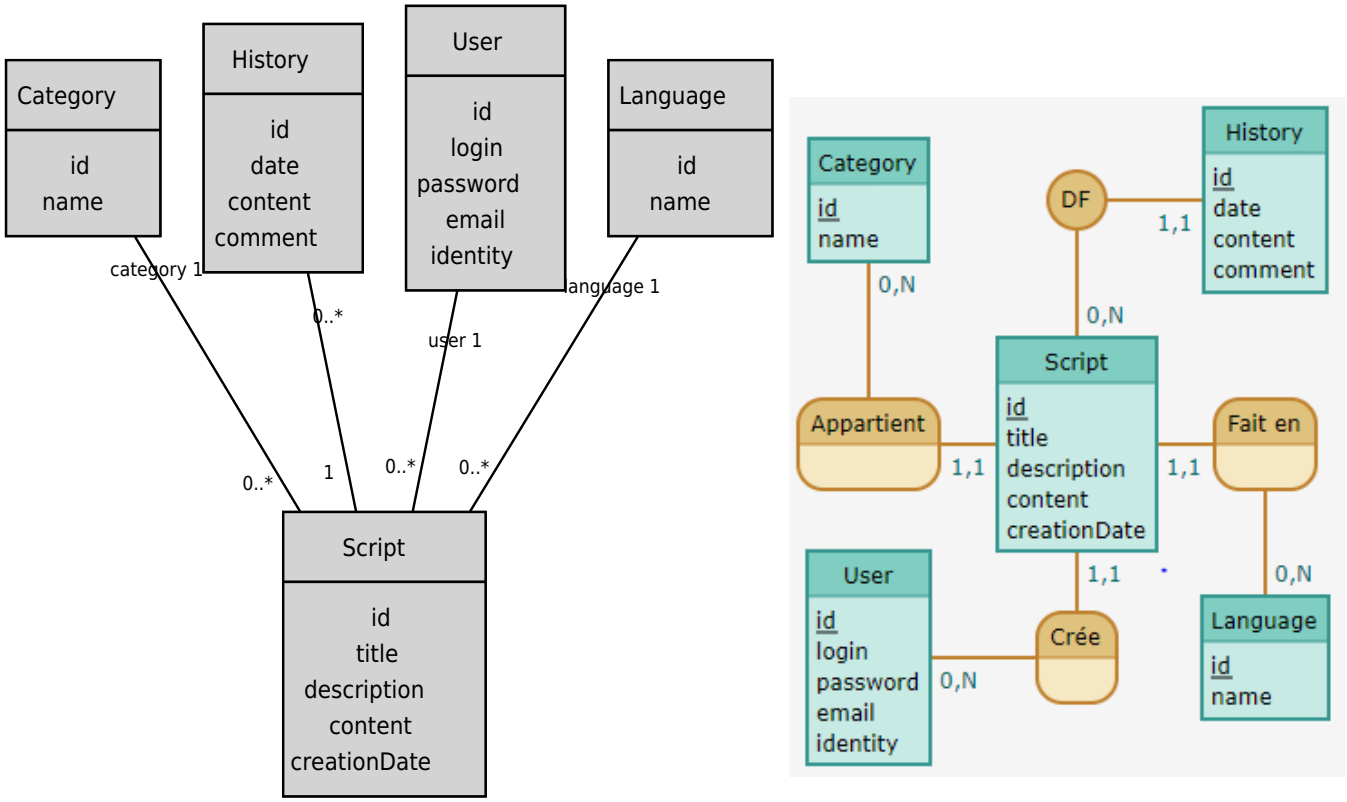
Les utilisateurs peuvent créer un compte, et se connecter à l'application.

Un script :

- est créé par un utilisateur
- appartient à une catégorie (config files, bash scripts...)
- est écrit dans un langage (php, bash, python...)

On conservera un historique des versions de chaque script, accompagné de la date de modification et d'un commentaire.

Diagramme de classes / MEA



CREATED WITH YUML

Fonctionnalités à mettre en place

Entities-Repositories

Créer le projet Créer les classes métier et les repositories nécessaires Configurer l'application (BDD H2)

Connexion

//TODO 1.1

Il s'agit d'une simulation de connexion à l'application :

- Créer des instances d'utilisateur (en dur dans le code) et les enregistrer dans la base de données.

Routes	Description
/login	Met en session l'un des utilisateurs chargé depuis la base de données (il devient l'utilisateur connecté), va ensuite vers l'accueil **/index**
/logout	Déconnecte l'utilisateur connecté (supprime la variable de session) puis va vers l'accueil **/index** et affiche un message

Ajout/modification script

//TODO 1.2

- Ajout/modification d'un script (uniquement ceux qui appartiennent à l'utilisateur)

Routes	Description
/script/new	Affiche le formulaire d'ajout d'un script, dont l'utilisateur connecté sera propriétaire. La page n'est pas accessible si l'utilisateur n'est pas connecté.
/script/submit [POST]	Valide la création ou la modification d'un script puis va vers l'accueil **/index** et affiche un message
/script/{id}	Affiche le formulaire de modification d'un script, dont l'utilisateur connecté doit être propriétaire. La page n'est pas accessible si l'utilisateur n'est pas connecté, ou s'il n'est pas propriétaire du script. Proposer la connexion, ou afficher un message si l'accès n'est pas autorisé au script.

La modification d'un script (content) doit créer une instance de **History** à chaque validation par **/script/submit**.

Liste des scripts / accueil

//TODO 1.3

Routes	Description
/index	Page d'accueil affichant : ◦ La connexion ou la déconnexion ◦ Les infos de connexion (login, email) ◦ La liste des scripts de l'utilisateur ◦ L'accès à l'ajout d'un script ou la modification d'un script existant

From:
http://slamwiki2.kobject.net/ - **SlamWiki 2.1**

Permanent link:
http://slamwiki2.kobject.net/framework-web/spring/td5

Last update: **2019/08/31 14:21**