

Compte rendu projet techno web serveur 2020-2021

Binôme: Serigne Saliou GUEYE && Mouhamed DIA

1) Choix technologiques : Quelle base de données ? Quelle architecture ? Quel code pour le serveur ?

Dans le projet nous avons utilisé:

- **Une base de données H2:**

Nous avons choisi cette base de données car H2 est une base de données très légère (1Mo). Elle va créer les tables et les données uniquement en mémoire vive. Ces données sont perdues une fois l'application fermée. H2 permet aussi de refaire des tests autant de fois que nécessaire en partant d'une base de données propre. L'autre avantage est qu'elle est très simple à mettre en place et complètement auto-configurée.

- **Spark Java pour le serveur Web:**

Nous avons choisi Spark car il présente plusieurs avantages par rapport aux autres technologies big data.

- Il propose un framework complet et unifié pour répondre aux besoins de traitements pour divers jeux de données
- Permet d'écrire rapidement des applications en Java
- Il supporte les requêtes SQL et le streaming de données

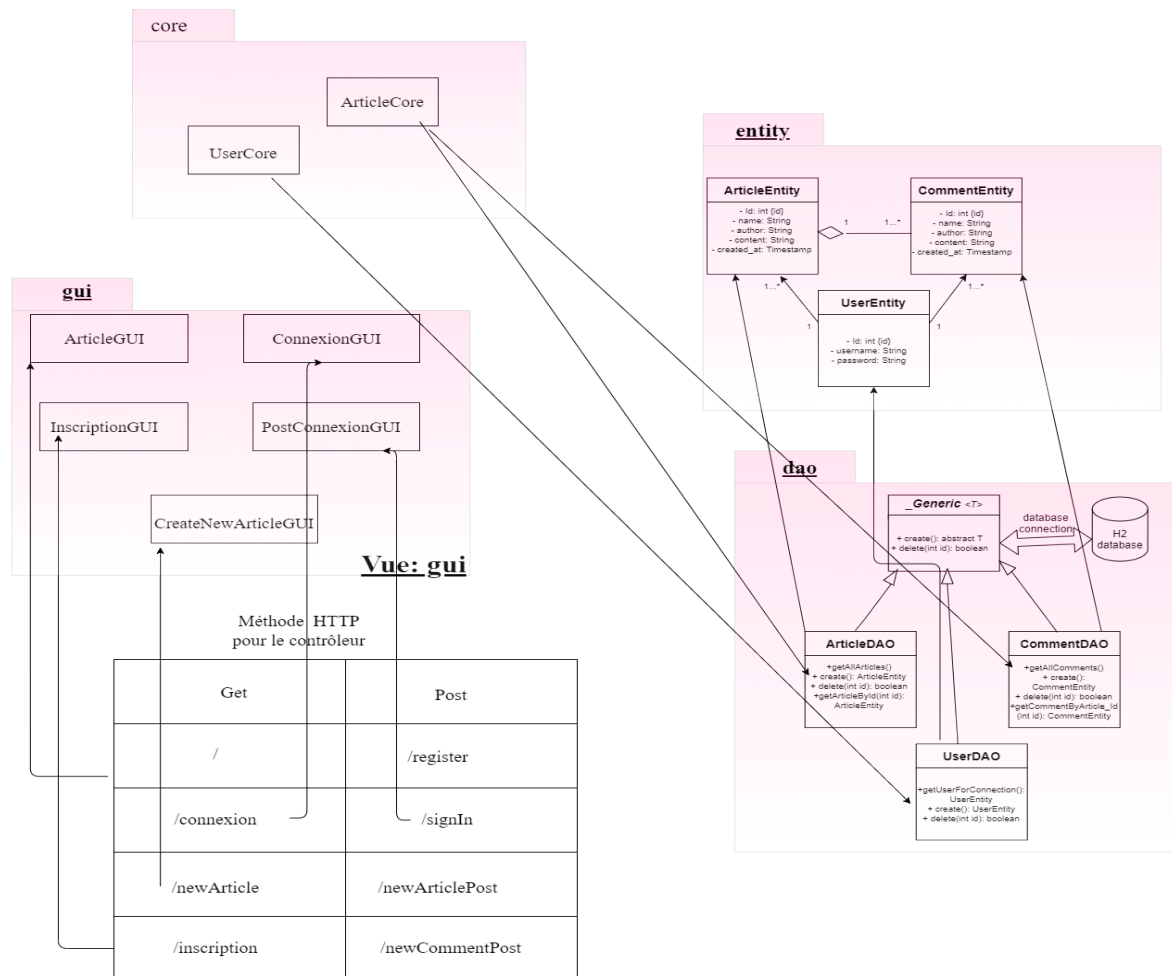
- **FreeMarker pour le templating:**

Nous avons utilisé freemarker car c'est une bibliothèque logicielle déjà écrite en Java et génère automatiquement l'affichage des pages web.

Pour l'architecture, nous avons utilisé l'architecture **MVC(Modèle-Vue-Contrôleur)**.

Ce modèle est choisi en général grâce à sa clarté au niveau de l'architecture qu'il impose. Il nous simplifie la tâche au niveau maintenance ou amélioration. Par exemple, on peut passer d'une base de donnée de type SQL à XML en changeant simplement les traitements d'interaction avec la base. Ce changement par exemple n'affecte pas la vue.

2) Structuration des données dans notre programme: UML et méthodes HTTP



On n'a mis que certaines méthodes HTTP pour le contrôleur

3) Comment compiler le programme?

Nous avons utilisé le compilateur gradle.

Pour compiler le programme, placez-vous déjà dans le dossier de ce projet et exécuter dans un terminal la commande `./gradlew run`

Ceci démarre le serveur de port 8081. A partir d'un navigateur web, mettez: `localhost:8081`.

Une page s'ouvre et c'est la page principale(articles.ftl). Vous pouvez après naviguer vers les menus(connexion, blog, inscription). Vous pouvez également créer un article et faire des commentaires sur des articles une fois connecté.

4) Problèmes rencontrés:

Nous n'avons pas réussi à bien gérer la connexion/déconnexion. Du coup on a opté pour une autre méthode.

Nous avons créé des pages à afficher si on est connecté et d'autres pages à afficher si on est pas connecté. On sait bien que c'est pas la bonne méthode puisque si une personne met par exemple le lien **localhost:8081/newArticle** aura accès à la page de création de nouvel article même si elle n'est pas connecté, alors que ça devrait pas se passer comme ça normalement.

Aussi, nous avons mis tous les utilisateurs admins puisqu'on n'arrivait pas à bien faire la connexion/déconnexion.

4) Les adresses sur lesquelles les ressources seront disponibles

Si on se connecte pas:

- Pour lire les articles et les commentaires: localhost:8081
- Pour s'inscrire: localhost:8081/inscription
- Pour se connecter: localhost:8081/connexion

Si on se connecte:

- Pour lire les articles et les commentaires et créer un commentaire: localhost:8081/articleForConnected
- Pour créer un article : localhost:8081/newArticle
- Pour se déconnecter, cliquez sur le bouton **Déconnexion**