

Objectifs

Le projet a comme finalité le développement d'une application de salon de discussion (chat). Dans cette application, chaque utilisateur peut programmer un canal de discussion (un chat) à une date avec une durée de validité et inviter au moins un autre utilisateur.

On ne souhaite pas sauvegarder les messages (contenu + heure) des canaux de discussions mais ils seront seulement diffusés sur les sessions (canaux connectés) de tous les utilisateurs connectés à ce chat. Une personne qui se connecte tardivement à un canal ne verra pas les messages publiés antérieurement.

Un canal de discussion a un identifiant unique, un titre, une description, une date avec un horaire et une durée de validité.

La gestion des utilisateurs (ajout, suppression, désactivation) se fait via une interface d'administration qui est accessible uniquement aux utilisateurs de type admin. Un utilisateur a un nom, un prénom, une adresse mail, un mot de passe, un champ admin (vrai ou faux)

Deux parties

Ce projet est séparé en deux parties :

Partie 1 : Application web pour les fonctionnalités de base

Elle consiste au développement de la gestion des utilisateurs, la planification, l'édition, l'affichage de la liste des salons de discussion (chats). Elle est composée de :

■ L'interface d'administration de l'application selon l'architecture MVC :

- Modèle : classes métiers (JAVA BEANS) avec une couche accès aux données (JPA et ORM)
- Vue : des pages HTMLs/Theamleaf + CSS + JS (Jquery, Bootstrap...)
- Contrôleur : des contrôleurs Spring (servlet)

■ L'interface utilisateur : planification, édition et affichage de la liste des salons de discussion (chats) et de la page de chat. La structure de l'application est en architecture single page (Composants React + APIs REST). Cette interface exploite le même modèle et donc la même couche accès aux données

Partie 2 : Développement du serveur de chat

Développement du serveur de chat avec le protocole WebSocket en utilisant l'API java websocket. L'interface d'utilisateur de la partie 1 utilise le serveur de chat via la fenêtre de chat. Ce serveur reçoit des messages depuis la fenêtre de chat et il les diffuse sur le canal approprié au niveau des clients connectés.

Livrables et date limite

Date limite : 10/06/2024

Vous devez déposer sur Moodle un seul fichier text (NomsEtudiants.txt) qui contient le lien gitlab de votre dépôt devoir 2. Ce dépôt contient deux livrables :

- Code sources
- Un readme (ex. en markdown) dans votre dépôt git qui doit contenir :

Dans ce document, on utilisera le terme « chat » pour désigner « un salon de discussion »

Ce document peut être mis à jour en cours de semestre pour apporter plus de précision

- Une description de l'architecture de l'application que vous proposez (un schéma avec sa description).
- La conception (diagramme de classes, schéma relationnel, justifier brièvement vos choix)
- Explications sur les interactions entre les différentes technologies : react, spring et web socket
- Résultat eco-index de votre site (plugin greenit) et des pistes d'amélioration