

Projet Spring + Hibernate

M2 TSI – ENSG - Java EE – 2024

Le but du projet est de réaliser **une application web** en utilisant Spring et Hibernate, vous pouvez réaliser le projet proposé ci-dessous, ou éventuellement un autre sujet similaire en difficulté sur un autre thème (me consulter en cas de doute).

1 Sujet proposé

Nous souhaitons réaliser **une application web** de gestion des **participants** à des réunions/colloques. Ces **événements** sont en général organisés par des groupes de personnes, autour d'un thème donné.

Le site doit permettre de gérer les événements et l'inscription des participants. Nous ne nous intéresserons ici qu'à une petite partie de l'application.

Le principe du site est simple: les administrateurs peuvent créer des événements avec un titre, un thème, une date de début, une durée en jours, le nombre maximum de participants, un texte de description, un organisateur, et un type d'événement.

- On doit pouvoir ajouter, lister, modifier, supprimer ces événements dans l'application.

Les administrateurs peuvent ajouter des participants avec un nom, un prénom, une adresse email, une date de naissance, une organisation/entreprise, et des observations éventuelles.

- On doit pouvoir ajouter, lister, modifier, supprimer ces participants dans l'application.
- Ce ne sont pas forcément des participants qui pourront se connecter au site, ils pourront être simplement inscrits aux événements via des formulaires.
- On doit disposer d'un formulaire permettant d'inscrire un participant à un événement (cela peut se passer sur la même page que la création d'un participant).

Pour aller plus loin (et obtenir une meilleure note !)

- Les administrateurs doivent s'authentifier sur le site (on peut supposer qu'il n'y a qu'un seul administrateur) et les pages doivent être accessibles uniquement pour ceux-ci.
- Si lors de l'inscription d'un participant celui-ci a la même adresse email qu'un participant existant, le site proposera de les fusionner.
- Si le nombre de participants maximum est atteint l'application devra interdire l'inscription.
- Au choix, un des bonus rouge proposé dans la section suivante.

Bonus possibles

- Gérer la persistance de la liste des organisateurs d'événements comme les participants et les événements.
- Gérer les inscriptions multiples, par exemple en proposant un import de fichier CSV pour ajouter des participants en masse à partir d'un fichier de tableur bien formé.
- Créer un système d'authentification complet avec auto-inscription des participants authentifiés sur le site.
- Créer un système de file d'attente pour l'inscription de participants à des événements déjà complets.
- Améliorer l'aspect et l'adaptabilité de votre site en utilisant CSS et/ou Javascript (avec Bootstrap par exemple)
- Toute autre amélioration (que vous pouvez me soumettre si vous avez un doute).

Remarques

- Vous travaillerez sur l'**ergonomie** et l'**architecture** de votre application mais aucun point ne sera retiré si celle-ci reste un peu « moche ».
- La réalisation d'une application qui répond parfaitement aux demandes de base (en vert) ne vous donnera qu'une note de 12/20 maximum. La réalisation parfaite de toutes les demandes bleues et d'une demande bonus rouge peut vous permettre d'atteindre la note maximale. La réalisation de tous les bonus, vous assure la gloire !
- Si vous réalisez le projet seul(e) : vous n'avez pas besoin de faire les bonus (en rouge) et vous pouvez retirer une demande bleue.
- Si vous réalisez le projet à trois, vous devez réaliser deux bonus aux choix (au lieu de un).

2 Technologies

Votre site devra utiliser **Spring** et **Hibernate** pour la persistance des données. Pour le contrôleur, vous devez utiliser **SpringMVC**. Vous pouvez utiliser **SpringBoot** pour la configuration de Spring, ainsi que d'autres composants de Spring, notamment pour la sécurité et l'authentification : **Spring Security** (<https://www.baeldung.com/security-spring>)

Votre projet doit utiliser **Maven** comme outil de build.

3 Base de données

Pour la configuration de la bases de données, deux options s'offrent à vous :

1. Utiliser PostgreSQL comme en TP, avec la configuration de bases de données suivante :

```
spring.datasource.url=jdbc:postgresql://localhost:5432/test
spring.datasource.username=test
spring.datasource.password=test
```

2. Alternativement, vous pouvez modifier votre projet pour utiliser H2 qui a l'avantage de stocker la base de données dans un fichier. Vous devez mettre ce fichier dans votre dépôt git. <https://www.baeldung.com/spring-boot-h2-database>

4 Rendu

Le travail est à réaliser **en groupe de 3 maximum**. Vous livrerez votre application **avant le 10 février 2024** inclus avec une documentation d'installation si nécessaire et un petit **compte rendu** PDF de 2 à 5 pages maximum. Le code devra être suffisamment commenté et/ou documenté.

Si vous mettez en place l'authentification, vous devrez me donner les identifiants du ou des utilisateurs sur le site web.

Vous déposerez vos fichiers et collaborerez avec votre binôme sur votre projet à l'aide d'un dépôt **git** distant de votre choix. Votre activité sur git pourra être prise en compte dans votre note finale.

Je vous demande de partager avec moi votre dépôt, **avant le 31 janvier 2024** voici mon identifiant **github** ou **gitlab** : [@cboin-descartes](https://github.com/cboin-descartes)

En cas de questions merci de me contacter par email : clement.boin@esiee.fr