Documentation du Test Java

1.Dependances

Pour ce test j’ai utilisé comme dépendance :

* Spring Jpa et Hibernate comme ORM pour gestion de ma base des données
* Lombok pour me permettre de générer les getters, setters et les constructeurs pour mes classes.
* MySQL comme base des données
* Spring Web pour me faciliter la gestion des endpoints.

2. Couches

Pour le test j’ai utilisé plusieurs couches afin de respecter l’architecture MVC où nous aurons différents packages qui seront :

* Models, Repositories, Services, Interfaces, Dto, config, exceptions et controllers

3. Evénément:

* - Création d’un événement : avant de créer un événement on vérifie si l’évènement n’existe pas au même lieu et à la même date et si la date de fin est bien après la date et l’heure de début de l’élément.
* - Modifier un événement : on vérifie si les informations sont correcte avant de sauvegarder et que la description et lieu n’existe pas dans la base des données.
* Lister tous les participants d’un évènement : on effectue la recherche grâce à l’ID de l’événement pour afficher tous les participants qui sont rattachés à cet ID.

4. Participant

* Inscrire un participant à un évènement : on rattache un participant à un événement grâce à son ID et on vérifie si le participant n’est pas encore inscrit à cet évènement.
* Annuler l’isncription d’un participant à un événement : on retire le participant de la liste de participant de l’événement.
* Lister tous tous les événements auxquels un participant est inscrit : on recupercure le participant grâce à son email pour afficher tous les évènements où il a participé.

5. les Endpoints :

* Pour Creer un participant : “participant/add”
* Pour Afficher les participants par évent : “/participant/evenement/{id}”
* Pour annuler l’inscrption d’un participant à un evenement : "/participant/annuler/{idPart}/{idEvent}"
* Pour créer un évènement : "/evenement/add"
* Pour afficher Tous les évènements : "/evenement/”
* Pour afficher les évènement assister par un participant : /evenement/participant/{id}"
* Pour afficher les évènement par lieu : "/evenement/lieu/{lieu}"
* Pour modifier un évènement : “/evenement/update/{id}"
* Pour annuler un évènement : /evenement/{id}

6. Test fonctionnel

Pour ce test j’ai réalisé 3 tests fonctionnels pour les evenements:

* creationEvenementTest()
* modifierEvenementTest()
* afficherEvenemtParLieu()

Et 3 tests fonctionnels pour participants:

* creerUnParticipant()
* afficherLesparticipantsDUnEvenement()
* annulerParticipantDUnEvenement()

7. Authentification

Malheuresement je n’ai pas pu finir la mise en place de l’authentification avec un token faute de temps pour limiter les accèes entre organisateur et participant;