

Compte-Rendu du projet de J2E

Lucie JEANNEST, Roaa MASRI, Adrienne ZEBAZE

31 Janvier 2022

Contents

1	Présentation du projet	2
1.1	Présentation des classes	2
1.2	Connexion entre classes et base de données	2
1.3	Connexion entre client et contrôleurs	2
1.4	Ajout de l'application WEB	3
2	Notice Utilisateur et Technique	3
3	Schémas	4
4	Fonctionnement de l'application	4
5	Organisation du travail	7
6	Limites	7

1 Présentation du projet

Le but de ce projet était de réaliser une application web de gestion des participants à des réunions/colloques.

1.1 Présentation des classes

Afin de réaliser ce projet, nous avons dû dans un premier temps créer deux classes principales et les relier à une base de données.

Nous avons tout d'abord créé une classe Participant, qui sert à modéliser les individus qui seront amenés à participer aux divers événements.

Chaque individu est enregistré en base grâce à un identifiant unique, un nom, un prénom, une adresse email, une date de naissance, une organisation, et des potentielles observations. Pour chaque individu, on a également une colonne contenant l'identifiant de l'évènement auquel iel est inscrit, qui permet de faire la jointure avec notre seconde table.

Cette table sert à modéliser les différents événements auxquels les participants s'inscrivent.

Chaque événement est enregistré en base via un identifiant unique, un titre, un thème, une date de début, une durée, un nombre maximal de participants, une description, un organisateur, et un type d'évènement.

1.2 Connexion entre classes et base de données

Pour chacune des classes précédemment introduites, on écrit quatre fonctions dans le côté service pour gérer les objets dans la base de données.

La première fonction est *listAll*, qui renvoie tous les éléments de la table sous forme de liste. Ensuite, on a la fonction *save* qui permet d'enregistrer un élément dans la table, et la fonction *get* qui permet de récupérer un élément dans la table grâce à son ID. Enfin, la dernière fonction est *delete*, qui permet de supprimer un élément de la table grâce à son ID.

Le côté service reçoit les informations du côté contrôleur et les envoie à la base de données via le côté repository(DAO).

1.3 Connexion entre client et contrôleurs

Nous avons créé quatre classes pour contrôler le flux d'exécution de l'application. *AppController* est chargé de renvoyer la réponse aux demandes d'index et de page d'accueil. *LoginController* est chargé de renvoyer la réponse aux demandes de page de connexion. *ParticController* est chargé de renvoyer la réponse aux

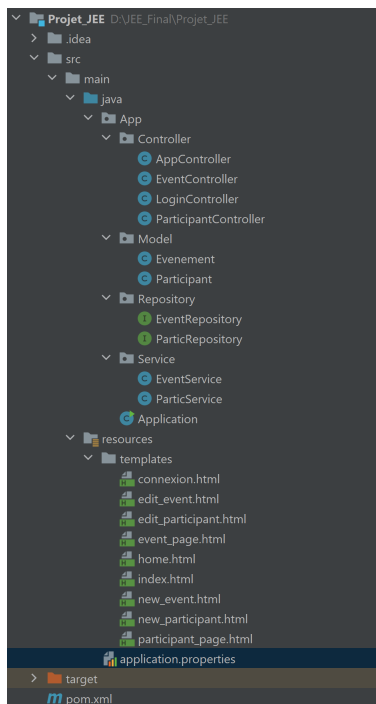
demandes de pages de participant. *EventController* est chargé de renvoyer la réponse aux demandes de pages d'événement.

Le côté contrôleur reçoit les informations du côté client et les envoie au côté service.

1.4 Ajout de l'application WEB

Nous avons enfin créé des formulaires d'inscription qui permettent de remplir la base de données directement depuis l'application WEB, et donc de la rendre plus ergonomique et facile d'utilisation pour un potentiel participant.

2 Notice Utilisateur et Technique



Pour lancer notre application, il faut au préalable avoir une base de données nommée "test", accessible par un utilisateur :

login : "test"

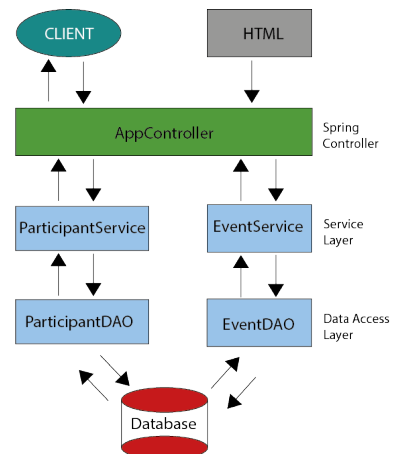
password : "test"

Une fois ce prérequis rempli, il faut ouvrir notre projet depuis IntelliJ, puis lancer (run) le programme "Application" qui se trouve dans le dossier : `Projet_JEE/src/main/java/App/`

Une fois l'application en train de tourner, il suffit d'ouvrir le navigateur à l'adresse : "localhost:8080"

3 Schémas

Comme on peut le voir ci-contre, notre application est conçue selon une architecture bien précise. Le utilisateur a accès à l'application via notre HTML. Toutes les interactions qu'il a depuis l'application passent par le controller. Ensuite, ces informations sont envoyé via la classe de service, puis la classe DAO qui communique avec la base de données.



4 Fonctionnement de l'application



Au lancement de l'application, on arrive sur notre page d'accueil, où on propose de rejoindre la page des évènements ou la page des participants. Dans un premier temps, nous allons nous concentrer sur la partie évènements.

On arrive donc sur une première page qui liste tous les évènements disponibles. A l'ouverture de l'application, il n'y a aucun évènement.

Evénement ID	Titre	Thème	Date de début	Date	Nombre maximum de participants	Description	Organisateur	Type d'événement	Action
--------------	-------	-------	---------------	------	--------------------------------	-------------	--------------	------------------	--------

Créer un nouvel événement

Titre :

Thème :

Date de début :

Date :

Nombre maximum de participants :

Description :

Organisateur :

Type d'événement :

En cliquant sur le bouton "Créer un nouvel évènement", on arrive sur le formulaire ci-joint, qui permet d'ajouter un nouvel évènement.

Si l'on retourne sur la page qui liste les évènements, on retrouve l'évènement qui vient d'être ajouté.

Evénement ID	Titre	Thème	Date de début	Date	Nombre maximum de participants	Description	Organisateur	Type d'événement	Action
1	Dinner with friends	Misc	16-2-2022	2 days	40				Modifier Supprimer
2	Dinner Night	Party	2-2-2022	4 days	30		Dinner with friends		Modifier Supprimer
3	Race	Game	06-09-2022	1 days	20		Race		Modifier Supprimer

ID	Titre	Thème	Date de début	Date	Nombre maximum de participants	Description	Organisateur	Type d'événement
1	Dinner with friends	Misc	16-2-2022	2 days	40		Dinner with friends	
2	Dinner Night	Party	2-2-2022	4 days	30		Dinner with friends	
3	Race	Game	06-09-2022	1 days	20		Race	

On le retrouve également dans la base de données.

On peut également modifier ou supprimer les évènements.

← → ↻ 🌐 localhost:8080/evén/1

Home Page

Modifier l'événement

Titre :

Dance with moon

Thème :

Mois

Date de début :

10-2-2022

Durée :

4 days

Nombre maximum de participants :

40

Description :

Organisateur :

Type d'événement :

Save

Ces modifications se retrouvent dans la base de données.

Elles se retrouvent également sur la page de listing des évènements.

← → ↻ 🌐 localhost:8080/evén

Home Page

Liste des événements

[Cliquez ici pour ajouter un événement](#)

Evénement ID	Titre	Thème	Date de début	Date	Nombre maximum de participants	Description	Organisateur	Type d'événement	Actions
1	Dance with moon	Mois	10-2-2022	2 days	40				Modifier Supprimer
2	Dance Night	Party	2-2-2022	4 days	30				Modifier Supprimer
3	Rim	Genre	06-09-2022	1 days	20				Modifier Supprimer
4	Dance with moon	Mois	10-2-2022	2 days	40				Modifier Supprimer

← → ↻ 🌐 localhost:8080/evén

Home Page

Liste des événements

[Cliquez ici pour ajouter un événement](#)

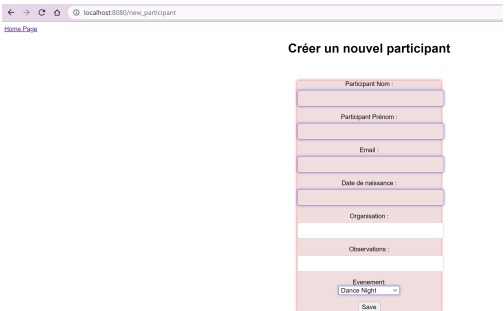
Evénement ID	Titre	Thème	Date de début	Date	Nombre maximum de participants	Description	Organisateur	Type d'événement	Actions
2	Dance Night	Party	2-2-2022	4 days	30				Modifier Supprimer
3	Rim	Genre	06-09-2022	1 days	20				Modifier Supprimer
4	Dance with moon	Mois	10-2-2022	2 days	40				Modifier Supprimer

On remarque qu'après suppression, un évènement a disparu.

Maintenant, concentrons-nous sur la partie Participants.
On arrive sur une page qui liste tous les participants enregistrés. Pour l’instant, cette liste est vide.



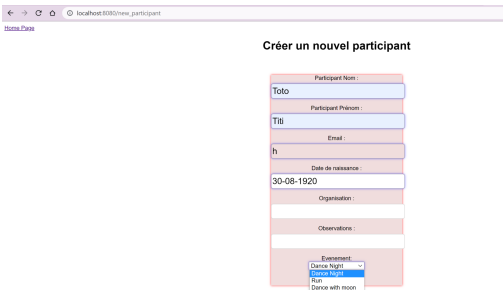
Participant ID	Participant Nom	Participant Prénom	Email	Date de naissance	Organisation	Observations	Evénement	Actions
----------------	-----------------	--------------------	-------	-------------------	--------------	--------------	-----------	---------



En cliquant sur le bouton ”Ajouter un participant”, on arrive sur le formulaire ci-joint, qui permet d’ajouter un nouvel évènement.



Le formulaire vérifie que l’email est valide, en regardant s’il contient bien un ”@”.



On retrouve les évènements créés précédemment dans la liste déroulante permettant de choisir l’évènement.

A la création du participant, il est ajouté dans la base de données.



id	email	email_verified	name	name_verified	observations	observations_verified	organisation	organisation_verified	premium	premium_verified	is_event
1	ti@toto.com	True	Titi	True							False
2	ti@toto.com	True	Titi	True							False



Participant ID	Participant Nom	Participant Prénom	Email	Date de naissance	Organisation	Observations	Evénement	Actions
4	Toto	Titi	ti@toto.com	30-08-1920			Dance with meow	Supprimer
5	Momo	Mami	mami@toto.com	30-08-1920			Dance with meow	Supprimer

On le retrouve également sur la page de listing des participants.

5 Organisation du travail

Nous avons travaillé sur un repository git, sur lequel on poussait notre travail régulièrement. Nous faisons également des réunions régulièrement afin de discuter de notre avancée sur le projet. Si jamais l'une d'entre nous venait à être bloquée sur un point en particulier, nous profitons des pauses entre les cours pour en discuter et résoudre le problème.

Les mission:

1. La partie Modèle : Roaa MASRI
2. La partie Repository(DAO) :
 - Participant : Adrienne ZEBAZE
 - Évènement : Roaa MASRI
3. La partie Service :
 - Participant : Adrienne ZEBAZE
 - Évènement : Roaa MASRI
4. La partie Contrôleur :
 - App : Roaa MASRI
 - Login : Adrienne ZEBAZE
 - Participant : Adrienne ZEBAZE et Roaa MASRI
 - Évènement : Roaa MASRI
5. Les HTML : Roaa MASRI et Adrienne ZEBAZE
6. Le rapport : Lucie JEANNEST et Roaa MASRI

6 Limites

Lors de l'inscription d'un participant à un évènement, l'application ne vérifie pas si le nombre maximal de participants à cet évènement est atteint. On pourrait donc potentiellement se retrouver avec plus d'inscrits à un évènement que de places disponibles.

Toujours au moment de l'inscription, notre application considère que l'email est valide à partir du moment où il contient un "@" . Le participant pourrait donc s'inscrire avec une adresse mail inexistante.

Nous n'avons pas créé la page de connexion correctement donc notre application fonctionne avec n'importe quel nom d'utilisateur et mot de passe.