

PPE09 – Gestion des colloques (Java)
Interface graphique
BTS2 SLAM – 2018/2019

1 Documents et liens

- Inspirez-vous des travaux réalisés en cours et TP pour mener à bien ce projet.
- Les fichiers CSV avec Java :
<http://thierry-leriche-dessirier.developpez.com/tutoriels/java/csv-avec-java/>
- Tutoriaux de prise en main du système automatique de création d'interfaces graphiques :
<https://netbeans.org/kb/docs/java/gui-functionality.html>
<https://netbeans.org/kb/docs/java/quickstart-gui.html>

2 Contexte

Le site web de réservation des séjours a été mis en place sur le site du Jura. Le conseil d'administration du CVVEN souhaite désormais étendre ce projet à un programme de gestion des colloques.

En dehors des périodes de vacances scolaires, le site du jura peut accueillir des colloques, congrès, séminaires, ou des groupes. Les salles de réunions sont équipées de matériel informatique, et de vidéoprojecteurs. L'hébergement et la restauration peuvent se faire sur place.

3 Votre mission principale

Dans le cadre du développement sous Java, vous êtes chargés de réaliser un applicatif pour la gestion des événements. Un événement peut être un colloque, un séminaire ou un congrès.

Le principal objectif de votre mission est la création d'une interface graphique sous Java, en utilisant les outils Java appropriés (Swing, SWT, awt, JavaFX ...).

L'exploitation de la base de données relationnelle à partir d'une application Java est réalisée par l'autre binôme (PPE08) en utilisant une couche de persistance utilisant le Framework Hibernate et l'outil de gestion des dépendances Maven. Cependant, votre travail doit être réalisé en étroite collaboration.

L'applicatif doit répondre aux besoins suivants :

- 1. Création d'une fenêtre d'authentification (Login, mot de passe) ;**
- 2. Saisie des nouveaux événements et des différents participants (Prévoir deux interfaces graphiques différentes) ;**
- 3. Affichage d'informations de synthèse sur l'ensemble des événements (date, durée, thème, nombre de participants, ...).**
4. L'application doit suivre et garder trace de tous les événements (y compris les anciens qui seront archivés).
5. Les participants peuvent être inscrits par les gestionnaires manuellement via les interfaces Java, ou via l'importation d'un fichier (CSV séparé par des virgules par exemple). L'inscription ne sera pas possible sur les événements archivés.
6. Lors de l'inscription, le formulaire doit permettre de choisir au moins un événement pour le participant.

Pour une meilleure organisation de la navigabilité entre les interfaces créées, pensez à utiliser les menus en Java (Jmenu avec Swing).

Bonus : Etat statistique sur les différents événements par type, par thème et par période éventuellement.

(Utilisation du composant graphique Jtable avec Swing)

4 Base de données de l'application

Les colloques sont en général organisés par des groupes de personnes, autour d'un thème donné. Le nombre de participants est tributaire de la capacité de la salle réservée.

Un extrait du modèle de données (A compléter)

```
Participant (num_pers, nom, prenom, email, date_naiss, organisation, observations)
Evenement (num_even, intitule, theme, date_debut, duree, nb_part_max, description, org
anisateur, type_even)
...
```

Une personne peut être inscrite à un ou plusieurs événements. Pour tout événement, une salle sera réservée avec l'ensemble des équipements qui s'y trouvent.

La gestion de l'hébergement, de la restauration et du matériel n'est pas exigée ici.

5 Contraintes

Environnement

Le langage retenu pour ce projet est le **Java**, avec une **interface graphique**. L'environnement de développement et la bibliothèque graphique (Swing, JavaFX...) sont laissés à votre libre appréciation.

SGBD

Le système de gestion de base de données est sous **PostgreSQL**.

Codage

L'accès à la base de données devra être obligatoirement réalisé à l'aide d'une **couche de persistance** (Framework Hibernate). La gestion des dépendances est assurée à l'aide de l'outil **Maven**.

Des tests unitaires devront être écrits avec **Junit**. Le code source devra être publié sur github.com dans l'organisation <http://github.com/ppe-slam>

Documentation

L'application devra être livrée avec une documentation (copies d'écrans, fonctionnement...). Le code devra également être bien documenté avec **Javadoc**.

Organisation du travail

La répartition du travail est laissée à votre libre choix, vous saisissez les tâches à effectuer dans l'outil Kanboard (votre chef de projet devra créer une nouvelle « **swimlane** »).