Projet JEE – Site internet CCI

Création d’un site internet pour les anciens étudiants de la filière CCI

15/04/2013

Université de Tours François Rabelais

KABA Mamady

Sommaire

[I- Description du Projet 3](#_Toc353834221)

[II- Réalisation du projet 3](#_Toc353834222)

[A- Description des choix technologiques 3](#_Toc353834223)

[1- Base de données 3](#_Toc353834224)

[2- Serveur d’application 3](#_Toc353834225)

[3- Environnement développement 3](#_Toc353834226)

[4- Outils de modélisation 4](#_Toc353834227)

[5- Gestion des projets 4](#_Toc353834228)

[B- Architecture de l’application 4](#_Toc353834229)

[a- Schéma de la base de données 5](#_Toc353834230)

[b- Diagramme de classes 6](#_Toc353834231)

[1. Description des classes 7](#_Toc353834232)

[1.1. Conteneur EJB 7](#_Toc353834233)

[Le package metier 7](#_Toc353834234)

[Le package beans 7](#_Toc353834235)

[1.2. Conteneur WAR 7](#_Toc353834236)

[Le package servlet 7](#_Toc353834237)

[Les pages JSP 7](#_Toc353834238)

[6- A savoir 8](#_Toc353834239)

[7- Lancement 8](#_Toc353834240)

# Description du Projet

Il s’agissait de mettre en place une application WEB/JEE pour la gestion de la formation CCI Tours. Cette application doit pouvoir :

* Enregistrer tout élève ayant suivi le parcours CCI à l’université de Tours
* Renseigner l’évolution des carrières de tous les anciens étudiants déjà enregistrés dans la base de données de l’application
* Fournir des informations sur les différents évènements (passés ou futurs) s’inscrivant dans le cadre des actions du CCI

# Réalisation du projet

Le projet a été réalisé en Java/JEE

## Description des choix technologiques

## Base de données

L'application est fonctionnelle aussi bien avec une base de données **Java Derby** qu'avec une base de données **MySQL**. Il suffit cependant de procéder aux configurations suivantes :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Base de données | Login | Mot de passe | Ressource JNDI |
| Java Derby | ccidb | kaba | kaba | ProjetCCI\_JNDI |
| MySQL | ccidb | root |  | jdbc/cciJNDI |

## Serveur d’application

***Glassfish 3.1:***

* Pour sa faciliter d'utilisation à travers la génération automatique des classes, des méthodes
* Pour la présence d’une interface d’administration très intuitive pour la gestion des fichiers de configuration
* Pour sa faciliter de créer facilement les fichiers de configuration au niveau du serveur

## Environnement développement

***Netbeans 7.3***

Outil assez complet pour faire du développement JEE. Il contient par défaut plein de serveurs d'applications s’intégrant dans le développement d’application WEB/JEE. Il offre également plein de fonctionnalités félicitant le développement.

## Outils de modélisation

***Mysql Workbench*** pour la conception de la base de données

***StarUML*** pour la modélisation UML

***Umbrello*** pour la génération du fichier de diagramme des classes

## Gestion des projets

***Subversion*** pour faciliter le développement d’application dans un groupe de projet, et pour la gestion des versions de l’application

## Architecture de l’application

SGBD



WEB

JSP

JSTL

Servlet

JPA

DAO

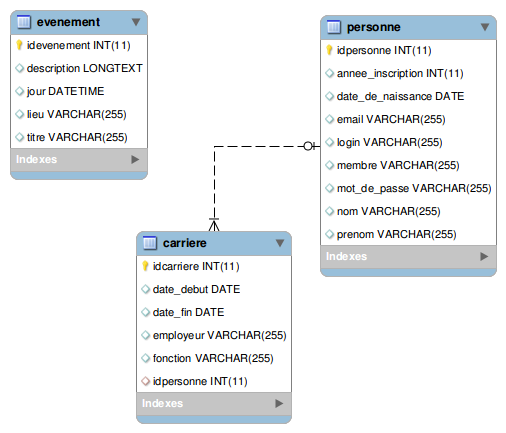
JTA

Conteneur EJB

Conteneur WAR

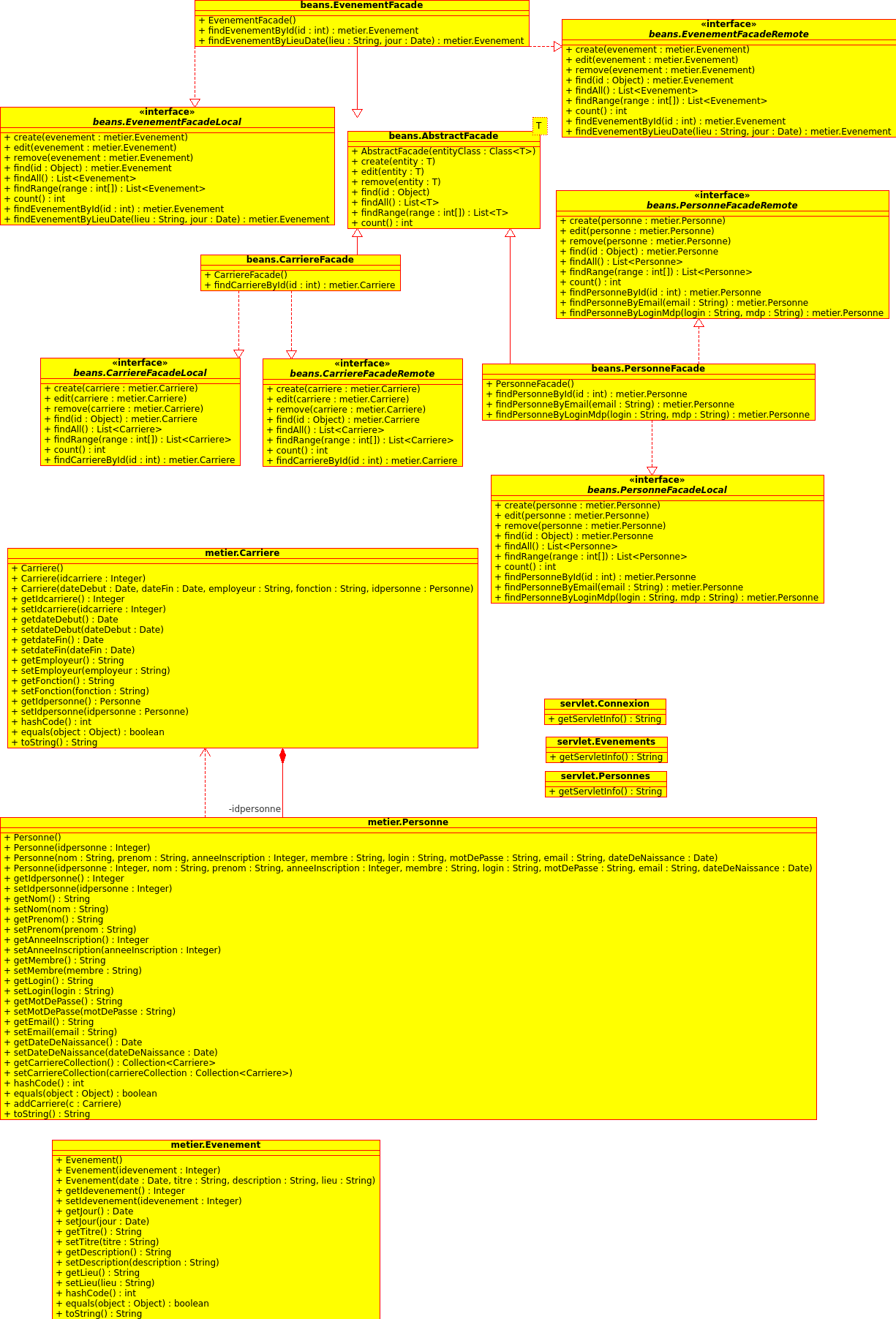
*Architecture JEE de l’application*

### Schéma de la base de données



*Schéma de la base de données*

### Diagramme de classes



### Description des classes

### Conteneur EJB

Le conteneur EJB contient deux packages :

### Le package metier

Contient les classes d’entités

* *Personne*
* *Carriere*
* *Evenement*

### Le package beans

Contient les interfaces DAO et leurs classes d’implémentation

* Les interfaces ***PersonneFacadeLocal et PersonneFacadeRemote*** sont implémentées par la classe PersonneFacade
* Les interfaces ***EvenementFacadeLocal et EvenementFacadeRemote*** sont implémentées par la classe EvenementFacade
* Les interfaces CarriereFacadeLocal et CarriereFacadeRemote sont implémentées par la classe CarriereFacade

### Conteneur WAR

Le conteneur WAR contient deux packages

### Le package servlet

Gère les relations entre couche présentation de l’application et la couche métier. Il contient les classes suivantes :

* Personnes
* Evenements
* Connexion

Ces classes utilisent des entités EJB pour stocker ou récupérer les données dans la base de données.

### Les pages JSP

**entete.jsp :** Est incluse dans toutes les pages jsp en entête

**pied.jsp :** Est dans toutes les pages jsp en pied de page

**personnes.jsp** : Affiches dans un tableau toutes les personnes enregistrées dans la base de données. Seul l’administrateur peut effectuer des opérations d’ajout de suppression et de modification.

Le tableau ne contient aucune information sur l’administrateur

**profil.jsp :** Affiche le profil de l’utilisateur connecté. A partir de cette page, l’utilisateur peut modifier les informations le concernant.

**enregistrement\_personne.jsp :** Affiche le formulaire d’inscription ou de modification d’une personne et de sa carrière.

**enregistrement\_ok.jsp :** Affiche le message de réussite après un enregistrement

**erreur.jsp :** Affiche le message d’erreur en cas d’échec lors d’un enregistrement

**gestion\_evenements.jsp :** Affiche les évènements créés. Seul l’administrateur à le droit de d’ajouter, de modifier ou de supprimer un évènement

**detail\_evenement .jsp :** Affiche le détail complet concernant un évènement

**enregistrement\_evenement.jsp :** Contient le formulaire d’enregistrement d’un nouvel évènement

# A savoir

Après le lancement, l’application créé automatiquement un administrateur

* Login : admin
* Mot de passe : admin

Il est possible de créer des utilisateurs tests et des évènements test

# Lancement

Pour lancer l’application il faut s’assurer d’avoir créé la base de données comme décrit ci-dessus, en suite déployer le projet en faisant un clic droit, puis exécuter (avec un clic droit).