

Synthèse du projet : capiprof

Sommaire

I. Présentation.....	3
II. Description des informations d'un intervenant	3
III. Base de données	3
IV. Technologies utilisées	4
V. Génération du fichier CSV	4
VI. Importation des données.....	6
VII. Interface Web	8
VIII. Pistes pour le futur.....	9

ATTENTION : En cas de modifications sur le projet :

1. Dans C:\Dev\workspace, sur le dossier plbsi, clique-droit : UPDATE
2. Dans MySQL Workbench, supprimer les tables Intervenant et IntervenantFormation avec les requêtes DROP TABLE. Vérifier la suppression des tables.
3. Dans Eclipse, dans Servers, faire Star sur JBoss. Vérifier la recréation des tables Intervenant et IntervenantFormation dans MySQL Workbench.
4. Création/modifications du fichier CSV.
5. Dans MySQL Workbench, importer les données du fichier CSV grâce aux requêtes d'importations. (Vérifier la bonne importation des données).
6. Dans Eclipse, relancer JBoss dans Servers.
7. Vérifier que tout marche dans l'application en local.
8. Une fois toutes les modifications terminées, dans C:\Dev\workspace, sur le dossier plbsi, clique-droit : COMMIT. Commiter seulement les nouveaux fichiers. ATTENTION : vérifier les fichiers à commiter.
9. Voir avec David pour la mise en ligne sur le site.

Le but de l'application est de proposer une interface Web permettant d'accéder à toutes les informations des intervenants. On doit pouvoir modifier ces informations, mais aussi pouvoir créer ou supprimer des intervenants.

I. Présentation

Pour l'instant les informations d'un intervenant sont accessibles via un fichier EXCEL alimenté par le progiciel de la société Gescof. Ce fichier EXCEL n'est pas optimal et la recherche des informations n'est pas pratique. C'est pourquoi on veut importer ces données dans une base de données, pour ensuite pouvoir faciliter l'utilisation de ces informations via une interface Web.

Un projet qui s'occupe des formations existe déjà et est déjà accessible sur le site : plbsi. On va donc regrouper ces deux projets dans le but de pouvoir relier plus tard les intervenants aux formations. On va donc insérer les données dans la base de données actuelle : plbconsult.sql qui gère les formations.

II. Description des informations d'un intervenant

Le fichier EXCEL Liec.xls contient des informations sur les intervenants. Celles à garder sont:

L'identifiant, le nom, le prénom, l'adresse, le rang, le numéro de téléphone, le numéro de portable, l'email, l'origine de la demande, le tarif, les centres, la maîtrise de l'anglais ou non et les observations.

Les informations à rajouter sont :

Le statut, la délégation, les informations sur le tarif, la date de mise à jour et enfin les certifications.

J'ai rajouté en plus l'attribut statutAutre, car un certain nombre de statut sont prédéfinis (indépendant, salarié, auto entrepreneur) et si c'est autre chose il pourra donc préciser la nature du statut.

Donc en tout un intervenant possède 19 attributs qui seront donc dans la table Intervenant.

III. Base de données

La base de données sur laquelle on travaille est plbconsult.sql. Elle est déjà composée d'une table Intervenant mais qui ne correspond pas aux attributs demandés. J'ai donc créé une nouvelle table Intervenant qui remplace celle déjà existante. Elle contient donc les 19 attributs répertoriés précédemment. Ensuite j'ai du créer la table IntervenantFormation qui relie les intervenants aux formations. Elle contient les attributs suivant : le support, le tarif et les commentaires.

IV. Technologies utilisées

Les technologies utilisées sont :

- Java qui permet de créer les différentes classes avec leurs différents attributs.
- JBoss qui est un serveur d'applications J2EE Libre entièrement écrit en Java.
- Seam qui est un serveur d'application pour Java EE5, et c'est une application Web développée par JBoss. Elle permet donc de développer l'application Web plbsi. Les pages Web sont créées en xhtml.

Et c'est grâce à hibernate que la relation est effectuée entre les tables de la base de données plbconsult et les classes Java. Hibernate est un framework open source gérant la persistance des objets en base de données relationnelle.

Logiciels utilisés :

- Eclipse qui est un environnement de développement qui permet de développer les classes Java et les fichiers Web, ainsi que de faire le lien entre ces classes et la base de données. Pour accéder à Eclipse : C:\Dev\eclipse et lancer eclipse.exe.

Pour relier la base de données à ces classes Java, dans Eclipse il faut aller dans "Servers" et démarrer "JBoss 7.1 runtime Server", et les changements effectués dans les classes Java seront alors présents dans la base de données.

- MySQL Workbench qui est un logiciel de gestion et d'administration de bases de données MySQL. C'est dans ce logiciel que les requêtes sont effectuées, notamment celles pour l'importation des données dans la base. Lors de l'ouverture du logiciel, il faut sélectionner "Local instance MySQL56", s'ouvre alors une fenêtre "SQL Editor" et c'est ici qu'on pourra effectuer les divers requêtes. Pour accéder à MySQL Workbench : C:\Dev\autres\MySQL Workbench 5.2.47 CE et lancer MySQLWorkbench.exe.

V. Génération du fichier CSV

Avant d'importer les données dans la base de données, il faut au préalable enregistrer le fichier EXCEL LIEC.xls en CSV et le trier de sorte que les colonnes de ce fichier soient les mêmes et dans le même ordre que les attributs de la table Intervenant comme dans le tableau ci-dessous.

NumIntervenent	Adresse complète	Champs Spe5	Champ SPe4	EmailIntervenent	NomIntervenent	NumPortableIntervenent	NumTelIntervenent	Observations	NumComptableIntervenent	PrenomIntervenent	CodeInternetIntervenent	CoutJournalierFC
Id	Adresse	Anglais	Centres	Mail	Nom	Num Port	Num Tel	Observations	Origine	Prenom	Rang	Tarif

Les colonnes à supprimer sont :

"NumTitre", "CoutHoraireInsertion", "CoutHoraireFC", "CoutJournalierInsertion", "FraisDepMiJournee", "FraisDeplacement", "SalarieASFO", "ContratTravail", "NumOrganisation", "DateNaissance", "LieuNaissance", "Nationalité", "NumSecu", "NbJoursPresence", "FraisHotel", "FraisLoc", "MontantCT", "CoutHoraireFD", "CoutJournalierFD", "Actif", "AdresseIntervenant", "CPIntervenant", "VilleIntervenant", "Pays" (car le champ Adresse complète renseigne sur ces quatre derniers champs), "CoefCharge", "NbHeureAnnee", "AbrPays", "ChampsSpec1", "ChampSpe2", "ChampSpe3", "PhotoAnimateur", "NumFaxAnimateur", "Adresse2Intervenant", "TypeFormateur", "CouleurIntervenant", "Profession", "ServiceAdresseItv", "CoutStagiaire", "NumGroupeUtil", "IntervenantFictif", "ModifPlanningFacett", "Fichier CV", "DateModifIntervenant", "CodeIntervenant", "Compétences".

Ensuite les deux colonnes "CoutJournalierFC" et "Tarif" renseignent toutes les deux sur le tarif de l'intervenant. Cependant, on garde la colonne "CoutJournalierFC" car elle est correctement rempli ce qui n'est pas le cas de l'autre, une fois l'autre colonne supprimée, on renomme "CoutJournalierFC" par Tarif.

De même pour les colonnes "NumComptableIntervenant" et "Origine/statut" qui renseignent toutes les deux sur l'origine de la demande. En revanche, la colonne correctement rempli est "NumComptableIntervenant" donc on garde celle-là et on supprime l'autre. De plus, on la renomme par Origine.

Ensuite, il faut rajouter les colonnes qui ont été demandées (infoTarif, statut, délégation, certifications, dateMisAJour) et bien entendu conserver les attributs nécessaires et qui existent déjà. Le champ "codeInternetIntervenant" dans le fichier CSV correspond au rang, le champ "ChampsSpec1" correspond à l'anglais.

Pour la colonne anglais "ChampSpe5", il faut remplacer les oui/non par 1/0 pour que ce soit adapté à la base de données.

Dans le fichier CSV, pour les colonnes des numéros de téléphone et de portable, il faut sélectionner toute la colonne, et modifier le format de cellule, aller dans la catégorie "spécial" et choisir le type : "Numéro de téléphone", pour que ce soit correctement rempli dans la base de données, sinon le 0 du départ est supprimé.

A la suite de la mise en forme du CSV il faut rajouter les attributs supplémentaires dans l'ordre alphabétiques :

Certifications, DateMisAJour, delegation, infoTarif, Statut, StatutAutre

Pour plus de sureté lors de l'importation des données, il faut mieux supprimer la première colonne qui renseigne sur les champs dans le CSV.

VI. Importation des données

L'importation des données du fichier CSV se fait grâce à une requête SQL.

Pour importer les données dans la base de données, on doit réaliser ce script SQL :

```
SET SESSION collation_database=latin1_swedish_ci;
```

```
SET SESSION character_set_database=latin1;
```

```
LOAD DATA LOCAL INFILE 'C:/Dev/LIEC.csv' INTO TABLE Intervenant FIELDS TERMINATED  
BY ';' ENCLOSED BY '"' LINES TERMINATED BY '\n';
```

```
UPDATE Intervenant SET dateMisAJour=null;
```

```
UPDATE Intervenant SET rang=50 WHERE rang=0;
```

Les deux SET SESSION permettent de garder les accents lors du transfert du fichier CSV vers la base de données. Par exemple, sans ces deux requêtes, une fois l'importation faite dans la base de données, pour les observations, dès qu'on était censé rencontrer un accent la suite du texte était tronquée.

C'est la requête LOAD DATA LOCAL INFILE qui permet d'importer les données du fichier CSV dans la base de données. Il faut bien préciser le chemin d'accès du fichier CSV dans la requête.

Ensuite, étant donné que la date de mise à jour est un nouvel attribut de la table Intervenant, il n'y a alors aucun champ rempli dans le CSV concernant cet attribut, et lorsqu'on importe les données, il met automatiquement la date à "00-00-0000 00:00", on remet donc la date à null grâce à la requête "UPDATE Intervenant SET dateMisAJour=null;". De plus, si on ne réalise pas cette commande, cela pose un problème par la suite lorsqu'on veut afficher les intervenants dans l'interface Web.

Enfin, la dernière requête "UPDATE Intervenant SET rang=50 WHERE rang=0;" est nécessaire pour tous ceux qui n'ont pas de rang au départ dans le CSV et qui ont alors un rang égal à 0 dans la base de données. On les met donc tous à 50 grâce à cette requête.

Pour chaque génération du CSV vers la base de données :

A chaque régénération d'un nouveau fichier CSV il faut relancer le script complet : "script.sql" (C:\Dev\workspace\plbsi\data).

Tout d'abord, dans MySQL Workbench, il faut donc supprimer les tables Intervenant et IntervenantFormation dans le logiciel MySQL Workbench grâce aux requêtes :

```
DROP TABLE IntervenantFormation;
```

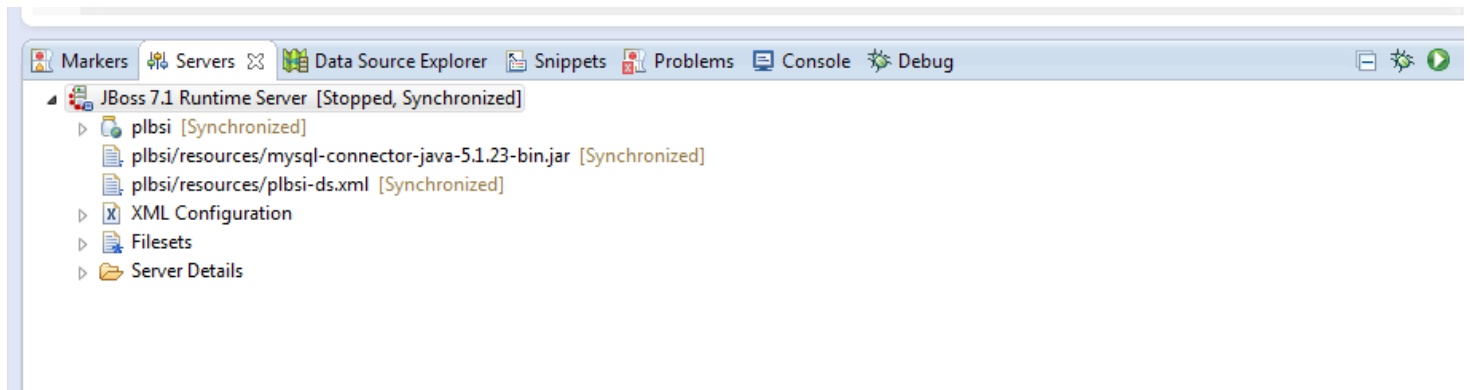
```
DROP TABLE Intervenant;
```

Pour exécuter les requêtes, il faut appuyer sur le petit éclair en haut.

(Sinon quand on régénère le fichier CSV les données seront ajoutées à celles déjà existantes).

Sur la table plbconsult, il faut faire un "Refresh All" pour vérifier que les tables sont bien supprimées (dans la fenêtre "Object Browser" à gauche).

Ensuite dans Eclipse, il faut démarrer JBoss (comme vu précédemment) pour que les tables soient à nouveau créées dans la base de données.



Si tout est bon, on obtient le résultat suivant :

```
(MSC service thread 1-2) JBAS015874: JBoss AS 7.1.1.Final "Brontes" started in 16686ms - Started 257 o
erver] (DeploymentScanner-threads - 2) JBAS018559: Deployed "plbsi.war"
erver] (DeploymentScanner-threads - 2) JBAS018559: Deployed "plbsi-ds.xml"
erver] (DeploymentScanner-threads - 2) JBAS018559: Deployed "mysql-connector-java-5.1.23-bin.jar"
```

Dans MySQL Workbench, on vérifie alors que ces tables soient créées grâce à un "Refresh All".

On peut alors importer le nouveau fichier CSV grâce aux requêtes d'import dans la base de données (cf script au début du chapitre). Les données sont alors bien importées, on peut le vérifier grâce à la requête :

SELECT * FROM Intervenant LIMIT 0, 2000;

Dans MySQL Workbench il faut mettre "LIMIT 0, 2000;" car par défaut c'est "LIMIT 0, 1000" et on ne voit pas donc toutes les données importées (puisque'il y a plus de 1000 intervenants).

Pour tester dans l'application que tout est bien importé, il faut redémarrer JBoss et aller sur <http://localhost:8080/plbsi/> pour tester en local. Il faut ensuite Lancer l'indexation dans cette application (l'indexation permet de faire des recherches).

Afin que David puisse mettre en ligne les modifications, il faut aller dans C:\Dev\workspace et sur le dossier plbsi faire un clic-droit SVN Commit. Comme sur l'image suivante, il faut commiter seulement les nouveaux fichiers et SURTOUT pas les fichiers : persistence.xml et tous les fichiers commençant par : WebContent/WEB-INF/dev...

Commit to:
<http://dev.plb.fr/svn/plbsi/>

Message:

Recent messages

Changes made (double-click on file for diff):

Check: **All** None Non-versioned Versioned Added Deleted Modified Files Directories

Path	Extension	Status	Property status	Lock
<input checked="" type="checkbox"/> data/script.sql	.sql	mo...		
<input type="checkbox"/> src/main/META-INF/persistence.xml	.xml	mo...	normal	
<input type="checkbox"/> hibernate-console.properties	.prope...	non...		
<input type="checkbox"/> plbsi.launch	.launch	non...		
<input type="checkbox"/> src.rar	.rar	non...		
<input type="checkbox"/> WebContent/WEB-INF/dev/com		non...		
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb		non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si		non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/converter		non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/converter/SessionsConverter.class	.class	non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/converter/URLConverter.class	.class	non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/dto		non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/dto/MonthSessionDto.class	.class	non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/dto/SessionDto.class	.class	non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/manager		non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/manager/ApplicationManager.class	.class	non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/manager/Authenticator.class	.class	non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/manager/FormationManager.class	.class	non...	non-versioned	
<input type="checkbox"/> WebContent/WEB-INF/dev/com/plb/si/manager/HomeManager.class	.class	non...	non-versioned	

VII. Interface Web

L'application est disponible sur l'URL suivante : dev.plb.fr:8081/plbsi

Pour se connecter au site, il y a quatre utilisateurs (Pierre, Julie, Laurent, Aurore) dont voici les identifiants :

Login : première lettre du prénom suivi du nom de famille pour chaque utilisateur, tout écrit en minuscule et sans espace.

Mot de passe : "secret" pour tout le monde.

Sur cette application, on peut donc consulter l'ensemble des informations des intervenants.

Concernant l'affichage des différents intervenants, les informations principales disponibles sont : le nom, le prénom, l'email, le numéro de téléphone portable, le rang, le tarif, les observations et enfin la dernière date de mise à jour. On peut trier cet affichage par nom, prénom, rang, tarif, et la date de mise à jour dans l'ordre ascendant ou descendant.

Les intervenants dont le rang est supérieur à 90 (blacklist) sont affichés en rouge.

On peut également effectuer des recherches sur les intervenants : il suffit de saisir un mot-clé et l'affichage s'effectuera en fonction de celui-ci.

Concernant la partie Compétences de l'affichage, celle-ci concerne la grille des compétences pour un intervenant, mais cette partie n'est pas développée.

Pour consulter les informations dans le détail, il suffit de cliquer sur le nom de l'intervenant, cela renvoie sur la fiche descriptive de l'intervenant avec toutes ses informations. A ce moment là, on peut modifier ses informations ou encore supprimer cet intervenant.

Pour la saisie d'un nouvel intervenant, le seul attribut obligatoire à remplir est le nom de l'intervenant. Le rang est par défaut à 50 et le tarif à 0 si on ne connaît pas le tarif de l'intervenant. Si l'intervenant possède un autre statut que ceux prédéfinis, alors un champ de texte apparaît pour préciser la nature de ce statut.

VIII. Pistes pour le futur

Le projet étant terminé, on peut penser à certaines améliorations dont voici les idées :

- Remplir la grille de compétences pour les intervenants, c'était le but principal du projet mais malheureusement cela n'a pas pu être réalisé.
- Avoir un lien avec les évaluations pour que lorsqu'on clique sur un intervenant on puisse consulter ses évaluations.
- Avoir un lien avec les facturations des derniers cours effectués par chaque intervenant.
- Aider au plus à la décision pour que n'importe qui puisse choisir un intervenant grâce à l'application.