

TP3 – J2EE

Gestion d'un club de tennis

On se propose de développer une application pour gérer de manière très sommaire un club de tennis. Chaque adhérent pourra :

- Consulter son dossier adhérent
- S'inscrire à des tournois

Le logiciel sera installé sur un serveur d'application Tomcat (à télécharger et à installer dans le dossier personnel).

Remarques

- Quelques tables devront être créées dans une base de données (schéma en annexe 2). Les tables « Adherent » et « Tournoi » seront initialisées manuellement avec quelques adhérents et quelques tournois.
- Chaque utilisateur devra s'authentifier (« login/password ») avant d'avoir accès à ces services
- Les mots de passe ne seront pas stockés en clair dans la base de données.
- Les interactions avec la base de données se feront avec la couche de persistance Hibernate. Les requêtes seront effectuées avec le langage HQL (Hibernate Query Language).
- Toutes les actions de l'utilisateur devront être concentrées sur **une seule servlet** qui redirigera les actions en fonction d'un « code action »
- Chaque page JSP devra inclure un « haut de page » standard (statique) contenant le nom du club et ses coordonnées (adresse et téléphone)
- On prévoira une inclusion de façon à avoir un lien ou un bouton « Retour au menu » **unique et réutilisable** dans toutes les JSP qui en ont besoin.
- Seule la méthode http « POST » sera utilisée.
- On utilisera les objets de session pour transmettre des objets entre les différentes servlets.
- La durée d'une session utilisateur sera limitée à 5 minutes.

Quelques instructions :

- A la création du projet sous IntelliJ, choisir « **java enterprise** » comme type de projet et cocher les options « **Web Application** », « **Hibernate** » et « **JavaEE Persistence** »
- Package des servlets : « servlet »
- Package des beans : « bean »

Accès à l'application :

- URL : « http://serveur:port/tennis/ »
- Afficher la page d'accueil « Login.html » (adapter le fichier de déploiement web.xml : <welcome-file-list>)
- Voir annexe 1 pour l'enchaînement des actions

Login.html :

- Formulaire avec deux champs de saisie :
- Email (identifiant d'authentification)
- Password
- Bouton « Validation » → /action?code=L

ActionServlet :

- Paramètres attendus dans la requête :
- action (« action=xxx »)
- Vérifie que l'utilisateur (adhérent) possède une session active (déjà authentifié) :
- Si non :
 - Si action == « L » (« login ») forward → LoginServlet
 - Sinon forward → Login.html
- Si oui :
 - Si action == « A » (« dossier adherent ») forward → AdherentServlet
 - Si action == « I » (« inscription tournoi ») forward → InscriptionServlet
- Autre cas (autre valeur ou pas de code action) forward → Menu.jsp

LoginServlet :

Paramètres attendus dans la requête : email et mot de passe saisis dans Login.html
Si le mot de passe est correct, l'utilisateur est authentifié et on ouvre une session pour cet utilisateur ; afficher la page « Menu » : forward → Menu.jsp (le nom de l'utilisateur authentifié doit toujours apparaître).

Sinon : retour à la page de saisie du « login / password » : forward → Login.html

Menu.jsp :

Affichage du nom de l'utilisateur

Affichage du menu des services proposés :

- « Consultation de votre dossier adhérent » /action ?code=A)
- « Inscription à un tournoi » /action ?code=I

AdherentServlet :

Préparation des informations relatives à l'adhérent (informations personnelles, liste des tournois auxquels il est inscrit) : forward -> Adherent.jsp

Adherent.jsp :

Affichage des informations relatives à l'adhérent (informations personnelles, liste des tournois auxquels il est inscrit)

InscriptionServlet :

Paramètre de la requête : « Code tournoi » (tournoi=xxx)

2 cas de figure :

- pas de « code tournoi », afficher la liste de tous les tournois pour inscription éventuelle : forward → InscriptionTournois.jsp

- « code tournoi » présent, enregistrer l'inscription puis forward InscriptionStatus.jsp (pour afficher l'état de l'inscription)

InscriptionTournois.jsp :

Affichage de la liste des tournois sous forme de table html

Pour chaque ligne :

Code – Nom – Date – Lieu + [bouton « Inscription »]

Actions : pour chaque bouton « Inscription »

→ /action?code=I&tournoi=xxx

InscriptionStatus.jsp :

Confirmation (ou non) de l'inscription :

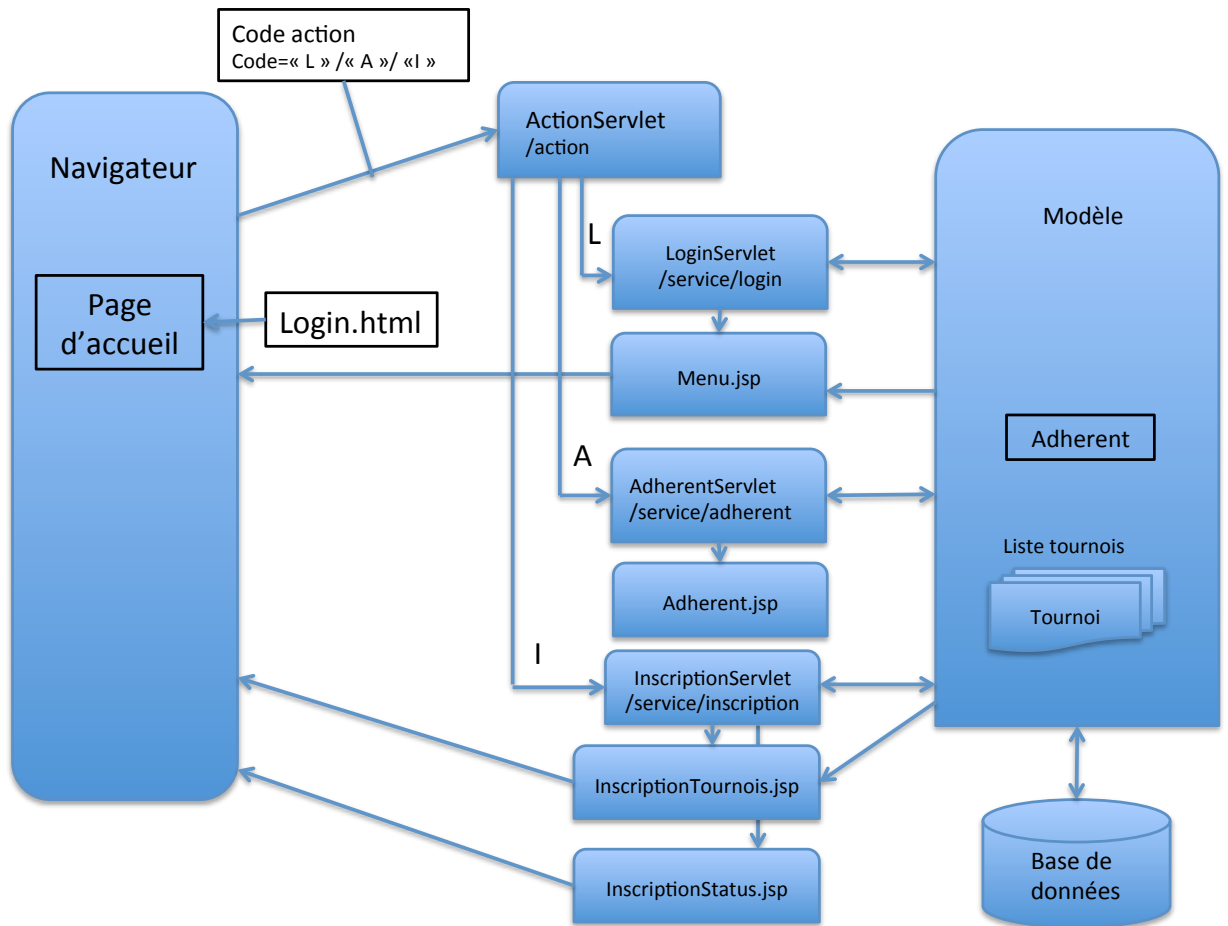
Affichage :

« Inscription enregistrée pour le tournoi » + Nom + « du » + Date + « à » + Lieu

Actions :

- Lien ou bouton « Retour au menu » (réutilisation/inclusion)

Annexe 1 : Enchainements



Annexe2 : schéma base de données

