« Gestion des Produit-SPRING MVC »

(Vous allez trouver plus d'infos au niveau du code)

I. Package Configuration :

SpringStart.java

Grace à l'annotation '`@ComponentScan(basePackageClasses = SpringStart.class)`' qui permet au compilateur de commencer par cette classe qui nous permet de :

- ✓ Déclarer notre servlet Dispatcher (on trouve une seule servlet qui reçoit les requêtes du client)
- ✓ Préciser l'extension utilisée pour notre fichier et chemin.

AppConfig.java

Avant que la servlet Dispatcher prendre en charge toute les requêtes d'utilisateur(client léger) et les acheminer aux contrôleurs (classe java), elle doit savoir l'emplacement du package Controller et l'emplacement des fichiers jsp qui sont placés dans le dossier view et représentent les fichiers qui vont être afficher au client ça dépend de son besoin(exprimé en requête) c'est pour cette raison on trouve :

- ✓ Détecter package Controller : `@ComponentScan("Controller")`
- ✓ Détecter emplacement des fichiers JSP grâce objet resolver qui fait appel à la classe `InternalResourceViewResolver()`

II. Package Controller :

On trouve aussi cette annotation ' `@Controller'` au niveau de fichier HelloController.java qui montre que le fichier joue le rôle du Controller c'est un intermédiaire entre client et les vues qui se composent de :

- ✓ `@RequestMapping :` associer une url à une méthode qui dépend de la requête du client .
- ✓ Une méthode : qui peut faire appel à des classes appartient au package Model et le résultat de cette méthode s'affichera au niveau d'un fichier JSP(`return "index", avec index c'est un JSP)`

III. Package Model :

Dans cette classe, on définit nos classes qu'on va utiliser dans notre projet et aussi une classe qui va nous connecter à notre Base de Donnée en utilisant des requêtes SQL soit en utilisant API JDBC or JPA .(pour ce projet j'ai utilisé JDBC)

On fait appel à ces classes au niveau du contrôleur.

IV. WEB INF /Views :

On a vu que le rôle du contrôleur c'est de retourner le nom de la vue et de transmet les données. On va trouver la bonne vue grâce au classe viewresolver() que j'ai déjà mentionner au niveau de la classe AppConfig.java.

Le rôle d'une vue c'est qu'elle permet d'afficher des données :JSP(HTML +code java (scriptlets)+ des tags).

EXECUTION DE NOTRE APPS :

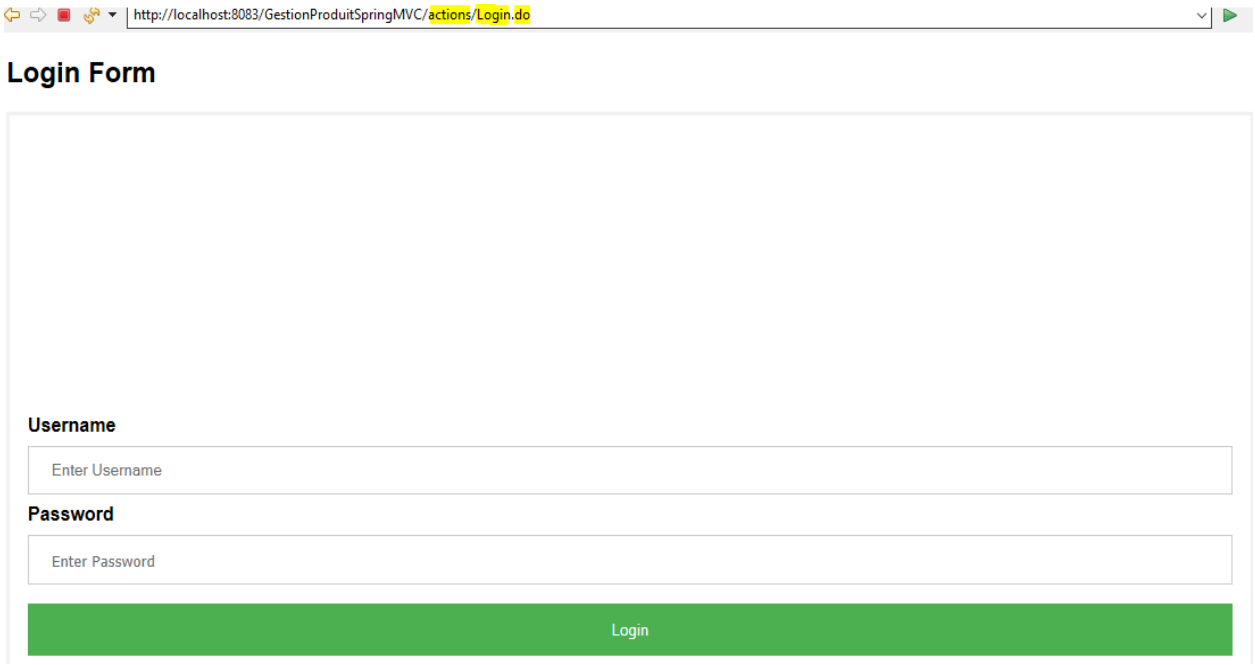
Vous devez utiliser serveur Tomcat pour exécuter votre programme. Je vous laisse une vidéo de ***Professeure Meriem Hnida*** *qui explique la Configuration du conteneur web Tomcat sur Eclipse https://youtu.be/1mxwLkodkKY.*

Et j'ai utilisé Xampp comme un serveur web qui va me permettre de me connecter à ma base de données.

Attention :

Au niveau de la barre de recherche on doit insérer le chemin de la première vue (Page login dans notre cas) et de respecter la forme de l'url et l'extension (.do) qu'on a déjà déclaré dans SpringStart.Et le package Controller va retourner la vue nommé index qu'on voit ci-dessous.

Avec : GestionProduitSpruitMVC ➔ le nom de mon projet



Je vous laisse le code qui est bien documenté pour découvrir les autres vues et comprendre les étapes de ce projet. Et si vous avez des questions n'hésitez pas de me contacter sur : « *meriem.nabil22@esi.ac.ma* »