JAVEE
PROJET MEDIATEK2022

20/03/2022

G208
Thomas Pasquier
Roche Axel

SOMMAIRE

Table des matières

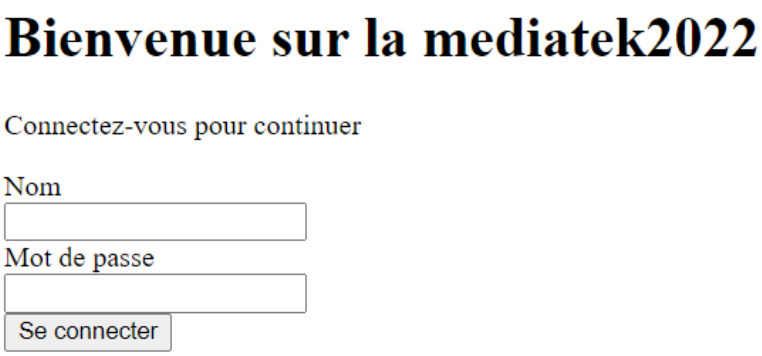
Fonctionnalités implémentées

Présentation du projet

Ce projet consiste à réaliser une application web implémentée avec sa base de données afin de fournir à une médiathèque une gestion de ses différents documents (Livres, CD et DVD) à ses abonnées en utilisant la technologie J2EE. Nous avons pour cela la nécessité de prendre

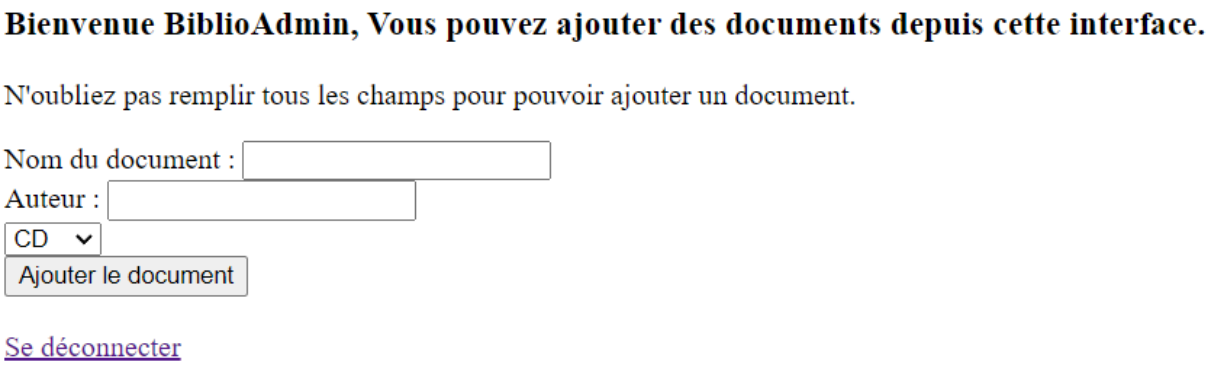
en compte plusieurs types de documents et deux différents types d'utilisateurs, Les clients (abonnés) et les bibliothécaires.
Leur gestion tel que l'ajout ou la suppression d'utilisateurs n'était pas à prendre en compte, nous avons seulement réaliser l'authentification des comptes déjà existants (jeu de données).

Interface de connexion



Fonctionnalités du bibliothécaire

Interface bibliothécaire



Ajouter des documents

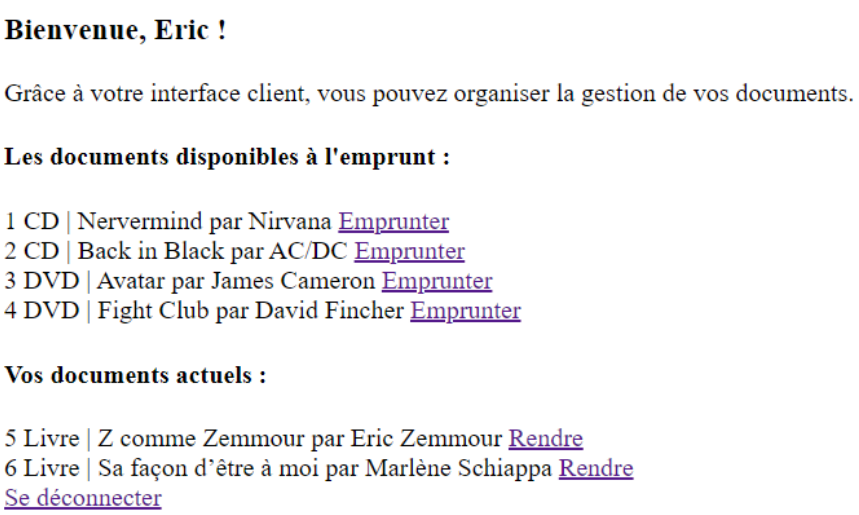
Depuis l'application web, les bibliothécaires peuvent accéder à leur propre interface web après leur authentification. Celle-ci leur permet **d'ajouter des documents** à la base de données de la bibliothèque en renseignant les caractéristiques du document à travers un formulaire.

Les caractéristiques :

- L'**auteur** de l'œuvre
- Le **nom** de l'œuvre
- Le **type** de l'œuvre

Fonctionnalités d'un client (Un abonné)

Interface client



Bienvenue, Eric !

Grâce à votre interface client, vous pouvez organiser la gestion de vos documents.

Les documents disponibles à l'emprunt :

1 CD | Nervermind par Nirvana Emprunter
2 CD | Back in Black par AC/DC Emprunter
3 DVD | Avatar par James Cameron Emprunter
4 DVD | Fight Club par David Fincher Emprunter

Vos documents actuels :

5 Livre | Z comme Zemmour par Eric Zemmour Rendre
6 Livre | Sa façon d'être à moi par Marlène Schiappa Rendre
Se déconnecter

Emprunter des documents

Un client peut **emprunter un document parmi la liste des documents disponibles**, c'est-à-dire tous les documents non-empruntés.

Rendre des documents

Après avoir emprunté des documents, il peut naturellement rendre **les documents qu'ils possèdent à la médiathèque**.

Implémentation du projet

Structure du projet

Notre projet est composé en 3 parties :

- Le package **mediatek2022** qui nous est fourni, il contient un ensemble d'interfaces qui va nous permettre de développer notre application. C'est une package stable, aucune modification n'y est autorisée.
- Le package **persistance**, composé des classes qui implémentent les interfaces de mediatek2022. On y effectue les appels vers la base de données relationnelle.
- **L'ensemble des services** de l'application qui est implémenté par des JSP.

Le package mediatek2022 nous permet de développer notre application web de manière AGILE à travers à package stable, tous nos services font appels aux interfaces de ce package, **ils ne dépendent que du package mediatek2022**.

Variables sessions

Pour la nécessité de ce projet, nous avons utilisé les variables sessions. Celles-ci nous permettent après l'authentification de l'utilisateur**, le stockage de ses données et la vérification d'accès au service** qu'il demande. Un client ne pourra pas accéder à l'interface web du bibliothécaire pour y ajouter des documents.

Chaque connexion d'un utilisateur lui offre le droit de **se déconnecter** après avoir utilisé nos services. Il peut donc se déconnecter, il sera ramené à la page de connexion et sa variable session sera vidée des données de l'utilisateur.

La gestion de la concurrence

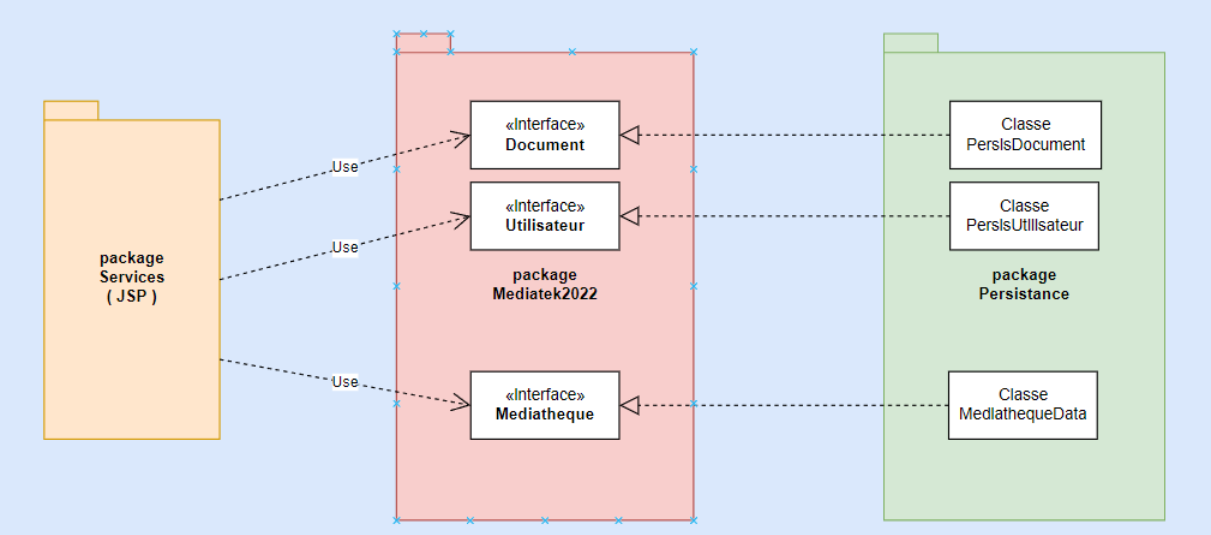
Notre application web possède quelques problèmes de concurrence. Comme plusieurs utilisateurs peuvent être connectés en même temps, deux utilisateurs peuvent **décider simultanément d'emprunter le même document**, il y a conflit et un problème de concurrence.

On remarque par ce problème que l'emprunt de document est **une ressource critique** de notre application, il est nécessaire **de bloquer les autres threads** pendant que le thread qui l'exécutera à l'aide d'un « synchronized » dans le code de la méthode. On vérifiera alors la disponibilité du document puis dans un second temps, nous mettrons à jour l'emprunt du document.

La base de données

Connexion à la base de données

La base de données que nous utilisons est une base de données relationnelle MYSQL. Nous l'avons utilisé car nous ne savons pas utiliser ORACLE XE.



Transformation objet-relationnel

Pour représenter les besoins de notre application web, nous avons besoin de deux tables dans la base de données :

- La table **utilisateurs** représente à la fois les abonnés et les bibliothécaires. Un champs type d'utilisateur sert à faire la différence entre les deux, c'est une représentation par héritage ascendant.
- La table **documents** représente les documents que possède la médiathèque.

Du côté du code Java, **une seule classe** document traduit les données des documents dans la BDD. Un objet document sera créée avec un constructor en fonction de ses données initiales.

Du côté d'un utilisateur, une distinction est faite naturellement, un bibliothécaire ne peut pas emprunter de livre alors qu'un abonné le peu. Pourtant nous avons fait le choix d'utiliser **la même classe** pour représenter les deux types. La liste de documents empruntés d'un bibliothécaire sera donc toujours vide.

Requêtes préparées

Nous avons utilisé uniquement des requêtes préparées dans ce projet afin de favoriser :

- Les **performances** : les requêtes préparées sont **précompilée**s, ce qui signifie qu'elles sont compilées qu'une seule fois contrairement aux requêtes "classiques".
- La **sécurité** : les requêtes précompilées **ne permettent pas les injections SQL.**

Améliorations à apporter

Notre projet répond globalement au sujet imposé par le cahier des charges, cependant il possède quelques axes d'améliorations secondaires :

- Une interface Web un peu plus à jour.
- La prise en compte d'autres critères sur les documents, par exemple le fait que le document ne soit disponible à l'emprunt en fonction d'une limite d'âge.
- Une meilleure gestion des erreurs utilisateurs lorsque qu'il ne remplit pas correctement un formulaire.