



Etape 6 - Projet libre

Application de Gestion Bibliothécaire

Contents

Explication de projet :	3
Liste des fonctionnalités :	4
Un schéma de base de donnée (idéalement MySQL) :	4
Table "user"	5
Table "book"	5
Table "address"	5

Explication de projet :

L'application de gestion bibliothécaire vise à assister efficacement le personnel de bibliothèque dans l'accomplissement de ses missions essentielles, à savoir :

- 1- Sélection de Titres** : L'application permet au bibliothécaire de sélectionner des titres en fonction des besoins en information, en formation, ou en divertissement des lecteurs fréquentant la bibliothèque.
- 2- Aide aux Lecteurs** : Elle offre une assistance précieuse aux lecteurs en facilitant leurs recherches et en les guidant dans le choix de leurs lectures.
- 3- Recherche Facilitée** : Les utilisateurs peuvent effectuer des recherches aisées d'ouvrages en utilisant des critères tels que le titre, l'auteur, la date d'édition, ou le contenu.

Pour atteindre ces objectifs, l'application attribue des rôles spécifiques aux utilisateurs en fonction de leurs responsabilités :

- 1. ROLE ADMIN (Gestionnaire Bibliothécaire)** : Ce rôle est attribué au gestionnaire bibliothécaire, conférant des droits complets pour exercer un contrôle total sur l'application, y compris la gestion des titres, des utilisateurs et d'autres aspects de la bibliothèque.
- 2. ROLE USER (Assistant Bibliothécaire)** : Les assistants bibliothécaires disposent du rôle ROLE_USER, adapté à leurs responsabilités spécifiques, comme l'aide aux lecteurs et l'ajout d'ouvrages.
- 3. ROLE GUEST (Lecteur Fréquentant la Bibliothèque)** : Les lecteurs fréquentant la bibliothèque se voient attribuer le rôle ROLE_GUEST, leur donnant des droits limités axés principalement sur la recherche et la consultation du catalogue des titres disponibles.

cette application facilite la gestion efficace d'une bibliothèque en attribuant des rôles spécifiques aux utilisateurs en fonction de leurs responsabilités, ce qui optimise la sélection de titres, l'assistance aux lecteurs, et la recherche d'ouvrages.

Liste des fonctionnalités :

L'application comprend les fonctionnalités suivantes :

- ❖ **Enregistrement et Connexion** : Les utilisateurs peuvent s'enregistrer et se connecter.
- ❖ **Accès Restreint** : La connexion est obligatoire pour accéder à toutes les fonctionnalités de l'application.
- ❖ **Gestion des Utilisateurs** :
 - Les gestionnaires bibliothécaires (ROLE_ADMIN) peuvent voir la liste des utilisateurs et leurs adresses.
 - En cliquant sur un utilisateur, l'application affiche ses informations et ses adresses.
 - Les utilisateurs (ROLE_USER) peuvent modifier leur profil.
 - Les administrateurs peuvent modifier et supprimer les utilisateurs.
- ❖ **Gestion des Adresses** :
 - Les utilisateurs (ROLE_USER) ont le contrôle sur leurs adresses (ajout / modification / suppression).
 - Les administrateurs ont le contrôle sur toutes les adresses (ajout / modification / suppression).
- ❖ **Gestion des Livres** :
 - Les utilisateurs (ROLE_USER) peuvent ajouter des livres.
 - Les utilisateurs (ROLE_GUEST) peuvent consulter des livres.
 - Les administrateurs ont le contrôle total sur les livres (ajout / modification / suppression).

Un schéma de base de donnée (idéalement MySQL) :

Voici le schéma de base de données proposé pour l'application :

Table "user"

```
CREATE TABLE user (
    id INT AUTO_INCREMENT PRIMARY KEY,
    username VARCHAR(255) NOT NULL UNIQUE,
    password VARCHAR(255) NOT NULL,
    role VARCHAR(255) DEFAULT 'ROLE_USER',
    creation_date DATETIME,
    updated_date DATETIME
);
```

Table "book"

```
CREATE TABLE book (
    id INT AUTO_INCREMENT PRIMARY KEY,
    title VARCHAR(50) NOT NULL,
    description VARCHAR(300) NOT NULL,
    author VARCHAR(30) NOT NULL,
    pages INT,
    quantity INT,
    image VARCHAR(255),
    date_publication DATETIME,
    creation_date DATETIME,
    last_modification_date DATETIME
);
```

Table "address"

```
CREATE TABLE address (
    id INT AUTO_INCREMENT PRIMARY KEY,
    street VARCHAR(100) NOT NULL,
    postal_code VARCHAR(30) NOT NULL,
    city VARCHAR(50) NOT NULL,
    country VARCHAR(50) NOT NULL,
    creation_date DATETIME,
    updated_date DATETIME,
    user_id INT NOT NULL,
    FOREIGN KEY (user_id) REFERENCES user(id)
);
```