Laura Huber

Direction interrégionale des services pénitentiaires de Paris

**PORTEFEUILLE D’ACTIVITES PROFESSIONNELLES**



Apprentie en développement

|  |  |
| --- | --- |
| **Compte rendu d’Activité** | Fiche n° 4 |

|  |  |
| --- | --- |
| Libellé court | Compte rendu Projet Personnel Encadré E5 |
| **Description** | Compte rendu d’un projet de développement d'un site de bibliothèque. |
| **Contexte** | Mission accomplie dans le cadre de la formation |

|  |  |
| --- | --- |
| **Localisation** | **Entreprise**  **CFA** |
| **Source** | **1ère année**  **2ème année**  **TP**  **PPE** |
| **Cadre** | **Seule**  **En équipe** |
| **Type** | **Vécu** **Observé**  **Simulé** |

|  |  |
| --- | --- |
| **Environnement Technologique** | Visual Studio Code, WAMP, phpMyAdmin |
| **Moyens** | HTML5, CSS, Bootstrap, PHP, JavaScript, MySQL, GitHub |
| **Avis personnel** | Projet de développement web qui m'a permis d'approfondir mes connaissances et qui a un fort potentiel d'évolution. |

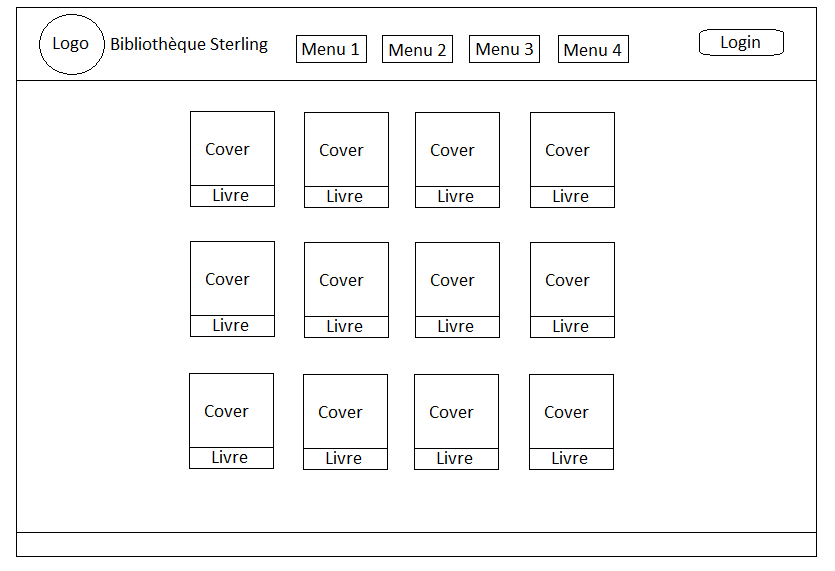
|  |  |
| --- | --- |
| **Activités mises en œuvre pour la réalisation de cette situation** | |
|  | 1. **Organisation de son développement professionnel** |
|  | 1. **Conception et développement d’une solution applicative** |
|  | 1. **Maintenance corrective et évolutive d’une solution applicative** |
|  | **Gestion des données** |

**Déroulement de l’activité :**

Dans le contexte de la M2L (Maison des Ligues de Lorraine), j'ai mis en place une application en client léger, plus particulièrement un site web, la "bibliothèque Sterling", qui permet des télécharger des livres numériques en format PDF.

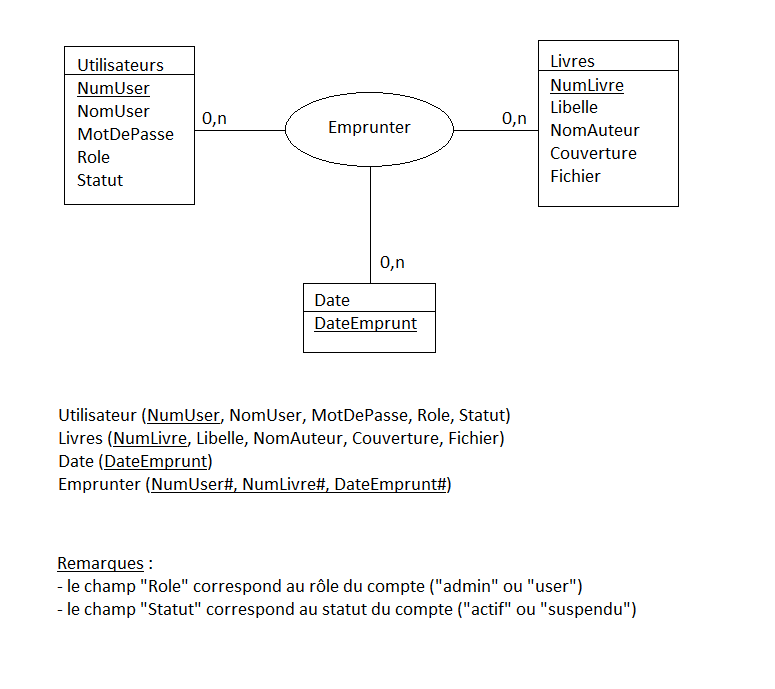
Tout d'abord, j'ai choisi les langages adaptés à mes besoins. J'ai décidé d'utiliser HTML et CSS pour le frontend, en utilisant le framework Bootstrap afin de gagner du temps et d'assurer une comptabilité avec la plupart des navigateurs, mais aussi parce que sa documentation est très complète. En outre, j'ai déjà utilisé Bootstrap au cours d'une autre mission et j'avais donc pu me familiariser avec celui-ci. Pour le backend, je me suis servie de PHP et de JavaScript et pour la base de données j'ai opté pour MySQL.

Pour commencer, j'ai créé une maquette simple afin d'avoir une idée de la position des différents éléments.



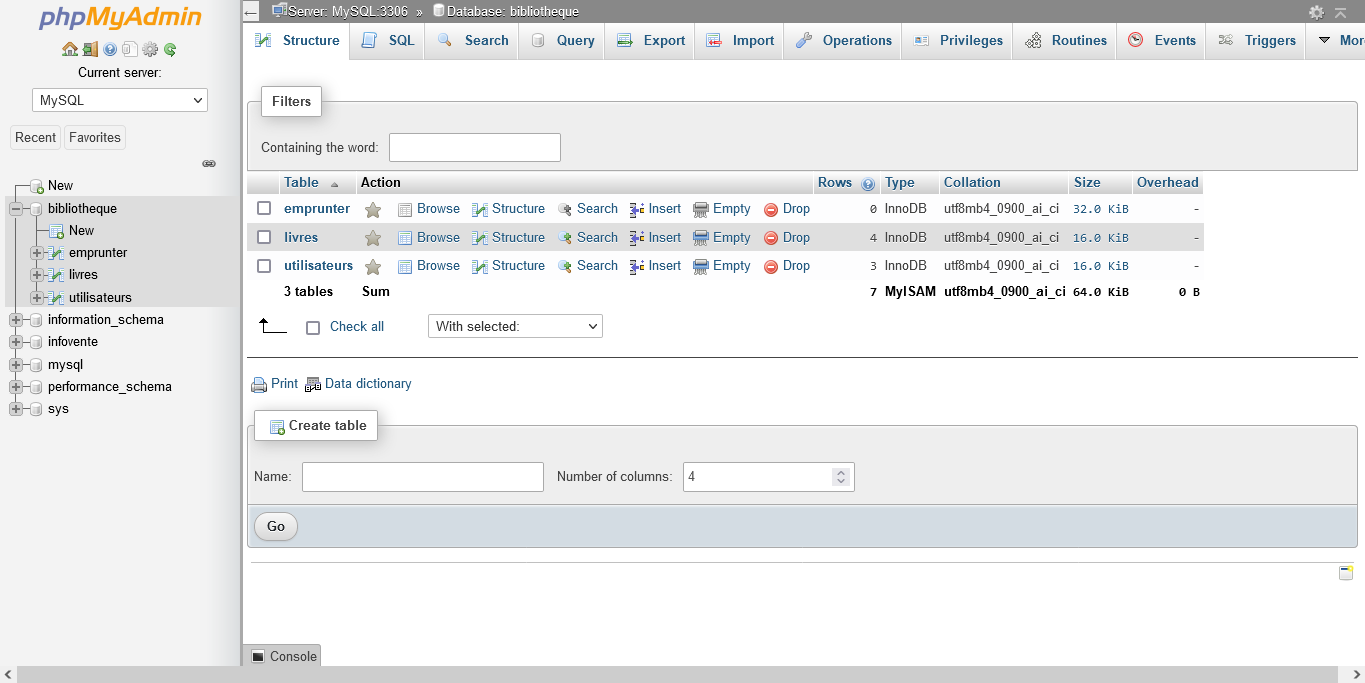
La maquette de la page d'accueil

J'ai également réalisé le MCD (Modèle Conceptuel des Données) et le MLD (Modèle Logique des Données) des tables dont j'aurai besoin dans la base de données.



Le MCD et le MLD de la base de données

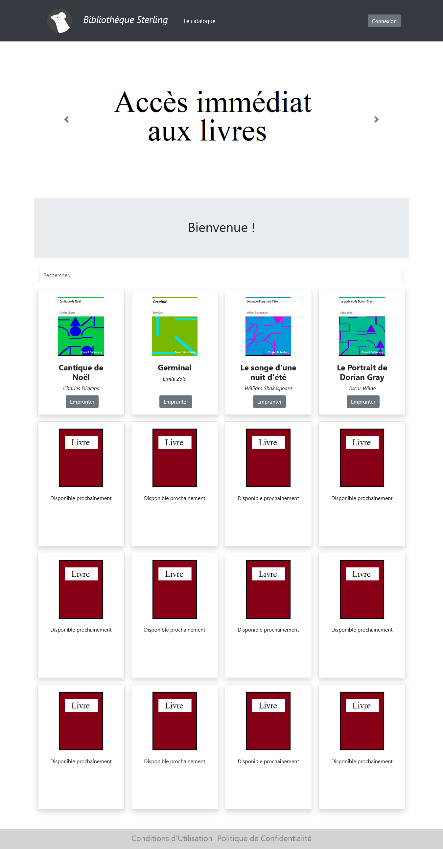
Ensuite, j'ai installé WampServer et j'ai mis en place la base de données MySQL dans phpMyAdmin.



La base de données dans phpMyAdmin

Avant de me lancer dans le projet, j'ai également installé l'application de bureau de GitHub pour la gestion de versions avec Git.

La première page que j'ai mise en place est la page d'accueil qui affiche le catalogue. Au début, j'avais inséré les livres en dur dans le code mais j'ai changé cela par la suite afin de stocker les livres dans la base de données. J'ai également créé la barre de menu et le bas de page avec les conditions d'utilisation et la politique de confidentialité ainsi qu'un carrousel qui affiche du texte.

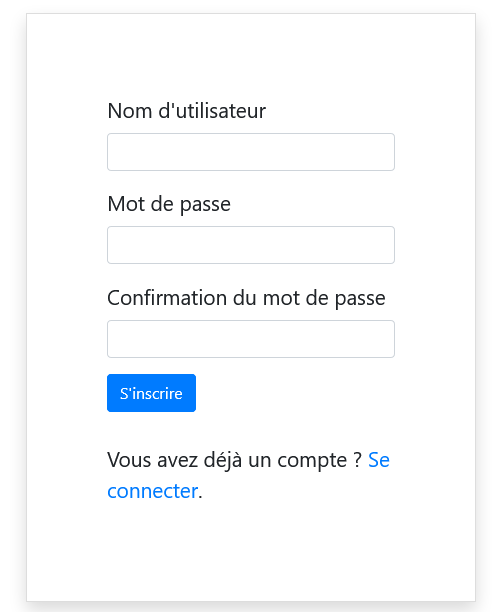


Le catalogue



Récupération des livres stockés dans la base de données

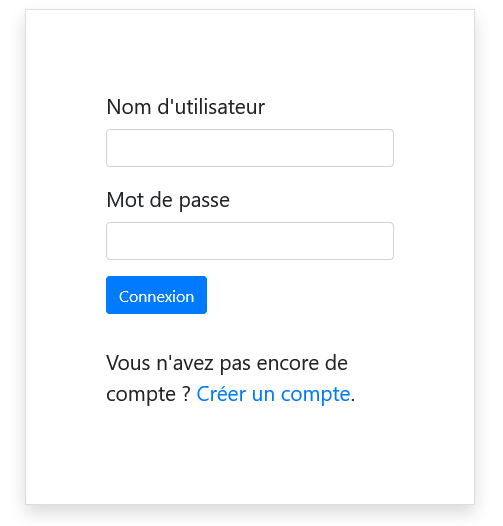
J'ai ensuite ajouté un formulaire de création de compte ainsi qu'un formulaire de connexion.



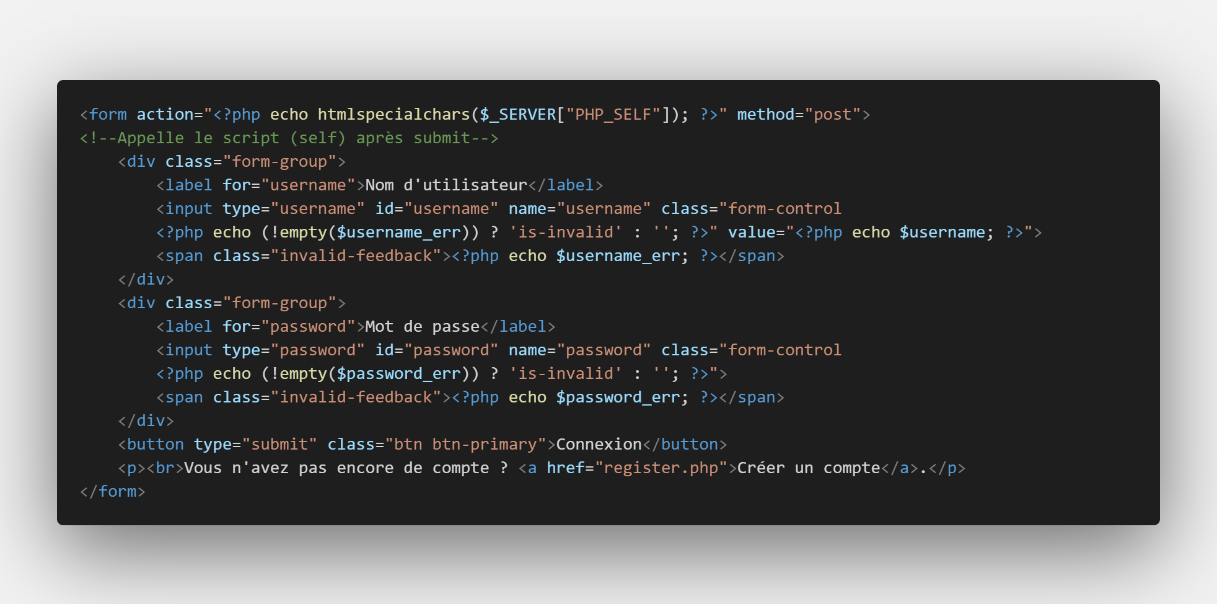
Le formulaire de création de compte



Le code du formulaire de création de compte



Le formulaire de connexion



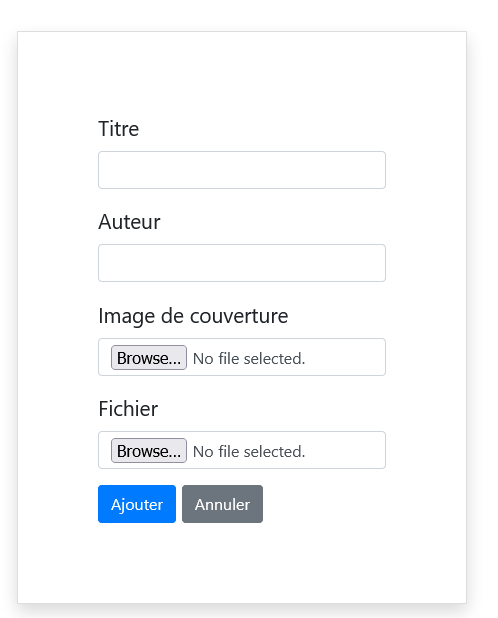
Le code du formulaire de connexion

Enfin, j'ai ajouté des rôles. Selon le rôle associé au compte, les fonctionnalités du site changent. Le rôle correspondant au compte connecté est stocké dans une variable globale afin de vérifier à quelles pages et fonctionnalités l'utilisateur peut accéder. Cela empêche également la manipulation d'URL.



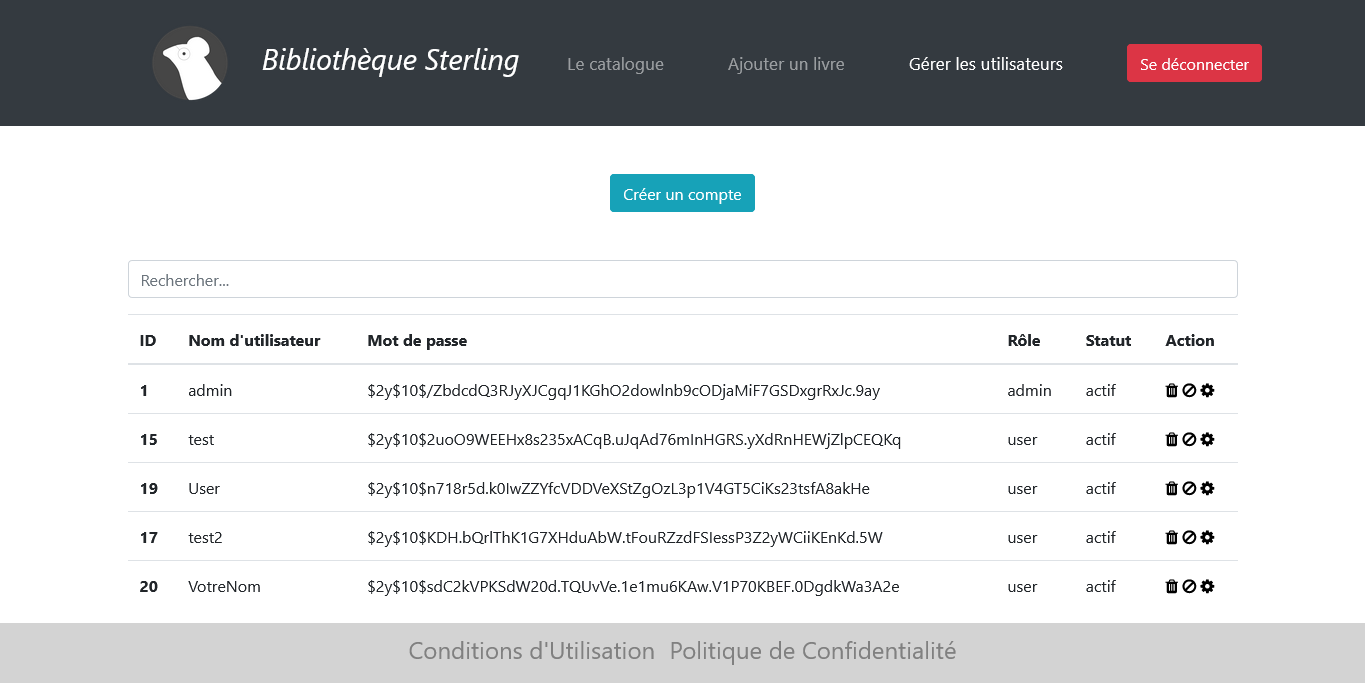
Vérification du rôle du compte

Pour les comptes "admin", j'ai mis en place un formulaire qui permet d'ajouter des livres au catalogue. La base de données stocke ensuite le chemin vers l'image de la couverture et vers le fichier pdf du livre et une copie des deux est ajoutée à un répertoire destiné à cet effet créé après envoi du formulaire.



Le formulaire pour ajouter un livre

En outre, j'ai mis en œuvre une page de gestion des utilisateurs pour les comptes "admin" qui leur permet de voir les utilisateurs existants, de créer et de supprimer des comptes et de modifier le nom d'utilisateur, le mot de passe et le rôle associé à un compte. Ils peuvent également suspendre des comptes. Les comptes suspendus ne se peuvent plus connecter jusqu'au moment où ils sont réactivés.



La page de gestion des utilisateurs



Code pour la suppression d'un compte

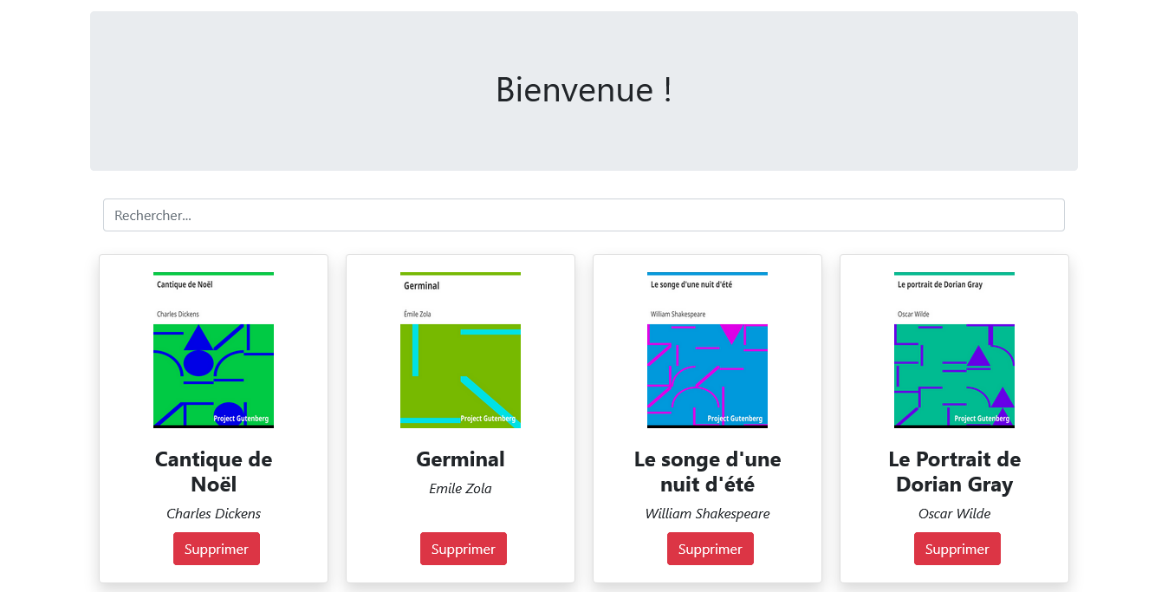


Code pour la suspension d'un compte



Code pour la modification d'un compte

J'ai aussi ajouté une fonctionnalité qui permet aux comptes "admin" de supprimer des livres du catalogue.

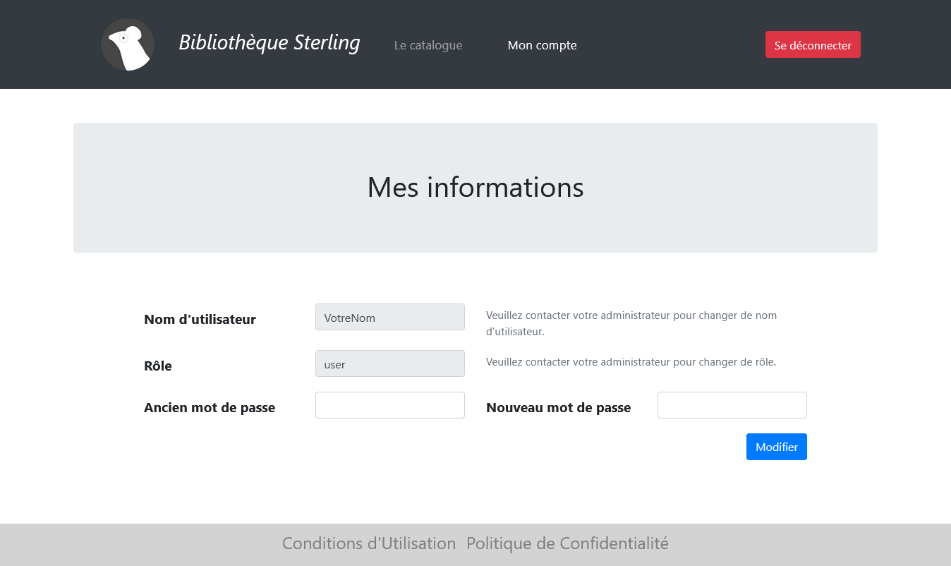


Le catalogue vu par un compte ayant pour rôle "admin"



Le code qui permet de supprimer un livre

Ensuite, j'ai implémenté un site de gestion de compte qui permet aux utilisateurs ayant le rôle "user" de changer leur mot de passe et de visualiser leur nom d'utilisateur ainsi que leur rôle.



La page de gestion de compte



Le code pour changer son mot de passe

Pour terminer, j'ai mis en place une historique d'emprunt en sauvegardant l'identifiant de l'utilisateur, l'identifiant du livre et la date dans la base de données lors qu'un utilisateur emprunte un livre.



Le code pour enregistrer les emprunts