TP3 - Gestion d'utilisateurs et BDD

Créez dans votre clone Git un répertoire `TP3` et copiez les fichiers du `TP2`.

Cet exercice vous propose de mettre en place la gestion des utilisateurs dans une base de données.

Note : il est fortement recommandé d'avoir réalisé l'exercice 5 du `TP2` afin de transmettre les message d'erreur de php au client.

Mise en place de la BDD et authentification

voir fin du sujet du `TP2`

Authentification

1. Créez un nouveau fichier `bdd.php` contenant 3 constantes correspondant aux 3 paramètres de la construction d'un objet PDO.
2. Modifiez le fichier `authenticate.php` pour qu'il authentifie les utilisateurs à partir de la BDD en utilisant PDO. Ceci remplace l'utilisation du fichier `users.php` du TP précédent.

Note : Pensez à gérer les exceptions PHP déclenchées par la construction d'un objet PDO. Si vous avez fait l'exercice 5 du TP2, vous pouvez ajouter ces messages d'erreur à la variable de session `message`.

Exercice 1 : Inscription

1. Écrivez un nouveau fichier `signup.php` qui propose un formulaire d'inscription pour un nouvel utilisateur et le soumet à la page `adduser.php`.
2. Écrivez un nouveau fichier `adduser.php` qui effectue les traitements suivants :
   - il vérifie que la méthode utilisée pour l'appeller est `POST`
   - il vérifie et sécurise les champs du formulaire de `signup.php`
   - il demande une redirection vers `signup.php` si le mot de passe et sa confirmation diffèrent
   - il tente d'insérer le nouvel utilisateur :
     - si la requête s'est bien passée, il demande une redirection vers `signin.php`
     - sinon il demande une redirection vers `signup.php`
3. Pour plus de navigabilité, ajoutez un lien pour s'inscrire sur `signin.php` et, inversement, un lien pour se connecter sur `signup.php`.

Note 1 : Si vous avez fait l'exercice 5 du TP2, vous pouvez ajouter les messages d'erreur et de réussite à la variable de session `message`.

Exercice 2 : Mots de passe cryptés

Actuellement les mots de passe sont codés en clair dans votre base de données. Vous allez donc mettre en place le cryptage (et le décryptage) des mots de passe.

Note : Si tout a bien été fait jusque là, les questions 2. et 3. de cet exercice nécessite de ne changer **qu'une seule ligne** dans chacun des fichiers.

1. Supprimez de votre BDD tous les utilisateurs inscrits, via PhpMyAdmin.
2. Modifiez le fichier `adduser.php` afin qu'il enregistre le mot de passe chiffré avec la fonction PHP `password_hash`. Attention, lisez bien sa documentation et, si besoin, procédez aux modifications des attributs de la colonne du mot de passe dans PhpMyAdmin.
3. Modifiez le fichier `authenticate.php` pour qu'il compare le mot de passe du formulaire avec celui récupéré dans la BDD à l'aide de la fonction `password_verify`.

Exercice 3 : Modification du mot de passe

Cet exercice propose d'ajouter la fonctionnalité de changement de mot de passe avec deux nouveaux fichiers : la vue `formpassword.php` et le contrôleur `changepassword.php`.

1. Créez un nouveau fichier `formpassword.php` contenant un formulaire avec deux champs : son son nouveau mot de passe et la confirmation du nouveau mot de passe. Ce formulaire a pour cible la page `changepassword.php`.
2. Assurez-vous que ce formulaire soit présenté uniquement si l'utilisateur est connecté ET qu'il demande cette page avec un requête HTTP de type GET. Si ce n'est pas le cas, redirigez-le vers `signin.php`.
3. Ajoutez un lien vers `formpassword.php` sur la page `welcome.php`, ainsi qu'un lien vers `welcome.php` sur la page `formpassword.php`.
4. Créez un nouveau fichier `changepassword.php` qui effectue, dans l'ordre, les traitement suivants :
   - il vérifie que l'utilisateur est connecté et accède à cette page via une requête HTTP de type POST. Si ce n'est pas le cas, il demande une redirection vers `formpassword.php`.
   - il vérifie et sécurise les données transmises en POST : si elles n'existent pas, il demande une redirection vers `formpassword.php`.
   - il vérifie que le mot de passe et sa confirmation sont identique : si ce n'est pas le cas, il demande une redirection vers `formpassword.php`.
   - si tout est bon jusque là, il change le mot de passe de l'utilisateur dans la BDD.
   - si la requête de changement de mot de passe se passe bien, il demande une redirection vers `welcome.php`. Sinon il demande une redirection vers `formpassword.php`.

Si vous gérez les messages d'erreur (voir Exercice 5 du TP2), pensez à mettre à jour la variable de session `message` avant chaque demande de redirection dans

`changepassword.php`.

Exercice 4 : Suppression d'un utilisateur

1. Ajouter un lien vers `deleteuser.php` dans le fichier `welcome.php`.
2. Créez un nouveau fichier `deleteuser.php` qui effectue, dans l'ordre, les traitement suivants :
   - il vérifie que l'utilisateur est connecté et accède à cette page via une requête HTTP de type GET. Si ce n'est pas le cas, il demande une redirection vers `welcome.php`.
   - si tout est bon jusque là, il supprime l'utilisateur de la BDD
   - si la requête s'est bien passée, il supprime la session et demande une redirection vers `signin.php`. Sinon il demande une redirection vers `welcome.php`.

Si vous gérez les messages d'erreur (voir Exercice 5 du TP2), pensez à mettre à jour la variable de session `message` avant chaque demande de redirection dans `deleteuser.php`.