
Module 151

Version 1.0.0

CPNV

août 31, 2024

Table des matières

No Module

151

Titre

Intégrer des bases de données dans des applications Web.

Compétence

Intégrer une base de données existante respectivement des éléments de données dans une application Web en tenant compte des exigences techniques et de sécurité.

Objectifs opérationnels

1. Analyser les exigences d'une application Web et de la base de données, respectivement des éléments de données à lier, définir et documenter la technique de liaison.
2. Identifier les informations importantes de protection et de sécurité en tenant compte de la protection des données, et définir les mesures.
3. Réaliser l'intégration de l'application Web avec la base de données, respectivement aux éléments de données, en prêtant attention aux transactions, à la protection et la sécurité des données.
4. Mettre en œuvre les souhaits de modifications conformément au déroulement prescrit des modifications.
5. Définir et exécuter la procédure de test et de remise, la documenter dans un procès-verbal de tests. Si nécessaire, entreprendre les corrections.

Author

Cindy Hardegger

Email

cindy.hardegger@eduvaud.ch

Mise à jour

août 31, 2024

Références

- Identification du module
- DEP

Table des matières

CHAPITRE 1

Connecteur MySQL

Afin de connecter une base de données MySQL avec le langage PHP, un connecteur est utilisé :

- Connecteur MySQL est de type procédural, il est devenu obsolète depuis la version PHP 5.5 (Ne plus utiliser)
- Connecteur MySQLi est de type OO ou procédural, existe depuis la version 4.1
- Connecteur PDO (PHP Data Objects) est de type OO et existe depuis la version 5.1 (celle que nous utiliserons)

1.1 PDO

PDO est une extension, autrement dit, elle doit être installée sur le serveur pour être utilisée. Le `phpinfo()` peut vous renseigner sur l'activation de cette dernière. Pour l'activer, l'extension `php_pdo_mysql` doit être cochée sur votre serveur.

L'avantage est de pouvoir se connecter sur différents systèmes de gestion de base de données tel que MySQL, MariaDB, Oracle ou PostgreSQL.

1.1.1 Se connecter

En créant un objet PDO, plusieurs paramètres sont passés, notamment l'hôte, le nom de la base de donnée, le charset, le nom d'utilisateur et le mot de passe. En cas de succès, un objet PDO est retourné.

```
// Se connecter via PDO
$connector = new PDO('mysql:host=localhost;dbname=db_1;charset=utf8' , 'root', 'pass');
```

En cas d'erreur, il faut la gérer, voici un exemple :

```
try
{
    $connector = new PDO('mysql:host=localhost;dbname=db_1;charset=utf8' , 'root', 'pass
    ↵');
}
catch (PDOException $e)
```

(suite sur la page suivante)

(suite de la page précédente)

```
{  
    die('Erreur : ' . $e->getMessage());  
}
```

1.1.2 Envoyer une requête SQL

Deux méthodes sont utilisées pour faire une requête.

La méthode query qui permet d'effectuer une requête et d'obtenir un objet PDOStatement. Avec ce retour, vous devrez utiliser un autre méthode pour traiter les données. Ce type de code est à utiliser uniquement sur des requêtes qui ne comportent **aucun** paramètre.

```
// Utilisation de query pour effectuer une requête  
$req = $connector->query('SELECT * FROM Table');
```

Une bonne pratique quand on envoie des requêtes et plutôt d'utiliser la préparation des requêtes. Elle permet en cas de paramètres de s'assurer que les variables passées soient sûres. Le risque en ne le faisant pas sont les injections SQL !

```
// Utilisation de prepare, bindValue et execute  
$req = $connector->prepare('SELECT * FROM Table WHERE id = :varId AND input = :varInput  
');  
$req->bindValue('varId', $id, PDO::PARAM_INT);  
$req->bindValue('varInput', $input, PDO::PARAM_STR);  
$req->execute();
```

1.1.3 Traiter les données

Dans le cas d'un affichage, les données recueillies lors de la requête doivent être traitée et retournée dans le format souhaité.

```
// Traitement, transformer le résultat en tableau associatif  
$result = $req->fetchALL(PDO::FETCH_ASSOC);
```

1.1.4 Vider le jeu d'enregistrements

```
// Vider le jeu d'enregistrements  
$req->closeCursor();
```

1.1.5 Fermer la connexion

```
// Fermer la connexion en mettant l'objet à null  
$connector = null;  
  
// Ou en utilisant la méthode unset  
unset($connector);
```

CHAPITRE 2

Fil rouge - Surnom dans les jeux vidéos

Il s'agit de mettre en place une petite application permettant de recenser les surnoms dans les jeux vidéos.

Les entités à gérer (de manière simple) sont :

- Les jeux vidéos
- Les pseudos
- Les utilisateurs (pour la connexion)

Un jeu vidéo (groupe dont dépend les pseudos) a un nom pour seule information.

Un-e pseudo a les informations suivantes :

- un pseudo
- un genre pour indiquer s'il s'agit d'une femme, d'un homme (valeurs possibles w (woman), m (man), o (other))
- une origine (texte qui définit pourquoi il a choisi ce pseudo)
- une date (qui permet de revoir depuis combien de temps, il utilise ce pseudo)

Un-e utilisateur-trice a les informations suivantes :

- un login
- un mot de passe
- un droit administrateur

2.1 Base de données

La première étape consiste à réaliser la base de données. Le modèle vous est fourni, il faut contrôler qu'il réponde aux exigences du dessus et aux normes du CPNV.

Téléchargement du fichier SQL

La base de données sera accessible pour l'utilisateur suivant :

- Utilisateur : dbPseudoUser
- Mot de passe : *** (fourni par votre enseignant)

Donc, vous devez créer dans phpMyAdmin un utilisateur pour la base de données. Cette dernière se nommera **db_pseudo_<prenom>**. Remplacer le <prenom> par votre prénom.

Créer votre base de données et y ajouter 2 pseudos et 2 jeux vidéos.

2.1.1 Arborescence

Mettre en place l’arborescence du site des pseudos en utilisant le modèle MVC.

2.1.2 Convention

Les normes de codage du CPNV doivent être utilisée.

2.2 Page d'accueil

Créer la première page du site en la plaçant correctement dans l’arborescence. Elle se nommera index.php.



Les pseudos

Zone pour le menu

Liste des pseudos

Pseudo	Genre	Option
Bubu	Homme	
Dédé	Homme	
Dobby	Autre	

Copyright nom de l’élève - Année

Cette page est faite avec du html et doit être validée W3C. L’utilisation du CSS est optionnel. Pour l’instant, les données sont insérées de manière statique.

2.2.1 Données dynamiques

Une fois que la page est prête, l’étape suivante consiste à récupérer les données de la base de données MySQL et de les insérer à la place des textes statiques.

2.2.2 Création d'une classe pour gérer la BD

Pour l'instant, les données sont traitées et affichées directement dans la page d'accueil. Mais, nous allons créer plusieurs pages avec des liaisons à la BD. A terme, nous ne voulons pas recopier toujours les mêmes instructions. De ce fait, nous allons créer une classe afin de regrouper les méthodes nécessaires à nos requêtes.

Téléchargement du fichier Database

Nom de la classe : Database

Méthodes contenues :

- __construct() : créer un nouvel objet PDO et se connecter à la BD
- querySimpleExecute(\$query) : permet de préparer et d'exécuter une requête de type simple (sans where)
- queryPrepareExecute(\$query, \$binds) : permet de préparer, de binder et d'exécuter une requête (select avec where ou insert, update et delete)
- formatData(\$req) : traiter les données pour les retourner par exemple en tableau associatif (avec PDO : :FETCH_ASSOC)
- unsetData(\$req) : vide le jeu d'enregistrement
- getAllPseudos() : récupère la liste de tous les pseudos de la BD
- getOnePseudo(\$id) : récupère la liste des informations pour 1 pseudo
- ... + tous les autres méthodes dont vous aurez besoin pour la suite (insertPseudo ... etc)

2.2.3 Externalisation des informations de la BD

Vous avez stocké les accès à la BD en « dur » dans la classe. Cette pratique implique que celui qui veut l'utiliser devra changer les informations d'accès. Normalement, une classe ne doit pas être modifiée par celui qui l'utilise. De ce fait, vous allez créer un fichier de configuration qui contiendra les données d'accès à la BD. Ce fichier se nomme config.php.

2.3 Page de détail d'un pseudo

Depuis la page d'accueil, en cliquant sur la loupe (détail) d'un pseudo, un lien nous amène sur la fiche détaillée du pseudo.

Dans un premier temps, sur la page d'accueil, le lien doit être ajouté pour chaque pseudo avec l'information de son id. Exemple si le pseudo a l'ID 17, mon lien sera :

```
<a href="detail.php?idPseudo=17">Détail</a>
```

Ensuite, il faut créer la page detail.php et récupérer l'id du pseudo, en utilisant \$_GET, puis d'afficher les informations selon le mockup fourni.



Les pseudos

Zone pour le menu

Bubu

Homme

Liste des jeux vidéos



Origine

Son nom de famille lui a donné son surnom !

Fortnite

League of legend

Utilise ce pseudo depuis 26 juillet 2018

[Retour à la page d'accueil](#)

Copyright nom de l'élève - Année

2.4 Insertion d'un pseudo

Créer une page addPseudo.php qui contiendra un formulaire avec les mêmes champs que sur la maquette.



Les pseudos

Zone pour le menu

Ajout d'un pseudo

Homme Femme Autre

Pseudo :

Date :

Origine :

[Retour à la page d'accueil](#)

Copyright nom de l'élève - Année

Un lien « Retour à la page d'accueil » doit être présent et fonctionnel.

Le bouton « Annuler » permet de vider tous les champs. Le bouton « Ajouter » permet de contrôler les champs et si tout est ok, une insertion du pseudo dans la BD est effectuée. Ensuite, on retourne sur la page d'accueil.

2.5 Modification d'un pseudo

Créer une page updatePseudo.php qui contiendra un formulaire avec les mêmes champs que sur la maquette. Les champs sont remplis avec les informations de la BD. Afin de reprendre le bon pseudo, il faut passer l'id du pseudo via URL.

Application pseudos

localhost:8080/pseudo?id=34

Les pseudos

Zone pour le menu

Modification d'un pseudo

Homme Femme Autre

Pseudo : Bubu

Date : 26.07.2018

Origine :

Son nom de famille lui a donné son surnom !

Fortnite
League of legend
Rocket League

Modifier [Retour à la page d'accueil](#)

Un lien « Retour à la page d'accueil » doit être présent et fonctionnel.

Le bouton « Modifier » permet de contrôler les champs et si tout est ok, une modification du pseudo dans la BD est effectuée. Ensuite, on retourne sur la page d'accueil.

2.6 Suppression d'un pseudo

En cliquant sur la corbeille, il est possible de supprimer le pseudo voulu, mais seulement après confirmation. A quoi faut-il faire attention lors de la suppression ?

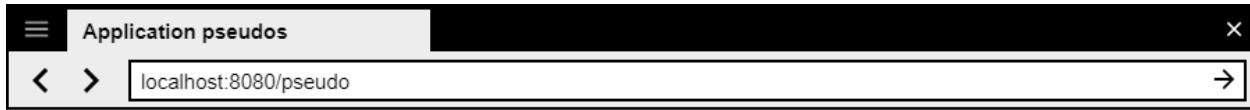
Pour la confirmation, utiliser JavaScript et la méthode confirm().



Copyright nom de l'élève - Année

2.7 Ajout de la connexion

Ajouter les champs nécessaires pour la connexion (Login, Mot de passe et Se connecter). En cas de non-connexion, rien n'est visible sauf la page d'accueil.



Les pseudos

Login

Mot de passe

Se connecter

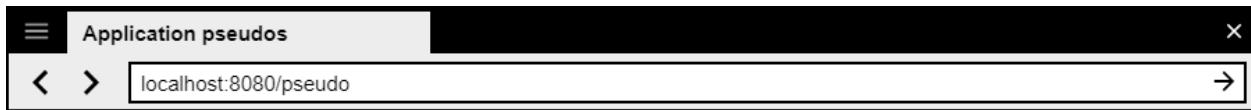
Zone pour le menu

Liste des pseudos

Pseudo	Genre	Option
Bubu	Homme	
Dédé	Homme	
Dobby	Autre	

Copyright nom de l'élève - Année

Si on se connecte avec un droit utilisateur, il est possible de visualiser la page d'accueil et la page de détail d'un pseudo uniquement. Les autres pages ne sont pas accessibles.



Les pseudos

Albert (user)

[Se déconnecter](#)

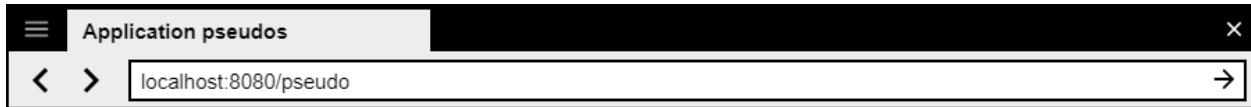
Zone pour le menu

Liste des pseudos

Pseudo	Genre	Option
Bubu	Homme	🔍
Dédé	Homme	🔍
Dobby	Autre	🔍

Copyright nom de l'élève - Année

Si on se connecte avec un droit admin, il est possible de tout visualiser et d'utiliser les fonctionnalités.



Les pseudos

Edouard (admin)

[Se déconnecter](#)

Zone pour le menu

Liste des pseudos

Pseudo	Genre	Option
Bubu	Homme	✍️ 🗑️ 🔎
Dédé	Homme	✍️ 🗑️ 🔎
Dobby	Autre	✍️ 🗑️ 🔎

Copyright nom de l'élève - Année

CHAPITRE 3

Sources

3.1 Livre

- Exemple, Auteur, Edition ENI, ISBN 9782746092143

3.2 Site

- Exemple