

# Forum minimal PHP et MySQL

## V3.1.0

---



Cette œuvre est mise à disposition selon les termes de la [licence Creative Commons Attribution – Pas d'Utilisation Commerciale – Partage à l'Identique 3.0 non transposé](#).

Document en ligne : [mickael.martin.nevot.free.fr](http://mickael.martin.nevot.free.fr)

---

## 1 Généralités

Les fonctionnalités d'un forum minimal PHP et MySQL décrites dans ce document ne forment qu'un exemple de réalisation (même s'il s'agit probablement de la base la plus utilisée) et ont pour objectif de guider et d'aider à la réalisation d'un tel forum : il ne s'agit pas de règles à appliquer absolument et encore moins d'un socle exhaustif de connaissances dans ce domaine.

## 2 Architecture de la base de données

Un forum PHP et MySQL doit avoir une base de données contenant au minimum les relations suivantes (les clefs primaires étant soulignées et les clefs étrangères en *italique*) :

- `user` (`user_id`, `name`, `firstname`, `nickname`, `email`, `password`) ;
- `post` (`post_id`, `title`, `message`, `post_date`, *`user_id`*).

En outre, dans la relation `user`, `email` est unique.

## 3 Architecture des fichiers du forum

Un tel forum doit contenir au minimum l'équivalent des fichiers suivants :

- `forum.php` ;
- `login.php` ;
- `subscribe.php` ;
- `add.php` ;
- `listpost.php`.

### 3.1 `forum.php`

La page Web correspondant à ce fichier est la page d'accueil du forum. Elle est composée de deux formulaires de méthode POST : un renvoyant vers `login.php`, l'autre vers `subscribe.php` :

## Formulaire de connexion au forum

Figure 1 – Formulaire de connexion au forum

### 3.2 login.php

La page Web correspondant à ce fichier permet de gérer la connexion au forum.

Si l'*e-mail* et le mot de passe sont présents dans la *supervariable* `$_POST`, il y a :

- connexion à la base de données ;
- récupération de l'utilisateur avec une requête de la forme :
 

```
<?php
    $query = 'SELECT user_id, email, password FROM user where email=\' ' .
    $_POST['email'] . '\';
    ?>
```
- vérification que le mot de passe en base de données soit identique à celui renseigné ; **si c'est le cas**, il y a :
  - démarrage de la session (`session_start()`) ;
  - mise en session de l'*e-mail* et de l'empreinte du mot de passe (par exemple) ;
  - redirection vers `listpost.php`, par exemple, de la manière suivante :
 

```
<?php
    header('location:./listpost.php');
    ?>
```
- **sinon** :
  - redirection vers la page de connexion (pour permettre une nouvelle connexion) :
 

```
<?php
    header('location:./forum.php');
    ?>
```
  - affichage d'un message d'erreur signalant que l'utilisateur n'existe pas ;

**sinon**, il y a :

- redirection vers la page de connexion ;
- affichage d'un message d'erreur précisant que toutes les données utilisateurs ne sont pas renseignées.

### 3.3 subscribe.php

La page Web correspondant à ce fichier permet de gérer l'inscription au forum.

Si toutes les informations nécessaires à l'inscription sont présentes, il y a :

- vérification que le mot de passe et la confirmation de mot de passe soient identiques ;
- connexion à la base de données ;
- insertion de l'utilisateur en base de données avec une requête du genre :
 

```
<?php
```

```
$query = 'INSERT INTO user (name, firstname, nickname, email, password) VALUES (\'' . $_POST['name'] . '\', \'' . $_POST['firstname'] . '\', \'' . $_POST['nickname'] . '\', \'' . $_POST['email'] . '\', \'' . md5($_POST['password']) . '\')';
?>
```

- envoi d'une *e-mail* pour avertir le nouvel inscrit ;
- authentification automatique en utilisant une redirection vers le fichier `login.php` par exemple (qui lancera donc lui-même une redirection vers le fichier `listpost.php`) ;

sinon, il y a :

- redirection vers la page de connexion ;
- affichage d'un message d'erreur précisant que toutes les données utilisateurs ne sont pas renseignées.

### 3.4 listpost.php

La page Web correspondant à ce fichier permet de visualiser l'ensemble des messages postés et d'en ajouter un nouveau.

Il y a tout d'abord l'affichage de l'ensemble des messages postés (ainsi que leur auteur) enregistrés en base de données. Les enregistrements peuvent être récupérés en base de données, par exemple de la manière suivante :

```
<?php
$query = 'SELECT * FROM post P, user U WHERE P.user_id = U.user_id';
?>
```

Il y a aussi un formulaire de méthode POST qui permet de poster un message (et qui renvoie vers `add.php`).

Pseudo : admin	2009-01-06
Prénom : Admin	Sans avis
Nom : Admin	
Pseudo : admin	2009-01-06
Prénom : Admin	Bof
Nom : Admin	

#### Ajouter un post

Titre :

Contenu :

Figure 2 – Messages postés

### 3.5 add.php

La page Web correspondant à ce fichier permet de poster un nouveau message.

Une session est démarrée, puis un test d'existence des variables de session est effectué (*l'e-mail* et le mot de passe).

Si le test est concluant, il y a :

- récupération de l'identifiant de l'auteur du message posté grâce, par exemple, à la requête suivante :

```
<?php
$query = 'SELECT user_id FROM user WHERE email=\' ' . $_SESSION['email'] . ' \
' AND password =\' ' . $_SESSION['password'] . ' \'';
?>
```

- **si** les données du formulaire d'ajout sont correctement renseignées, il y a :
  - ajout du message posté en base de données avec une requête du genre (la fonction `date(...)` permettant de générer la date du jour, la variable `$user_id` étant l'identifiant en base de données de l'utilisateur connecté qui poste le nouveau message) :

```
<?php
$query = 'INSERT INTO post (title, message, post_date, user_id) VALUE
S (\'' . $_POST['title'] . '\', \'' . $_POST['message'] . '\', \'' . date
('Y-m-d') . '\', ' . $user_id . ' \'';
?>
```

- redirection vers `listpost.php` ;
- **sinon** :
  - affichage d'un message d'erreur ;

**sinon** :

- affichage d'un message d'erreur.

## 4 Autres fonctionnalités

Quelques indications pour des fonctionnalités supplémentaires :

- contrôler les données des formulaires (en JavaScript) avant leur envoi au serveur ;
- détecter si *l'e-mail* n'est pas au bon format ;
- vérifier que le pseudonyme ne soit pas déjà présent en base (ajout d'une contrainte d'unicité sur le pseudonyme), que *l'e-mail* ne soit pas déjà présent non plus ;
- générer un nouveau mot de passe en cas d'oubli ;
- etc.