1 Introduction

Dans ce chapitre nous allons aborder plus précisément le développement de notre application. Les fonctions et concepts de base détaillés dans le chapitre précédent seront exploités dans les pages suivantes pour aboutir à la réalisation de notre interface avec ses différentes fonctionnalités :

- les accès sécurisés pour définir les droits des utilisateurs,
- la gestion du carnet d'adresses,
- les paramétrages du carnet d'adresses,
- la gestion des mots de passe,
- la gestion des administrateurs.

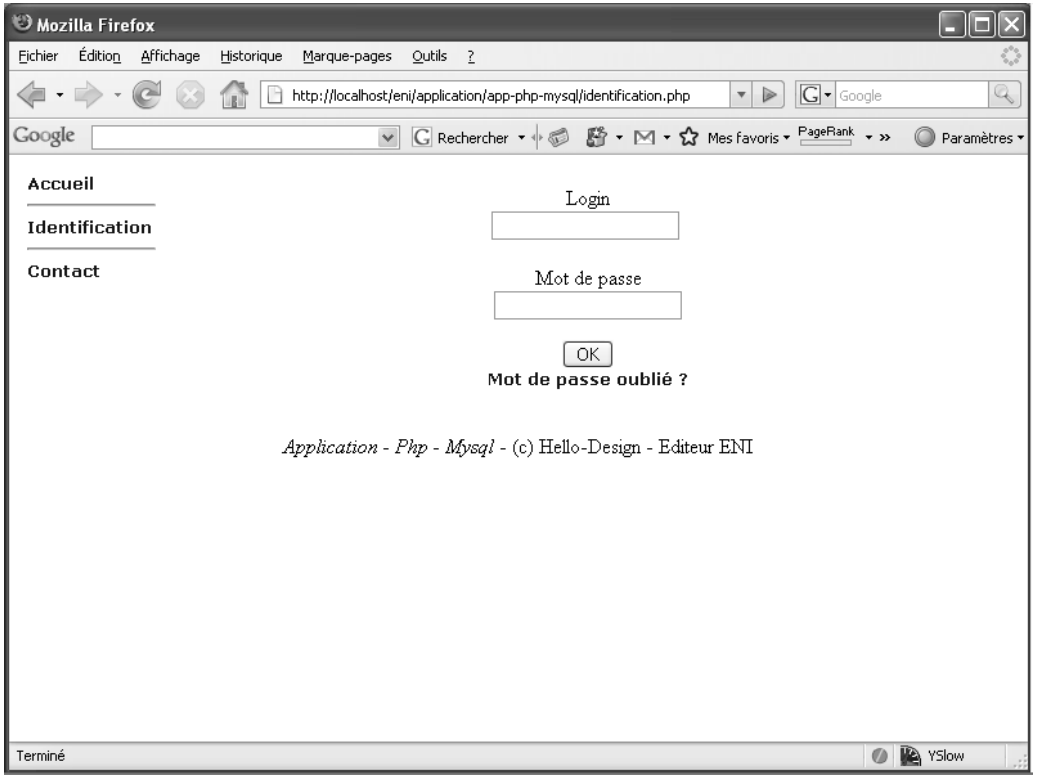
2 Connexion à un compte

2.1 L'identifiant

Comme nous l'avons vu dans le chapitre La préparation du développement, nous pouvons accéder à certaines parties du site par l'intermédiaire d'une interface. L'identifiant permet d'avoir un filtrage simple mais utile pour accéder à certaines pages du site internet.

Nous allons réutiliser ce principe pour permettre à un utilisateur d'accéder à la partie administration du site (admin) tout en gardant la possibilité pour ce même utilisateur de bénéficier comme tous les autres utilisateurs de la gestion de son propre carnet d'adresses.

Dans l'exemple, nous allons accéder à notre espace privé à partir d'un formulaire, comme ci-après :



Nous proposons dans ce formulaire deux champs à remplir : un champ Login (identifiant) et un champ Mot de passe. Ces deux champs sont obligatoires et permettront au visiteur d'accéder à son compte (si les champs ont été remplis correctement).

Si les valeurs ne sont pas bonnes, nous afficherons un message signalant le problème qui se présentera comme ceci :

Fichier indentification.php

```
<?php
if(isset($_GET['erreur']) && ($_GET['erreur'] == "login"))
{
echo "login ou mot de passe incorrect";
}
if(isset($_GET['erreur']) && ($_GET['erreur'] == "intru"))
{
echo "Echec d'identification !!!";
}
if(isset($_GET['erreur']) && ($_GET['erreur'] == "session"))
{
echo "Session expirée";
}
?>
```

Un peu plus loin dans ce chapitre, section Connexion à un compte - L'e-mail, le mot de passe, nous ajouterons un lien pour le cas où le visiteur aurait oublié ses identifiants afin qu'il puisse les récupérer (nous lui ferons parvenir ces éléments sur son adresse mail pour être sûrs qu'il s'agit de la bonne personne qui demande ces identifiants).

2.2 La connexion

Avant d'utiliser notre application, rappelons certains points concernant le contrôle des champs :

Login : l'identifiant saisi va être gardé en l'état, sans aucune conversion à part la conversion des caractères au format HTML. Nous réaliserons cette manipulation avec la fonction **HTML-ENTITIES()** que nous avons étudiée précédemment.

Mot de passe : le mot de passe va être protégé par la fonction **md5**. Ainsi nous allons envoyer une valeur cryptée ce qui permettra d'assurer la confidentialité auprès des personnes qui possèdent un compte.

La fonction md5 est une chaîne de caractères (**string**) qui utilise un algorithme et retourne un résultat sur 32 caractères hexadécimaux.

En voici un petit exemple.

motdepasse/md5.php

```
<?php
$exemple="Editions ENI2007";
echo md5($exemple)."<br />";
?>
```

md5 Calcule le MD5 d'une chaîne.

Voici le résultat que nous obtenons :

Lorsque le visiteur va se connecter, nous devons effectuer un certain nombre de vérifications avant de pouvoir lui donner accès à son compte.

Nous allons d'abord vérifier s'il possède un compte, avec tous les contrôles des caractères spéciaux comme ci-après.

Nous allons avoir besoin du fichier fct.inc.php qui contient une fonction qui va nous permettre de générer une clef de repérage.

login.inc.php

```
<?php
include "include/fct.inc.php";
?>
```

Nous testons si le champ n'est pas vide et contient bien une valeur avant de continuer. Si ce n'est pas le cas, nous revenons à l'écran précédent et nous affichons un message d'erreur.

Nous utilisons la fonction **mysql_real_escape_string** :

mysql_real_escape_string Protège les caractères spéciaux d'une chaîne pour l'utiliser dans une requête SQL.

```
<?php
if (isset($_POST['login']) && !empty($_POST['login']) )
{
$login=htmlentities($_POST['login'], ENT_QUOTES,'UTF-8');
$login=mysql_real_escape_string($login);
$password=htmlentities($_POST['password'], ENT_QUOTES,'UTF-8');
$password=md5($password);
$password=mysql_real_escape_string($password);
}
else
{
  header("Location: identification.php?erreur=login") ;
}
?>
```

Maintenant, nous allons vérifier si les caractères saisis sont bien présents dans la base de données. Nous devons avant tout convertir le login pour être sûr qu'il est composé de caractères HTML pour éviter toutes les attaques possibles.

Nous allons aussi chiffrer le champ mot de passe car les mots de passe de la base de données sont cryptés pour éviter que les comptes soient utilisés par des personnes auxquelles ils n'appartiennent pas.

```
<?php
$sql="SELECT * FROM user WHERE login='$login' AND password='$password' ";
$req=mysql_query($sql);
if(! $req ) echo ('Requête invalide : ' . mysql_error());
if (mysql_num_rows($req)==0)  header("Location:identification.php?erreur=login");
?>
```

Si nous ne trouvons pas le compte, nous revenons sur le formulaire d'authentification en signalant le problème.

Si nous avons trouvé le compte dans la base de données, nous modifions la clef de repérage qui nous permet d'être certains d'identifier la bonne personne lors des différentes manipulations et navigations dans le site Internet. Nous enregistrerons en même temps la date de passage qui peut nous être utile pour plus tard.

```
<?php
$idclef=recup_clef();
$date=DATE("Y-m-d");
$sql="UPDATE user SET idclef='$idclef',date_lastpass='$date'  WHERE login='$login'
AND password='$password' ";
$query = mysql_query($sql) OR die("Mise à jour de la clé impossible : <br />".
mysql_error());
?>
```

La clef est déterminée par un appel à la fonction recup_clef() que nous avons vue dans la partie précédente. Cette clef change tout le temps permettant ainsi de ne pas avoir de doublon dans notre site Internet.

Après avoir effectué les vérifications d'usage, nous chargeons quelques informations que nous plaçons dans une variable SESSION.

```
<?php
$sql="SELECT * FROM user WHERE login='$login' AND password='$password'
AND idclef='$idclef'  ";
$req=mysql_query($sql);
if(! $req ) echo ('Requête invalide : ' . mysql_error());
$row = mysql_fetch_object($req);

        if (mysql_num_rows($req)!=0)
        {
         session_start();
         $_SESSION['login'] = $row->login;
         $_SESSION['idclef'] = $row->idclef;
         $_SESSION['niveau'] = $row->niveau;
         $_SESSION['ip'] = $_SERVER['REMOTE_ADDR'];
         $_SESSION['iduser'] = $row->id;
         $destination=$row->page;
         header("Location:$destination");
        }
        else
        {
             header("Location:identification.php?erreur=intru");
        }
?>
```

Si la personne est correctement identifiée, nous allons lui permettre de continuer en accédant à la partie complète du menu.