

# TDM08 de Technologie Web: PHP3

## ASI4 - INSA Rouen

### CORRECTION

## Contexte

Un concepteur de sites web a commencé à développer une application de partage de connaissances. Les visiteurs du site peuvent ajouter des URL d'autres sites avec un descriptif. Ces URL sont visibles à partir de la page `index.html`. Chaque URL peut être consultée par l'intermédiaire d'une `iframe`, ce qui permet de revenir facilement à la liste et de naviguer ainsi entre les ressources partagées.

Vous récupérerez ce code sur Moodle. Il est constitué de 3 pages PHP :

- `index.php` affiche la liste et le formulaire d'ajout ;
- `iframe.php` affiche le site ;
- `init.php` crée et initialise la base de données.

Vous trouverez également 3 classes : `message.class.php` qui gère les objets messages, `db.class.php` qui gère l'abstraction BD et `message.db.class.php` qui gère la persistance BD des objets messages.

Votre objectif est de mettre ce développement dans un framework que vous allez vous-même créer. Attention, certains sites interdisent l'affichage en `iframe`, notamment `google.com`. Choisissez donc bien vos exemples.

Ce TP est volontairement long car il clôture la partie PHP. Le découpage en étapes devraient vous permettre d'itérer dans sa construction en maintenant une version fonctionnelle.

Vous trouverez ci-dessous un visuel vous permettant de vous donner une idée du résultat final.

Partage de ressource Ajouter Connexion

#	Ressource	URL	Actions
1	Symfony, l'un des framework les plus utilisés	<a href="http://symfony.com/">http://symfony.com/</a>	
2	L'un des premiers frameworks PHP, inspiré de RAIL	<a href="http://cakephp.org/">http://cakephp.org/</a>	
3	Code Igniter, l'une des références	<a href="http://ellislab.com/codeigniter">http://ellislab.com/codeigniter</a>	
4	FuelPHP un framework qui monte en ce moment	<a href="http://www.fuelphp.com/">http://www.fuelphp.com/</a>	

### Ajoutez votre ressource

Email

URL

Descriptif

Décrivez rapidement la ressource partagée.

### Connexion

Mot de passe

Avec le mot de passe administrateur, vous pourrez modifier et supprimer des ressources.

## Première étape : routage

Dans un premier temps vous mettrez uniquement en place le « routage » de votre framework.

- La page avec la liste des sites sera accessible sur l'URL directe /.
- La page contenant l'iframe de chaque site mémorisé sera disponible sur /site/ID (ID étant l'identifiant du site correspondant en BD).
- Pour toutes les autres URL, vous afficherez une page 404.

Vous utiliserez la ré-écriture d'URL. Vous créerez également une classe `route` chargée de gérer la liste des routes autorisées. Celle-ci variera bien évidemment en fonction des ressources ajoutées.

Pensez dès à présent et durant tout le TDM à bien organiser votre code dans des répertoires aux noms appropriés (`classes`, `db`, `routes`, `templates`, etc.).

## Seconde étape : charte graphique et templates

Le concepteur vous a fourni deux pages en HTML pour charter le développement. Vous commencerez donc par appliquer les CSS aux pages déjà fournies.

La liste des pages à afficher étant amenée à évoluer régulièrement, vous devrez mettre en place un système de templates pour que le concepteur puisse continuer à les faire évoluer sans votre intervention. Ainsi, vous définirez un template pour les pages non trouvées (404), un pour la page d'accueil du site et un pour les pages référencées (iframe contenant une instance de la liste de sites).

Optionnel : ajoutez une classe `tpl` factorisant les éléments communs à tous les templates (ex : formatage des dates) et mettant en place un espace de nommage.

## Troisième étape : gestion des erreurs

Ajoutez la gestion des exceptions pour remonter les erreurs au client dans tous les cas. Pensez bien à proposer un template pour l'affiche de cet erreur.

## Quatrième étape : administration (authentification)

En se connectant avec un mot de passe stocké dans un fichier `config.php`, les utilisateurs peuvent supprimer un enregistrement de la base de données.

En étant connectés, les utilisateurs peuvent également modifier un élément de la BD.

### Correction

```
.htaccess
RewriteEngine On
# ATTENTION si dans un dossier /~dossier_utilisateur
# RewriteBase /URL_JUSQUA_L_URL_DU_DOSSIER_DU_HTACCESS
RewriteBase /~paucheta/TW/PHP-TDM08/
RewriteCond %{REQUEST_FILENAME} !-f
RewriteCond %{REQUEST_FILENAME} !-d
RewriteRule (.*) index.php?arg=$1 [QSA,L]

index.php
<?php
header('Content-Type: text/html; charset=utf-8');
// Initialisation du Framework
include('conf.php');
include('classes/tpl.class.php');
include('classes/route.class.php');
include('classes/db.class.php');
// Chargement des classes métiers
include('classes/message.class.php');
include('classes/message.db.class.php');

try {
    // Initialisation du moteur de template
    $tpl = new asi4frame\template();

    // Initialisation de la base de données
    if (!file_exists(DB_PATH)) {
        asi4frame\db::Connect('sqlite:'.$DB_PATH);
        messageDb::CreateTable();
        messageDb::LoadSample();
    }
}
```

```

} else {
    asi4frame\db::Connect('sqlite:'.$DB_PATH);
}
// gestion de la connexion
session_start();
if (!empty($_POST['auth']) && sha1($_POST['auth'].SALT)==ADMIN_PASS) {
    $_SESSION['auth'] = true;
}
$tpl->isauth = !empty($_SESSION['auth']);

// Initialisation du routage
asi4frame\route::init(array('home','site')); // definition des routes autorisées
asi4frame\route::setDefaultRoute('home'); // definition des routes autorisées

// Execution du routage
$maroute = asi4frame\route::getCurrentRoute();
include('routes/'.$maroute.'.route.php');
} catch (Exception $e) {
    include('templates/erreur.tpl.php');
}

```

### conf.php

```

<?php
define('DB_PATH', 'db/messages.sqlite');
define('ADMIN_PASS', '4def8264d87d22a4432b5b1968ecf819ec5b0f2b'); // stocké haché
define('SALT', '1grosGrainDeSel');

// pour calculer un mot de passe haché
// echo sha1('motdepasse'.SALT);

```

### db.class.php

```

<?php
namespace asi4frame;

class db {
    private static $conString;
    private static $conActive = false;
    private static $db;
    private static $preparequery = false;
    public static function Connect ($constring = false) {
        if ($constring) {
            db::$conString = $constring;
        }
        try {
            self::$db = new \PDO(db::$conString);
            self::$db->setAttribute(\PDO::ATTR_ERRMODE, \PDO::ERRMODE_EXCEPTION);
        } catch (Exception $e) {
            self::$conActive = false;
            throw $e;
        }
        self::$conActive = true;
    }

    public static function OnlyConActive () {
        if (!self::$conActive) {
            throw new \Exception("DB non connectée");
        }
    }

    public static function Query ($sql) {
        self::OnlyConActive();
        try {
            return self::$db->query($sql);
        } catch (Exception $e) {
            throw $e;
        }
    }

    public static function Prepare ($sql) {
        self::OnlyConActive();
        try {
            self::$preparequery = self::$db->prepare($sql);
        } catch (Exception $e) {
            throw $e;
        }
    }

    public static function Execute ($params) {
        self::OnlyConActive();
        if (!self::$preparequery) {
            throw new \Exception("On ne peut pas executer avant d'avoir préparé sa query");
        }
    }
}

```

```

        try {
            self::$preparequery->execute($params);
            return self::$preparequery->fetchAll();
        } catch (Exception $e) {
            throw $e;
        }
        self::$preparequery = false;
    }
}

message.class.php
<?php
class Message {
    private $id=NULL;
    private $timestamp;
    private $email='';
    private $texte='';
    private $url='';

    public function __construct() {
        $this->timestamp = time();
    }

    public function getDate($format = "%A %d %B %Y %T") {
        return strftime($format, (int)$this->timestamp);
    }

    public function getDateSql() {
        return $this->getDate("%Y-%m-%d %H:%M:%S");
    }

    public function setDate($d) {
        $this->timestamp= (int) $d;
    }

    public function getEmail() {
        return $this->email;
    }

    public function setEmail($e) {
        if (filter_var($e, FILTER_VALIDATE_EMAIL)) {
            $this->email=$e;
        } else {
            throw new \Exception("Email non valide");
        }
    }

    public function getTexte() {
        return $this->texte;
    }

    public function setTexte($t) {
        $this->texte=filter_var($t, FILTER_SANITIZE_STRING);
    }

    public function getId() {
        return $this->id;
    }

    public function setId($id) {
        $this->id = (int) $id;
    }

    public function getUrl() {
        return $this->url;
    }

    public function setUrl($u) {
        if (filter_var($u, FILTER_VALIDATE_URL)) {
            $this->url=$u;
        } else {
            throw new \Exception("URL non valide");
        }
    }
}

message.db.class.php
<?php
class MessageDb extends mysqli {
    public static function CreateTable () {

```

```

static::Query ("CREATE TABLE IF NOT EXISTS messages (
message_id INTEGER PRIMARY KEY,
message_date TEXT NOT NULL,
message_email TEXT NOT NULL,
message_texte TEXT NOT NULL,
message_url TEXT NOT NULL
)");
}

public static function LoadSample() {
    $allm = array(
        array('email'=>'olivier@sb.am', 'texte'=>"Symfony, l'un des framework les
plus utilis ", 'url'=>"http://symfony.com/"),
        array('email'=>'olivier@sb.am', 'texte'=>"L'un des premiers frameworks PHP,
inspir  de RAIL", 'url'=>"http://cakephp.org/"),
        array('email'=>'olivier@sb.am', 'texte'=>"Code Igniter, l'une des r f rences",
'url'=>"https://www.codeigniter.com/"),
        array('email'=>'olivier@sb.am', 'texte'=>"FuelPHP un framework qui monte en
ce moment", 'url'=>"http://www.fuelphp.com/")
    );
    foreach ($allm as $m) {
        $message = new Message();
        $message->setEmail($m['email']);
        $message->setTexte($m['texte']);
        $message->setUrl($m['url']);
        static::Create($message);
    }
}

public static function Create($message) {
    static::Prepare("INSERT INTO messages (message_date, message_email,
message_texte, message_url) VALUES (:date, :email, :texte, :url)");
    static::Execute(array(':date'=>$message->getDateSql(),
':email'=>$message->getEmail(),
':texte'=>$message->getTexte(),
':url'=>$message->getUrl()
));
}

public static function Read($id) {
    static::Prepare("SELECT * FROM messages WHERE message_id = :id");
    $liste_messages = self::Execute(array(':id'=>(int)$id));
    $row = $liste_messages[0];
    $message = new Message();
    $message->setId($row['message_id']);
    $message->setDate(strtotime($row['message_date']));
    $message->setEmail($row['message_email']);
    $message->setTexte($row['message_texte']);
    $message->setUrl($row['message_url']);
    return $message;
}

public static function Update($message) {
    static::Prepare("UPDATE messages SET message_date = :date, message_email = :
email, message_texte = :texte, message_url = :url WHERE message_id = :id");
    static::Execute(array(':date'=>$message->getDateSql(),
':email'=>$message->getEmail(),
':texte'=>$message->getTexte(),
':url'=>$message->getUrl(),
':id'=>$message->getId()
));
}

public static function Delete($id) {
    static::Prepare("DELETE FROM messages WHERE message_id = :id");
    static::Execute(array(':id'=>$id));
}

public static function ListMessage() {
    $messages = array();
    $liste_messages = self::Query("SELECT * FROM messages ORDER BY message_id ASC");
    foreach($liste_messages as $row) {
        $message = new Message();
        $message->setId($row['message_id']);
        $message->setDate(strtotime($row['message_date']));
        $message->setEmail($row['message_email']);
        $message->setTexte($row['message_texte']);
        $message->setUrl($row['message_url']);
        $messages[] = $message;
    }
    return $messages;
}
}

```

```
}
```

### route.class.php

```
<?php
namespace asi4frame;
class route {
    private static $routesautorise = array();
    private static $defaultroute;

    public static function init ($routes) {
        if (is_array($routes)) {
            route::$routesautorise = $routes;
        } else {
            throw new \Exception("Route a besoin d'un tableau de routes pour être
                initialisé");
        }
    }

    public static function setDefaultRoute ($route) {
        if (in_array($route, route::$routesautorise)) {
            route::$defaultroute = $route;
        } else {
            throw new \Exception("La route definie par default n'est pas dans la liste
                des routes autorisés");
        }
    }

    public static function getCurrentRoute () {
        $route = array();
        // si pas de PATH_INFO = default route
        if (!isset($_GET['arg']) && isset($_SERVER['PATH_INFO'])) {
            $path = trim($_SERVER['PATH_INFO'], '/');
        } elseif (isset($_GET['arg'])) {
            $path = $_GET['arg'];
        }
        if (empty($path) && !empty(route::$defaultroute)) {
            return route::$defaultroute;
        }
        // extraction de la route du PATH_INFO
        elseif (preg_match('/^([a-z]+)\/?([0-9]*)$/', $path, $route)) {
            // vérification si la route est autorisé
            if (in_array($route[1], route::$routesautorise)) {
                return $route[1];
            }
        }
        return "404"; // route pas trouvé
    }

    public static function getCurrentArg () {
        $route = array();
        // si pas de PATH_INFO = default route
        if (!isset($_GET['arg']) && isset($_SERVER['PATH_INFO'])) {
            $path = trim($_SERVER['PATH_INFO'], '/');
        } elseif (isset($_GET['arg'])) {
            $path = $_GET['arg'];
        }
        if (preg_match('/^([a-z]+)\/?([0-9]*)$/', $path, $route)) {
            if (in_array($route[1], route::$routesautorise)) {
                return (int) $route[2];
            }
        }
        return false; // pas default
    }

    public static function show404 () {
        include('routes/404.route.php');
        exit;
    }
}
}
```

### tpl.class.php

```
<?php
namespace asi4frame;
class template {
    // Ici on pourrait mettre des methodes spécifiques pour le formatage de date par
    // exemple...
}
}
```

### 404.route.php

```

<?php
header("HTTP/1.0 404 Not Found");
// affichage du template 404
include('templates/404.tpl.php');

```

### home.route.php

```

<?php
// gestion de la suppression
if (!empty($_GET['del'])) {
    MessageDb::Delete($_GET['del']);
}

// gestion de la modification
if (!empty($_GET['edit'])) {
    $editmessage = MessageDb::Read($_GET['edit']);
    $tpl->editmessage = new ArrayObject();
    $tpl->editmessage->texte = $editmessage->getTexte();
    $tpl->editmessage->url = $editmessage->getUrl();
    $tpl->editmessage->id = $editmessage->getId();
    $tpl->editmessage->email = $editmessage->getEmail();
}

// gestion du formulaire
if (!empty($_POST['url'])) {
    // Insertion du nouveau
    try {
        $m= new Message();
        $m->setUrl($_POST['url']);
        $m->setEmail($_POST['email']);
        $m->setTexte($_POST['texte']);
        if (!empty($_POST['id'])) {
            $m->setId($_POST['id']);
            MessageDb::Update($m);
        } else {
            MessageDb::Create($m);
        }
    } catch (Exception $e) {
        // rechargement du formulaire pour la correction
        $tpl->editmessage = new ArrayObject();
        $tpl->editmessage->texte = $_POST['texte'];
        $tpl->editmessage->url = $_POST['url'];
        $tpl->editmessage->email = $_POST['email'];
        // gestion des erreurs
        $tpl->err = $e->getMessage();
    }
}

// chargement du "moteur" de template
$listedesmessages = MessageDb::ListMessage();
foreach ($listedesmessages as $m) {
    $message = new ArrayObject();
    $message->date = $m->getDate();
    $message->texte = $m->getTexte();
    $message->id = $m->getId();
    $message->url = $m->getUrl();
    $message->email = $m->getEmail();
    $tpl->messages[] = $message;
}

// affichage du template home
include('templates/home.tpl.php');

```

### site.route.php

```

<?php
// récupération du parametre
$id=asi4frame\route::getCurrentArg();
if (!$id) {
    asi4frame\route::show404();
}

// recherche du message
$message = MessageDb::Read($id);
if (!$message) {
    asi4frame\route::show404();
}

// chargement du "moteur" de template
$tpl->texte = $message->getTexte();
$tpl->email = $message->getEmail();

```

```

$tpl->date = $message->getDate();
$tpl->url = $message->getUrl();

// lancement du template
include ('templates/site.tpl.php');

```

#### 404.tpl.php

```

<html>
<head>
<title>Page non trouvée</title>
<style>
    div {
        font-size:300px;
        color:grey;
        margin-top: 100px;
        width : 100%;
        text-align: center;
    }
</style>
</head>
<body>
<div>404</div>
</body>
</html>

```

#### erreur.tpl.php

```

<html>
<head>
<title>Erreur</title>
<style>
    div {
        font-size:24px;
        color:grey;
        margin-top: 100px;
        width : 100%;
        text-align: center;
    }
    .t {
        font-size:200px;
    }
</style>
</head>
<body>
<div class="t">Erreur</div>
<div class="m"><?=$e->getMessage()?></div>
</body>
</html>

```

#### site.tpl.php

```

</DOCTYPE html>
<html lang="en"><head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
<title>Partage de ressources</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta charset="utf-8">
<link rel="stylesheet" href="../static/bootstrap.css" media="screen">
<link rel="stylesheet" href="../static/bootswatch.css">
<style>
    .bs-docs-section {
        margin-top : 3em;
    }
    iframe {
        border:0;
        margin:0;
        padding:0;
        width:100%;
        height:1000px;
    }
</style>
</head>
<body>
<div class="navbar navbar-default navbar-fixed-top">
<div class="container">
<div class="navbar-header">
<span class="navbar-brand"><?=$tpl->texte ?></span>
</div>
<div class="navbar-collapse collapse" id="navbar-main">
<ul class="nav navbar-nav navbar-right">
<li><a type="button" class="btn btn-warning" href="..">Retour à la liste</a></li>
</ul>

```



```

    </div>
  </div>
</div>
<iframe src="<?=$tpl->url ?>"></iframe>

<script src="../../static/jquery.js"></script>
<script src="../../static/bootstrap.js"></script>
<script src="../../static/bootswatch.js"></script>
</body></html>

```

home.tpl.php

```

</DOCTYPE html>
<html lang="en">
<head>
<meta http-equiv="content-type" content="text/html; charset=UTF-8">
<title>Partage de ressources</title>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<meta charset="utf-8">
<link rel="stylesheet" href="static/bootstrap.css" media="screen">
<link rel="stylesheet" href="static/bootswatch.css">
<style>
    .bs-docs-section {
        margin-top : 3em;
    }
</style>
</head>
<body>
<div class="navbar navbar-default navbar-fixed-top">
  <div class="container">
    <div class="navbar-header">
      <span class="navbar-brand">Partage de ressource</span>
    </div>
    <div class="navbar-collapse collapse" id="navbar-main">
      <ul class="nav navbar-nav navbar-right">
        <li><a href="#insert">Ajouter</a></li>
        <li><a href="#connect">Connexion</a></li>
      </ul>
    </div>
  </div>
</div>
<div class="container">

  <?php
  if (!empty($tpl->err)) { ?>
    <!-- Affichage des éventuels messages d'erreur -->
    <div class="row">
      <div class="col-lg-12">
        <div class="bs-example">
          <div class="alert alert-dismissable alert-warning">
            <button type="button" class="close" data-dismiss="alert">x</button>
            <h4>Erreur</h4>
            <p><?=$tpl->err?></p>
          </div>
        </div>
      </div>
    </div>
  </div>
  <?php } ?>

  <!-- Tables
  =====>
  <div class="bs-docs-section">
    <div class="row">
      <div class="col-lg-12">
        <div class="bs-example table-responsive">
          <table class="table table-striped table-bordered table-hover">
            <thead>
              <tr>
                <th>#</th>
                <th>Ressource</th>
                <th>URL</th>
                <th>Actions</th>
              </tr>
            </thead>
            <tbody>
              <?php foreach ($tpl->messages as $message) { ?>
                <tr>
                  <td><?=$message->id ?></td>
                  <td><?=$message->texte ?></td>
                  <td><a href="site/<?=$message->id ?>"><?=$message->url ?></a></td>

```

```

        <td>
            <?php if ($tpl->isauth) {?>
                <!-- si on est connecté uniquement --><a href='./?edit<?=$
                    $message->id ?>' class="btn btn-xs btn-primary">
                    Modifier</a> <a href='./?del<?=$message->id ?>' class
                    ="btn btn-xs btn-danger">Supprimer</a>
                <?php } ?>
            </td>
        </tr>
    <?php } ?>
</tbody>
</table>
</div><!-- /example -->
</div>
</div>
</div>
<!-- Formulaire d'ajout -->
<div class="bs-docs-section">
    <div class="row">
        <div class="col-lg-6">
            <div class="well">
                <form id="insert" method="POST" action="." class="bs-example form-
                    horizontal">
                    <fieldset>
                        <legend>Ajoutez votre ressource</legend>
                        <div class="form-group">
                            <label for="inputEmail" class="col-lg-2 control-label">Email</label>
                            <div class="col-lg-10">
                                <input class="form-control" name="email" id="email" placeholder="
                                    Email" type="text" value="<?=(!empty($tpl->editmessage->email)?
                                    $tpl->editmessage->email: '' )?>">
                            </div>
                        </div>
                        <div class="form-group">
                            <label for="inputPassword" class="col-lg-2 control-label">URL</label>
                            <div class="col-lg-10">
                                <input class="form-control" name="url" id="url" placeholder="URL"
                                    type="text" value="<?=(!empty($tpl->editmessage->url)?$tpl->
                                    editmessage->url: '' )?>">
                            </div>
                        </div>
                        <div class="form-group">
                            <label for="textArea" class="col-lg-2 control-label">Descriptif</
                                label>
                            <div class="col-lg-10">
                                <textarea class="form-control" name="texte" rows="3" id="texte"><
                                    ?=(!empty($tpl->editmessage->texte)?$tpl->editmessage->texte: ''
                                    )?></textarea>
                                <span class="help-block">Décrivez rapidement la ressource partagée
                                    </span>
                                <?=(!empty($tpl->editmessage->id)?<input type='hidden' name='id'
                                    value='.'. $tpl->editmessage->id.'/'>": '' )?>
                            </div>
                        </div>
                        <div class="form-group">
                            <div class="col-lg-10 col-lg-offset-2">
                                <button type="submit" class="btn btn-primary">Enregistrer</button>
                            </div>
                        </div>
                    </fieldset>
                </form>
            </div>
        </div>
        <div class="col-lg-5 col-lg-offset-1">
            <div class="well">
                <?php if (!$tpl->isauth) {?>
                    <form id="connect" method="POST" action="." class="bs-example form-
                        horizontal">
                        <fieldset>
                            <legend>Connexion</legend>
                            <div class="form-group">
                                <label for="inputEmail" class="col-lg-4 control-label">Mot de
                                    passe</label>
                                <div class="col-lg-8">
                                    <input class="form-control" name="auth" id="auth" placeholder="
                                        Mot de passe" type="password">
                                    <span class="help-block">Avec le mot de passe administrateur,

```

```

        vous pourrez modifier et supprimer des ressources.</span>
    </div>
</div>
    <div class="form-group">
        <div class="col-lg-8 col-lg-offset-4">
            <button type="submit" class="btn btn-primary">Connecter</button>
        </div>
    </div>
</fieldset>
</form>
<?php } else {?>
    Connecté
<?php } ?>
</div>

</div>
</div>
</div>

<footer>
    <div class="row">
        <div class="col-lg-12">
            <p>Made by <a href="http://thomaspark.me/">Thomas Park</a>. Contact him at
                <a href="mailto:hello@thomaspark.me">hello@thomaspark.me</a>.</p>
            <p>Code licensed under the <a href="http://www.apache.org/licenses/LICENSE
                -2.0">Apache License v2.0</a>.</p>
            <p>Based on <a href="http://getbootstrap.com/">Bootstrap</a>. Icons from <
                a href="http://fortawesome.github.io/Font-Awesome/">Font Awesome</a>.
                Web fonts from <a href="http://www.google.com/webfonts">Google</a>.
                Favicon by <a href="https://twitter.com/geraldhiller">Gerald Hiller</a>
            </p>
        </div>
    </div>
</footer>

</div>

<script src="static/jquery.js"></script>
<script src="static/bootstrap.js"></script>
<script src="static/bootswatch.js"></script>
</body></html>

```

## Remarques

1. Le serveur `asi-technoweb` utilisé lors de la session précédente est toujours à votre disposition et est configuré pour interpréter les scripts PHP.
2. Pour plus d'informations sur PHP : <http://www.php.net/manual/fr/>
3. Continuez à vérifier vos pages HTML et CSS sur le site du W3C (<http://validator.w3.org/>).
4. À l'issue de la séance, vous aurez accès à la correction de ce TDM au format PDF.
5. Déposez votre compte-rendu sur moodle sous la forme d'un fichier PDF de 2p. (1 feuille recto-verso) nommé `TDM08-login.pdf`, chez chacune des 2 personnes du binôme.