

Notions abordées à travers ce TP

le modèle MVC

PROJET – PHP / MYSQL

RÉALISATION D'UN SITE DE PARTAGE DE RECETTES DE CUISINE



Objectifs

L'objectif de ce TP est de remplacer le cœur de votre projet par le duo CodeIgniter & Smarty. Si vous avez bien respecté l'architecture proposée en TP, cela se fera relativement facilement.

Prérequis

Avoir fini le TP3

Travail préliminaire

Avant de commencer, il faut installer et configurer CodeIgniter. Pour cela, rendez-vous sur la page web suivante :

<http://codeigniter.com/>

Téléchargez la dernière version de CodeIgniter. Puis suivez les étapes suivantes : Installez CodeIgniter et regardez comment il fonctionne. Commencez par jouer avec le contrôleur **welcome** fourni en exemple et la vue **welcome_message.php**. Vous pouvez en complément vous référer à la documentation de CodeIgniter (http://codeigniter.com/user_guide/) et regardez plus particulièrement les sections suivantes :

- CodeIgniter URLs
- Controllers
- Views

Une fois cet aspect de CodeIgniter compris, il est temps d'intégrer Smarty. Pour cela, rendez-vous sur les pages suivantes :

https://github.com/cryode/CodeIgniter_Smarty

<http://www.smarty.net/>

Téléchargez la librairie préparée par Cryode et la dernière version de Smarty. Suivez alors les instructions d'installation fournies par Cryode. Une fois terminée, mettez à jour la version de Smarty intégrée par Cryode à l'aide de la version téléchargée sur le site de Smarty. Pour cela, il vous suffit de copier les fichiers du répertoire **libs** de la librairie Smarty dans le répertoire **application/third_party/Smarty** de CodeIgniter.

Une fois Smarty configuré (dans le répertoire **application/config/smarty**), faites quelques essais en créant une vue (dans **application/views**) et en l'appelant depuis un contrôleur (comme **controllers/welcome.php** par exemple). Tachez de transmettre à votre vue différents types des paramètres, comme des tableaux et des objets pour utiliser les différentes fonctions et modificateurs de Smarty. Pensez à vous référer à la documentation : <http://www.smarty.net/docs/en/>.

Exercice :

Affichage de la liste des catégories, des recettes d'une catégorie et du détail d'une recette

Il est maintenant temps de s'attaquer au fond du problème.

L'objectif de cette partie est d'utiliser CodeIgniter associé à Smarty pour réafficher la partie publique du site, tel qu'elles s'affichaient avant mais avec l'aide ces outils. Pour cela, vous devez procéder étape par étape.

Etape 1

Créer les modèles dont vous avez besoin pour gérer vos données et y déplacer le code correspondant. Attention, si vous souhaitez utiliser les Active Record, vous devrez ré-écrire une partie du CRUD de chaque model (http://codeigniter.com/user_guide/database/).

Vous devriez assez rapidement voir apparaître des modèles tels que : mRecettes, mIngredients, mCategories, mUnites.

Pour cette partie, seul les méthodes `get_all` et `find($id)` devraient vous suffire. Inutile de réécrire de suite le reste du CRUD.

Etape 2

Mettre en place le contrôleur ***controllers/recettes.php*** et y définir les trois méthodes dont vous avez besoin pour afficher les catégories, les recettes d'une catégorie et le détail d'une recette. Vous pouvez également définir la méthode `index` qui peut simplement faire appel à la méthode qui liste les catégories.

Une fois que cela est fait et que vous vous êtes assuré que vous accédez via le *routing* à chacune des méthodes, il ne vous reste plus qu'à les implémenter. Lors de cette étape, faites en sorte de générer l'ensemble des données dont vous avez besoin pour la vue dans un tableau associatif. En général, on le nomme ***\$data*** ou ***\$output***. Vous pourrez alors utiliser dans le contrôleur un simple `printf("<pre>%s</pre>", print_r($data, true));` pour vous assurer que les données produites sont correctes.

Etape 3

Vous êtes maintenant prêt pour afficher les données dans une vue et appeler son affichage dans le contrôleur. Pour cela, crée les fichiers dont vous avez besoin dans le répertoire ***application/views*** et appelez les via la méthode ***view*** de CI :
`$this->load->view('recettes/liste_recettes', $data);`

Enjoy !