Programmation orientée objet en PHP

Le projet

Dans ce TP nous allons reprendre une architecture en couche comme dans le module précédent. Nous retrouverons donc les couches dao, controller et view.
Nous allons ajouter à cette architecture un front controller, qui servira de point d'entrée unique à l'application. Lors la réception d'une requête HTTP, la requête sera redirigée vers le front controller, puis le front controller sollicitera à son tour un controller adéquate.
L'un des intérêts de cette architecture est qu'elle permet une conception orientée objet de l'application. Les controllers pourront donc être implémentés sous forme de classes.
Des classes utilitaires pourront venir compléter cette architecture, telles que des classes "routers" gérant les "routes" (c'est-à-dire les correspondances entre les URL(s) et les contrôleurs), des classes gérant les formulaires et leurs validations, etc...

Pour ce projet nous allons utiliser le serveur web interne de PHP. Pour démarrer notre serveur, dans un terminal nous nous plaçons dans le répertoire web de notre projet, puis nous exécuterons la commande *php -S localhost:8080 router.php*. En passant en paramètre le fichier router.php contenu dans le répertoire web, nous demanderons à notre serveur interne de passer tout d'abord par ce script quelque soit l'URL demandée. Si l'URL demandée correspond à une ressource (image, css...) alors le script router.php renverra null et l'URL sera traitée normalement. Si la ressource ne correspond pas à une ressource alors le front controller entrera en action. Le script router.php jouera ici le rôle du fichier .htaccess du serveur HTTP Apache.

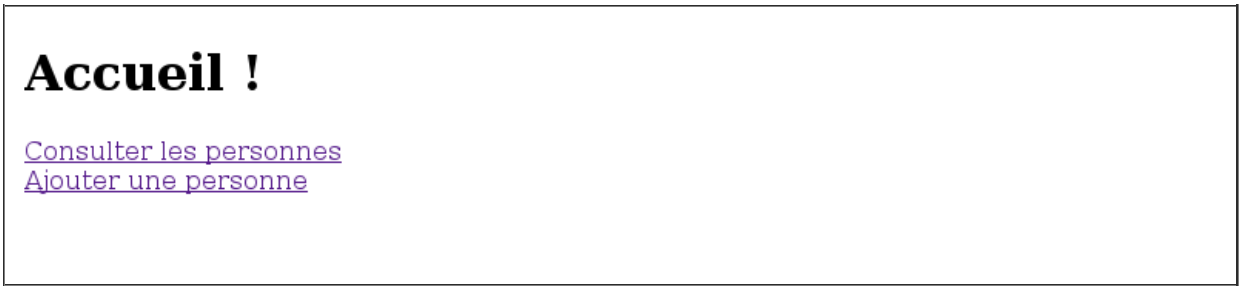
```php
<?php

// web/router.php
if (preg_match('/\.(png|jpg|jpeg|gif|css)$/', $_SERVER["REQUEST_URI"])) {
    return false;
} else {
    include 'index.php';
}
```

Le projet sera constitué de trois vues :

- une vue "accueil" avec deux liens => http://localhost:8080/
- une vue de consultation des personnes => http://localhost:8080/personnes
- une vue d'ajout d'une personne => http://localhost:8080/personnes/ajouter
- une vue d'erreur

L'accueil



La consultation des personnes :

4 Kirk James T.
Ajouter une personne

L'ajout d'une personne :

Nom :
Prénom :
Valider

La page d'erreur :

Oups...

Nous allons créer un projet nommé petite_application. Dans ce projet nous allons créer un répertoire pour chaque couche applicative (dao, controller, view), un répertoire config qui contiendra le fichier de configuration ainsi que le fichier de routing, un répertoire entity qui contiendra les entités du modèle, un répertoire web qui contiendra toutes les ressources publiques de notre application (images, css...) mais aussi le front controller (index.php), un répertoire util qui contiendra les classes utilitaires de routing, et enfin le répertoire test qui contiendra tous les tests de l'application. Nous aurons besoin de bibliothèques de test (PHPUnit, DBUnit et PHPUnit-selenium), que nous installerons via composer. L'archive composer.phar sera à la racine de notre application ainsi que le fichier composer.json.

```
{
    "require-dev": {
        "phpunit/phpunit": "4.*",
        "phpunit/dbunit": "1.*",
        "phpunit/phpunit-selenium": "1.*"
    }
}
```

Une fois les dépendances installées via composer, l'arborescence du projet sera la suivante :

Nous créerons pour ce TP deux bases de données MySQL : une base de production nommée petite_application et une base de test nommée test_petite_application.

Fin

NAVIGATION

Accueil
- Ma page

ADMINISTRATION

Connecté sous le nom « Arnaud Lemais » (Déconnexion)
POO PHP