Programmation orientée objet en PHP

Les contrôleurs

Les contrôleurs vont pouvoir récupérer des services grâce à une instance de la classe ServiceLocator. Cette classe ServiceLocator devra donc être accessible au sein de chaque contrôleur. Chaque contrôleur possédera un attribut d'instance $serviceLocator qui pointera vers l'instance unique et commune de ServiceLocator. Cette instance de ServiceLocator sera créée dans le script index.php (le front controller). Le front controller utilisera le service locator pour récupérer le service de routing :

```php
<?php

// public/index.php

include '../setup.php';

$serviceLocator = new util\ServiceLocator();
$router = $serviceLocator->get('router');
$request_uri = $_SERVER['REQUEST_URI'];
$route = $router->getRoute($request_uri);
if ($route == null) {
    include VIEW . '/index.php';
} else {
    try {
        $controller_name = $route->getController();
        $action_name = $route->getAction();
        $class_name = "controller\\$controller_name";
        $class = new ReflectionClass($class_name);
        $controller = $class->newInstance();
        $method = $class->getMethod("$action_name");
        $method->invoke($controller);
    } catch (Exception $ex) {
        include VIEW . '/errorPage.html';
    }
}
```

L'unique instance de ServiceLocator sera donc créée dans le front controller et devra être aussi attribut d'instance de chaque contrôleur. Pour cela, nous allons créer une classe abstraite qui possédera en attribut d'instance l'unique instance de ServiceLocator :

```php
<?php

// controller/MyController.php

namespace controller;

abstract class MyController {

    protected $serviceLocator;

    public function __construct() {
        global $serviceLocator;
        $this->serviceLocator = $serviceLocator;
    }

}
```

Nos contrôleurs hériteront donc de cette classe abstraite, et pourront utiliser le ServiceLocator qu'ils possèdent en attribut d'instance :

```php
<?php

// controller/PersonneController.php

namespace controller;

use controller\MyController;

class PersonneController extends MyController {

    public function addPersonne() {
        if ($_SERVER['REQUEST_METHOD'] == 'POST') {
            $nom = $_POST['nom'];
            $prenom = $_POST['prenom'];
            $personneService = $this->serviceLocator->get('personne_service');
            $result = $personneService->addPersonne($nom, $prenom);
            if ($result) {
                $this->getAllPersonnes();
            } else {
                $msg = 'Champs incorrects';
                include VIEW . '/formulaire.php';
            }
        } else {
            include VIEW . '/formulaire.php';
        }
    }

    public function getAllPersonnes() {
        $personneService = $this->serviceLocator->get('personne_service');
        $result = $personneService->getAllPersonnes();
        include VIEW . '/display_personne.php';
    }

}
```

Les tests fonctionnels des contrôleurs seront les mêmes que ceux du TP précédent.

Fin

La dao
Les classes de traitement métier
Les contrôleurs
Mes cours

ADMINISTRATION
Administration du cours
Réglages de mon profil

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