

Travaux dirigés n° 4

Programmation MVC

Exercice 1 (Les actualités : la modélisation)

Nous souhaitons développer une application web qui permet d'afficher et gérer une liste d'actualités créées par des utilisateurs. Lorsqu'un visiteur arrive sur le site, les dernières actualités sont affichées. Les autres sont accessibles via une barre de navigation. Un visiteur peut créer un compte et/ou se connecter. Un utilisateur peut ajouter de nouvelles actualités, modifier ou supprimer ses actualités. L'administrateur doit pouvoir gérer les utilisateurs.

1. Donnez les cas d'utilisation.
2. Donnez les diagrammes de navigation.

Exercice 2 (Le modèle MVC)

Soit le script PHP suivant :

```
<!DOCTYPE html>
<html lang="fr">
<head> ... </head>
<body>
    <h1> Chouette ! Des actualités </h1>
<?php
$BD = new PDO("mysql:host=localhost;dbname=news;charset=UTF8", "root", "");
if($request = $BD->query("SELECT_*_FROM_news")) {
    while($news = $request->fetch(PDO::FETCH_ASSOC)) {
        echo <<<HTML
        <div>
            <h2>{$news['new_title']}</h2>
            <p>{$news['new_content']}</p>
            <b>{$news['new_date']}</b>
        </div>
    HTML;
    }
}
?>
</body>
</html>
```

- 1°) Identifiez dans ce fichier les différentes parties correspondantes au modèle MVC.
- 2°) Réécrivez ce code dans trois fichiers : `controller.php`, `view.php` et `NewsModel.php`.

Exercice 3 (Les actualités au pays du MVC : l'application)

Nous reprenons maintenant l'application des actualités que vous avez modélisée dans l'exercice 1.

- 1°) Quelles modèles sont nécessaires et quelles méthodes contiennent-ils ?
- 2°) Donnez toutes les vues nécessaires dans l'application. Pour chacune, indiquez son contenu.
- 3°) Combien de templates sont nécessaires ? Quelles données doit-on y placer et quels sont les paramètres éventuels ?

Nous souhaitons structurer les routes à l'aide de deux paramètres : le mode correspond au contrôleur et l'action à la méthode.

4°) Détaillez toutes les routes nécessaires pour votre application. Nous supposons que toutes les données nécessaires sont passées dans l'URL (par exemple : `index.php?mode=toto&action=titi`).

5°) Quels contrôleurs sont nécessaires ? Pour chacun, donnez leurs méthodes.

6°) Proposez une arborescence pour votre application et expliquez le contenu de chaque répertoire. Vous préciserez quelles parties doivent être publiques.

7°) Quel est le rôle du routeur ? Quelles actions doivent être effectuées ?

8°) Donnez le script complet `autoload.php`.

9°) Donnez le script complet du routeur.

Annexes - Manuel PHP

Fonction **file_exists** : vérifie si un fichier existe.

```
bool file_exists ( string $filename )
```

Fonction **method_exists** : vérifie si une méthode existe pour un objet ou une classe.

```
bool method_exists ( mixed $object , string $method_name )
```

Fonction **ob_start** (version simplifiée) : enclenche la temporisation de sortie.

```
bool ob_start( void )
```

Fonction **ob_get_contents** : retourne le contenu du tampon de sortie.

```
string|false ob_get_contents( void )
```

Fonction **ob_end_clean** : détruit le tampon et stoppe la temporisation.

```
bool ob_end_clean( void )
```

Fonction **spl_autoload_register** : enregistre une fonction

```
bool spl_autoload_register ( ?callable $callback, bool $throw, bool $prepend )
```