

Exercice : Développement d'un site internet sur l'apprentissage du golf en utilisant PHP et le modèle MVC

- Créer une architecture ressemblant à :
 - `/models/[UserModel.php, PostModel.php, ...]`
 - `/controllers/[user_controller.php, post_controller.php, ...]`
 - `/views/[layout.phtml, user_login.phtml, user_register.phtml, post_list.phtml, post_add.phtml, ...]`
 - `/config/routing.php` // fichier contenant les routes de l'application dans un tableau
 - `index.php` // El Patron
- Créer une base de données MySQL pour stocker les informations sur les utilisateurs et les étapes de l'apprentissage du golf.
 - Tables : **user** (name, email, password, role_id, created_at), **role** (name), **post** (title, content, user_id, created_at)
 - 2 types de **user** : User et Admin
- Créer une classe de connexion à la base de données en utilisant PDO de manière sécurisée (try/catch).
- Créer un fichier **routing.php** pour gérer les requêtes HTTP. (cf. tableau de routes ['register' => 'register_controller.php'])
- Créer une classe **Modèle(.php)** par table pour gérer les données.
- Créer des fichiers **Contrôleur(.php)** pour gérer les actions de l'utilisateur et afficher les données dans les vues.

Créer les pages suivantes en utilisant les classes modèle, vue et contrôleur :

- **Page d'accueil** : Cette page doit afficher une description de l'apprentissage du golf et une vue d'ensemble des étapes.
- **Page d'authentification et d'inscription** : Cette page doit permettre à l'utilisateur de s'inscrire en renseignant son email et son mot de passe. Elle doit également permettre à l'utilisateur de se connecter en utilisant son email et son mot de passe.
- **Page compte utilisateur** : Cette page doit permettre à l'utilisateur de consulter et de modifier ses informations personnelles, telles que son email et son mot de passe. **CRUD** Juste pour le **User** connecté
- **Page d'administration (Back-Office)** : Réservé à l'**Admin**
 - Cette page doit permettre à l'administrateur de visualiser :
 - le nombre d'utilisateurs inscrits sur l'année en cours

- la répartition des créations de comptes utilisateurs par mois sur l'année en cours (en pourcentage).
- Créer une vue commune pour toutes les pages web (`layout.phtml`) en incluant un header avec une bannière et une barre de navigation, une barre de golf, et un footer avec les mentions légales en lien.

Cette classe doit respecter les principes SOLID et les bonnes pratiques PSR-FIG (1, 12).

Tester les différentes pages pour s'assurer qu'elles fonctionnent correctement et respectent les contraintes énoncées.

Documenter le code pour aider les futurs développeurs à comprendre les différentes classes et les principes utilisés dans le développement du site internet.