


| | | |
|---|---|-----------|
|  | Université de Corse - Pasquale PAOLI | |
| | Diplôme : Licence SPI 2ème année parcours Info | 2024-2025 |
| | Module : Programmation Web Back End TP N°4: Mini-projet Application MVC Gestion d'un club de randonnées ou projet libre Enseignants : Evelyne Vittori- Ghinevra Comiti | |

Objectif du TP

Mettre en pratique l'ensemble des concepts étudiés dans le cadre des chapitres 2 et 3 du cours en développant un site structuré selon une architecture MVC OO(avec typage explicite des variables, attributs, paramètres et résultats de méthodes).

Modalités de rendu

Le travail est à réaliser de manière individuelle.

La date de rendu est fixée au 27 mars.

Vous devrez déposer sur l'ENT dans l'onglet travaux, un **fichier archive** contenant l'ensemble des fichiers de votre site.

Description générale du TP

Le TP consiste à créer un site de gestion d'un club de randonnées mais vous avez tout à fait la **possibilité de choisir un autre thème plus personnel** (exemple : club sportif ou autre ...) dans la mesure où vous respectez les contraintes suivantes :

- Base de données comportant au minimum 3 tables
- Fonctionnalités similaires en complexité aux fonctionnalités demandées pour la gestion du club de randonnée

Vous devrez organiser votre code selon le pattern d'architecture MVC conformément à l'exemple accessible sur l'ENT **exempleMVCCH3OORoutes.zip** étudié lors du TP3.

Dans le cas de l'application de randonnées, le club est composé de membres, organise des randonnées pédestres et permet d'enregistrer la participation des membres aux randonnées proposées.

L'arborescence du site et l'organisation des pages sont laissées libres, de même que la présentation visuelle des pages et le choix des modalités des interactions avec l'utilisateur (zones de texte, cases à cocher, zones de liste , ect...).

Vous êtes également libres d'ajouter des champs dans les tables si vous le souhaitez.

Vous pouvez aussi ajouter des fonctionnalités supplémentaires.

De même au niveau de l'architecture, vous êtes tout à fait libre de l'améliorer sous réserve de respecter les principes du MVC . Vous pouvez très bien définir plusieurs fichiers contrôleurs et modifier les routes en conséquence. Vous pouvez aussi utiliser .htaccess pour gérer les routes avec des URLs plus propres comme l'avait suggéré Papillon mais ce n'est pas un élément essentiel.

N'oubliez pas que si vous utilisez des fonctionnalités non vues en cours, vous devrez absolument les expliquer et les justifier.

Fichier readMe.pdf :

Si vous optez pour une application personnelle (ou si vous ajoutez des fonctionnalités ou des tables supplémentaires dans l'application de randonnées), vous devrez rédiger un petit texte (fichier readMe.pdf) expliquant brièvement l'objectif et les fonctionnalités de votre site ainsi que la description de votre base de données. Ce fichier figurera dans le répertoire config et sera intégré à l'archive de rendu du projet.

Base de données

La base de données « club » est composée des trois tables suivantes (les clés primaires sont soulignées et les clés étrangères sont en italique) :

Table MEMBRE (liste des membres du club)

| nom du champ | type | commentaires |
|---------------|------------|------------------|
| <u>pseudo</u> | varchar 15 | pseudo du membre |
| age | float | age du membre |

Table RANDONNEE (liste des randonnées)

| nom du champ | type | commentaires |
|-----------------|----------------------|------------------------|
| <u>numRando</u> | int (auto incrément) | numéro de la randonnée |
| titre | varchar 20 | titre de la randonnée |
| dateDep | date | date de la randonnée |

Table PARTICIPATION (liste des participations des membres aux randonnées)

| nom du champ | type | commentaires |
|-----------------|------------|--|
| <u>numRando</u> | int | numéro de la randonnée |
| <u>pseudo</u> | varchar 15 | pseudo du membre participant à la randonnée ayant pour numéro numRando |

Fonctionnalités à implémenter et contraintes

Votre site devra :

1. Être construit selon une **architecture MVC fonctionnelle ou orientée Objet** de manière similaire à ce qui est présenté dans les exemples du cours exempleMVCCH2 et exempleMVCOOCH3.
2. Votre code devra utiliser les déclarations explicites de types pour les paramètres, variables et résultats de fonctions et méthodes.
3. Implémenter les **fonctionnalités** suivantes (ou des fonctionnalités similaires en cas de site « personnel ») :
 - Gestion des membres du club :
ajout et affichage de la liste des membres
 - Gestion des randonnées :
ajout, affichage de la liste, recherche par titre
 - Gestion des participations aux randonnées :

- ajout d'une participation (le membre et la randonnée choisie seront sélectionnés dans des zones de listes ou des cases à cocher comme par exemple sur la figure suivante)

Figure 1: Exemple de formulaire de saisie d'une participation à une ou plusieurs randonnées

- suppression d'une participation
- recherche de la liste des participants à une randonnée donnée (sélection de la randonnée dans une liste par exemple)
- recherche de la liste des randonnées auxquelles a participé un membre donné (recherche à partir de son pseudo)

Conseils méthodologiques

- Implémentez les fonctionnalités les unes après les autres : Commencez par une fonctionnalité, testez-la et ne passez à la suivante que lorsque votre code fonctionne correctement.
- Les fonctionnalités liées à la gestion des membres et des randonnées ne devraient pas vous poser de problème, cela correspond aux fonctionnalités que vous avez déjà réalisé dans le TP2.
- La gestion des participations aux randonnées est un peu plus compliquée car elle est liée à plusieurs tables. Ne l'abordez qu'après avoir fini les membres et les randonnées.

Critères d'évaluation

- Qualité et Structuration du code
- Qualité et pertinence des commentaires (explications de vos choix) dans votre code.
Ce point est très important car il permettra d'évaluer votre niveau de compréhension du code produit. Il ne s'agit pas de commenter chaque ligne de code mais d'en expliquer les points spécifiques.
En particulier, si vous utilisez des instructions ou fonctions particulières non vues en cours, il est indispensable que vous justifiez leur emploi par des commentaires dans votre code.
- Soins apportés à la présentation et l'ergonomie de votre site. Nous n'attendons pas une présentation artistique mais simplement une présentation soignée et agréable qui facilite la saisie des informations par l'utilisateur