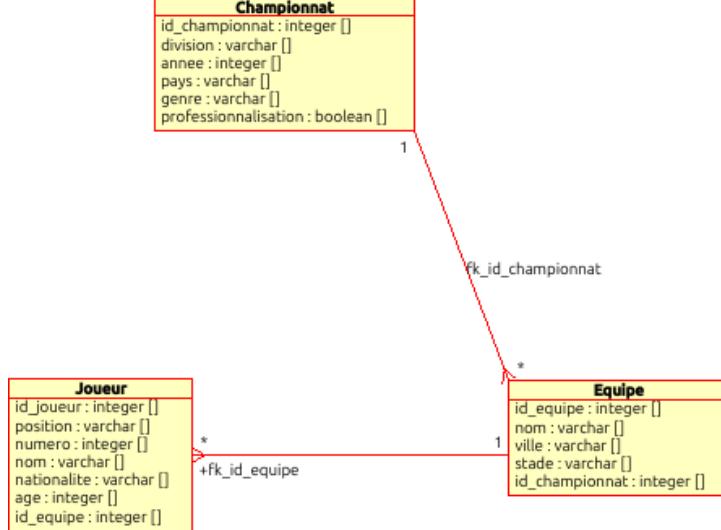


Contrôle 5

TP Football

L'application «TPFootball » est basée sur le diagramme de relation d'entité suivant :



En vous basant sur l'exemple Atelier_core_PHP, je vous demande de :

- Créer les tables en BDD (SQLite) avec PHP
- Créer les Modèles correspondant avec leur Getter et Setter méthodes
- Créer les Contrôleurs de ces modèles avec un CRUD
- Créer des contrôleurs gérant les balises HTML, les Formulaires et les Styles.
 - Créer les Vues pour chaque Modèle avec un CRUD dans un dossier « /Views/ » contenant les fichiers championnat.lame.php, joueur.lame.php et equipe.lame.php
 - Implémenter les CRUD dans les Vues grâce à des Formulaires
 - Créer un Menu de navigation en utilisant HtmlController dans « /Routes/Web.php » que l'on peut appeler avec « \$web = new Web(); » dans les pages de vue (« Routes » est un nouveau dossier dans la racine du serveur)
 - Ajouter un champ « photo » aux Joueurs et un « blason » aux Equipes et Championnats qui permet d'associer et d'afficher une photo (ou blason) à chaque instance

The screenshot shows the TP Football application interface. At the top, there is a navigation bar with links: Accueil, Championnat, Joueur, and Equipe. Below the navigation bar, there are two main sections: 'Créer un Joueur:' and 'Afficher toutes les joueurs:'.

Créer un Joueur: This section contains a form for creating a player. It includes fields for Position du Joueur (Position du Joueur), Numero du Joueur (Numero du Joueur), Nom du Joueur (Nom du Joueur), Nationalite du Joueur (Nationalite du Joueur), Age du Joueur (Age du Joueur), and a dropdown for Equipe du Joueur (Equipe du Joueur). A file input field for Entrer la Photo du Joueur (Entrer la Photo du Joueur) is also present. A 'Créer' button is at the bottom of the form.

Afficher toutes les joueurs: This section displays a list of players. It shows a table with columns: position, numero, nom, nationalite, age, and id_equipe. One player is listed: position = Milieu, numero = 10, nom = Michel Platini, nationalite = Française, age = 24, and id_equipe = 1. Below the table, there is a thumbnail image of Michel Platini.

- Créer une page spécifique (par exemple `equipe_joueur.lame.php`) qui réalise une jointure/vue de deux tables (Joueur + Equipe) affichant chaque joueur avec son équipe associée
 - Rendre toutes les classes PHP accessibles grâce à `Autoload.php` et « `namespace` »
 - Au sein de MAMP ou toute autre serveur et BDD, de pouvoir tester la bonne fonctionnalité de tout le CRUD pour chaque modèle.

Attention à rajouter ces lignes de code afin de bénéficier de l'implémentation de clefs étrangères dans SQLite :

Dans les créations de table :

```
FOREIGN KEY(id_<entite>) REFERENCES <entites>(id) ON DELETE CASCADE
```

Dans Database.php

```
$this->query("PRAGMA foreign_keys = ON");
```

Je vous laisse libre du détail de chaque tâche, néanmoins, je vous conseille de toujours garder en tête l'utilisation de la POO en parallèle avec la structure MVC, autant que possible.

Créer une branche à votre nom dans tp-ete et envoyer des commit/push régulièrement tout au long du développement de votre application.

Vous tâcherez de mettre votre application au sein d'un réseau LAN virtuel en utilisant une combinaison de ssh et sftp afin d'effectuer la migration et de configurer le serveur Apache de manière appropriée.

La cerise sur le gâteau serait de valider toutes les entrées de donnée en base avec un ValidationController en utilisant du regex.

Bon codage estival !