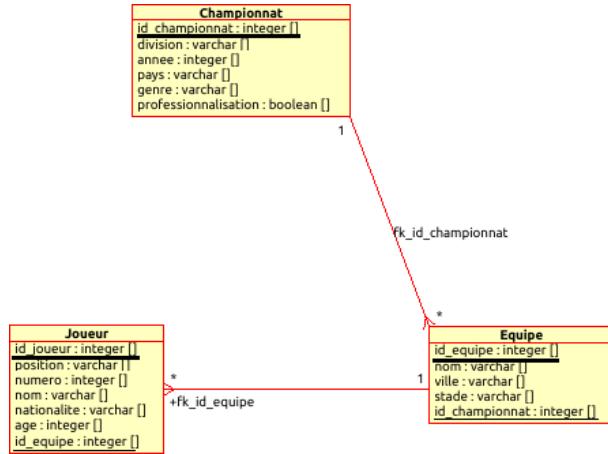


Contrôle 5

TP Football

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



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

- Créer les tables 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 a registration form titled "Football Registration Form". The form fields include:

- Player Name: First (text input), Last (text input)
- Player Email (text input) and Player Team Group (dropdown menu, currently set to "Please Select")
- Offense or Defense: Offense (radio button), Defense (radio button), Don't Know (radio button)
- Height: Height in inches (text input)
- Weight: Weight in lbs (text input)
- 40-Yard Dash (text input)
- Comments (text area)

The background features a faint graphic of a soccer ball.

- 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.

Bon codage estival !