

# Projet Professionnel Encadré 2

## SIO SLAM 2020-2022

### Présentation de la M2L (Maison des Ligues de Lorraine)

La Maison des Ligues de Lorraine (M2L) a pour mission de fournir des espaces et des services aux différentes ligues sportives régionales et à d'autres structures hébergées. La M2L est une structure financée par le Conseil Régional de Lorraine dont l'administration est déléguée au Comité Régional Olympique et Sportif de Lorraine (CROSL).

### Expression des besoins

La ligue régionale de basket de Lorraine a émis le besoin de pouvoir alimenter les résultats de leur championnat régional au travers d'un site web qui leur permettrait ainsi d'afficher le classement des diverses équipes au fil des différentes journées et éventuellement mettre en place une série de statistiques sur les résultats.

La ligue aimerait également utiliser ce site afin de pouvoir y placer diverses informations, comme des articles ou des actualités.

Le site devra être destiné à la fois au grand public et aux responsables de la ligue. Ces derniers pourront alimenter le site en utilisant un back-office (interface d'administration) avec des accès personnalisé (login et mot de passe).

Trois phases composent donc ce projet. Une première phase est consacrée à l'appropriation du présent cahier des charges, une deuxième phase de modélisation (MCD et MLD) et préparation de la maquette (mockup). Une dernière phase qui permettra la réalisation du site web (Développement et base de données) et du back-office.

### Cahier des charges

La première étape consiste à développer une interface web en PHP (que l'on l'appellera back-office) et relié à une base de données MySQL qui permettra :

- D'ajouter les différents clubs (nom équipe, nom entraîneur, logo équipe, etc...). on limitera à 10 le nombre d'équipes maximal. On pourra modifier ou supprimer une équipe en cas d'erreur.
- D'ajouter les scores des matchs sur la période du championnat. Le championnat se déroule sur une durée de 6 mois avec un match toutes les semaines (aller/retour). On pourra modifier ou supprimer des scores en cas d'erreur.
- De créer des comptes utilisateurs avec 2 niveaux d'autorisation : 'administrateur' et 'contributeur'. Un seul administrateur devra exister et pourra créer des comptes 'contributeur'. Il faudra également gérer la modification et la suppression d'un utilisateur.

# Spécifications fonctionnelles

Vous trouverez dans cette section les spécifications fonctionnelles du projet.  
L'application doit être entièrement responsive web design.

Développement en PHP et MySQL de la partie front-office (site web) qui correspondra au site grand public avec les fonctionnalités suivantes :

## **FX1 - Accueil : retour sur la page d'accueil**

Cette fonctionnalité doit permettre à l'utilisateur de revenir sur la page d'accueil du site en gardant sa session ouverte.

## **FX2 - Présentation des équipes**

Cette fonctionnalité doit permettre d'afficher la liste des équipes avec le nom de l'entraîneur, le logo  
Data : teams [ Nom des équipes, Nom entraîneur, logo, information sur équipe ]

## **FX3 - Les résultats sur une journée**

Description : Cette fonctionnalité doit permettre d'afficher les résultats d'une journée sous forme de tableau avec les noms des équipes, les scores et le gagnant. Le gagnant remporte 3 points, 1 point pour chaque équipe en cas de match nul.

On affichera uniquement les résultats sur une journée, les autres journées seront accessibles via une liste de sélection. Le choix des journées se fera sous forme de liste avec un libellé de type "journée x" où x représente le numéro du jour (de 1 à 20). Lors de l'ouverture de la page on affichera le tableau de la dernière journée en cours.

Le championnat se déroule sur 6 mois sachant que tous les matchs d'une journée se déroulent le même soir (le samedi). La saison comporte donc 20 journées.

Data : results [Résultat équipe domicile, Résultat équipe visiteur, journée]

## **FX4 - Le classement des équipes**

Description : Cette fonctionnalité doit permettre d'afficher le classement des équipes sous forme de tableau avec le total des points par équipes. Ce classement est établi sur la base d'une saison de championnat, c'est à dire 6 mois calendaire, ce qui représente 20 journées.

Data : results [Résultat équipe domicile, Résultat équipe visiteur, journée]

## **FX5 - Bouton d'accès login au back-office**

Description : Ce bouton doit donner l'accès au formulaire de connexion au back-office via un login et mot de passe. Le mot de passe devra être vérifié en utilisant tous les moyens sécuritaires mis à disposition avec PHP. La position de ce bouton ne doit pas être sur le même emplacement que les autres rubriques (résultats, classement, etc...)

Data : users [Nom, Prénom, Login, Mot de passe, Est-ce un administrateur, Observation]

La partie back-office permet la gestion des équipes et des utilisateurs devra avoir les fonctionnalités suivantes :

#### **BX1 - Bouton de déconnexion**

Description : Cette fonctionnalité doit permettre de fermer la session d'un utilisateur au back-office. Il est conseillé de garder le login dans un fichier de cookie pour une future connexion.

Data : users [ Nom, Prénom, Login, Mot de passe, Est-ce un administrateur, Observation ]

#### **BX2 - Lien d'accès au site**

Description : Cette fonctionnalité doit permettre d'accéder au site grand public à partir de back-office. L'accès au site se fera en ouvrant un nouvel onglet.

En mode connecté il faudra faire apparaître sur le site, en lieu et place du bouton de login, un bouton d'accès au back-office et un bouton de logout permettant de se déconnecter.

#### **BX3 - Gestion des équipes**

Description : Cette fonctionnalité doit permettre d'ajouter, de modifier ou supprimer une équipe.

L'affichage doit se présenter sous forme de tableau avec le nom de l'équipe, le nom de l'entraîneur, le logo, un bouton "modifier" et un bouton "supprimer".

Le bouton "modifier" pointe vers un formulaire permettant la modification des informations d'une équipe. Le bouton "supprimer" pointe vers une page qui demande la confirmation avant la suppression. Il faudra également placer un lien sur le nom de l'équipe permettant d'arriver sur une page présentant la fiche de l'équipe.

En fin de tableau il faudra placer un bouton "Ajouter une équipe" qui pointerait sur un formulaire permettant l'enregistrement d'une nouvelle équipe avec tous les champs demandés.

Data : teams [ Nom des équipes, Nom entraîneur, logo, information sur équipe ]

#### **BX4 - Gestion des résultats**

Description : Cette fonctionnalité doit permettre d'ajouter, de modifier ou supprimer le résultat d'un match. L'affichage doit se présenter sous forme de tableau avec le nom de l'équipe jouant à domicile, celle jouant à l'extérieur, le numéro de la journée, le nombre de points de l'équipe à domicile et celui de l'équipe à l'extérieur, un bouton "modifier" et un bouton "supprimer". Un tri sera effectué pour ordonner le tableau de la première à la dernière journée.

Le bouton "modifier" pointe vers un formulaire permettant la modification d'un résultat. Le choix des équipes doit se faire par une liste de sélection ainsi que le choix de la journée. Une requête de type Ajax pourra permettre d'afficher uniquement les journées non encore attribuées en fonction du choix des équipes.

Le bouton "supprimer" pointe vers une page qui demande la confirmation avant la suppression.

En fin de tableau il faudra placer un bouton "Ajouter un résultat" qui pointerait sur un formulaire permettant l'enregistrement d'un nouveau résultat.

Data : results [ Résultat équipe domicile, Résultat équipe visiteur, journée ]

#### **BX5 - Gestion des utilisateurs**

Description : Cette fonctionnalité doit permettre d'ajouter, de modifier ou supprimer un utilisateur du back-office. L'affichage doit se présenter sous forme de tableau avec le nom, le prénom, le login, le statut, un bouton "modifier" et un bouton "supprimer".

Le statut d'un utilisateur peut être soit "Contributeur" soit "Administrateur". Le contributeur a un accès au back-office mais pas à la gestion des utilisateurs. L'administrateur lui a un accès total au back-office.

Le bouton "modifier" pointe vers un formulaire permettant la modification d'un utilisateur. Le choix du statut devra se faire par la mise en place de boutons radio. La ligne de saisie du mot de passe devra être de type "password".

Il est demandé de mettre en place un système de protection du mot de passe efficace en utilisant la bibliothèque PHP Libsodium.

En fin de tableau il faudra placer un bouton "Ajouter un utilisateur" qui pointerait sur un formulaire permettant l'ajout d'un nouvel utilisateur.

Data : users [ Nom, Prénom, Login, Mot de passe, Est-ce un administrateur, Observation ]

Les tables suivantes sont obligatoires dans la construction du projet :

- Une table 'teams',
- Une table 'results',
- Une table 'users'

D'autres tables pourront être ajoutées en fonction des choix ou des besoins nécessaires identifiés par l'étudiant.

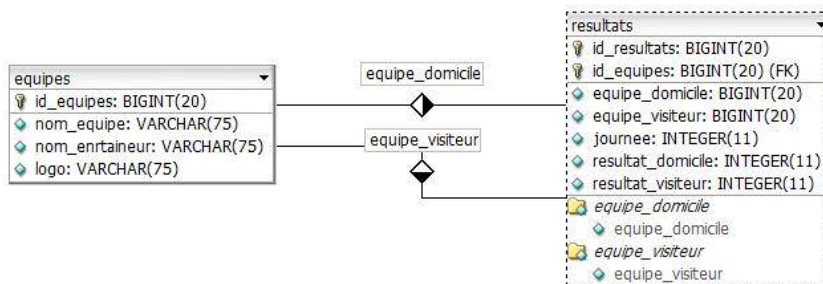
## Evolution à prévoir :

- Une table 'navigation' qui permettra de créer dynamiquement des menus pour le front-office
- Une table 'pages' qui permettra de connecter une page à un menu
- Une table 'articles' qui permettra de créer des articles à placer dans les pages

## Précision sur les relations entre les tables :

Revoir le cours sur les bases de données MySql et notamment sur les clés primaires et étrangères, ainsi que les jointures entre les tables.

Exemple de modèle relationnel de données entre les tables 'teams et 'results' :



Clés primaires : id\_equipes, id\_resultats

Clés étrangères : equipe\_domicile, equipe\_visiteur

Exemple de schéma fonctionnel non représentatif du projet :

La capture d'écran montre l'interface d'un gestionnaire de contenu web, 'GestCont Manager'. Le menu de navigation en haut inclut 'Accueil', 'Contenu', 'Pages', 'Navigation' (sélectionné) et 'Utilisateurs'. Le titre principal de la page est 'Gestion de la navigation'. Les champs de saisie sont :

- Nom : Boutique
- Connecté à la page : Boutique en ligne
- Position : TOP
- Ordre : 3

Un bouton 'Enregister' est visible en bas à droite.

## Travail à fournir

Au travers de leur espace de travail (serveur web) les étudiants devront organiser les répertoires des différents contenus (images, feuille de style, javascript, etc..). Prévoir également un répertoire pour y placer les différentes notices et documentations.

Accès à votre espace de développement : [http://51.178.86.117:<numero\\_de\\_port>](http://51.178.86.117:<numero_de_port>)

Accès à la base de données : <http://51.178.86.117:8087/phpmyadmin/>

Les étudiants devront fournir en plus des divers développements demandés, une notice technique (présentation des scripts, modèle relationnel des données, etc..) et fonctionnel (schéma des divers écrans, parcours de navigation,...)

Travail à réaliser pour la présentation orale de février :

1. Mettre en place l'interface back-office en proposant un design de type responsive en utilisant un framework de type Bootstrap,
2. Mettre en place le site grand public en proposant un design de type responsive en utilisant un framework de type Bootstrap,
3. Mettre en place les diverses tables et représenter le modèle relationnel et logique,
4. Fournir les dessins fonctionnels des écrans du back-office avec un outil de type Balsamiq Mockups (voir exemple d'écran ci-dessus),
5. Développer les rubriques du module back-office (admin) et front (site) en utilisant une architecture MVC. Il est interdit d'utiliser des modèles de template existant. Il est par contre recommandé d'utiliser les frameworks et bibliothèques suivantes :
  - Bootstrap
  - JQuery
6. La phase de développement devra se faire via l'outil de versioning GIT puis déployer sur GitHub avec au minimum les branches "master", "develop".

Des annexes pourront être rajoutées au présent document en fonction des remarques ou évolutions proposées au cours de l'année.