

FAQ

Mise en place d'une Foire Aux Questions

Table des matières

Situation initiale	2
La mission.....	2
Etapes.....	2
1. Création du projet.....	2
2. Création du menu de l'application.....	4
3. Création de la base de données.....	4
4. Création de la page d'accueil de la FAQ.....	6
5. Création de la page de connexion.....	7
6. Création de la page d'inscription	8
7. Création du script PHP de connexion.....	9
8. Création du script PHP de déconnexion	9
9. Création du script PHP d'inscription	10
10. Création du script PHP d'affichage de la liste des messages	11
11. Création du script PHP de création d'un message (question)	12
12. Création du script PHP de modification d'un message (question + réponse)	13
13. Création du script PHP de suppression d'un message (question+réponse)	14
14. Amélioration du script PHP d'affichage de la liste des messages (optionnel).....	15
15. Création du moteur de recherche (optionnel)	15
16. Création du script PHP de notification (optionnel).....	15
17. Réorganisation de la FAQ par catégorie (optionnel)	15
18. Amélioration de la saisie des messages (optionnel)	15
19. Création d'un back-office (optionnel)	15
Annexes.....	16

Situation initiale

Les différentes ligues sportives ont un site qui présente leur activité. Par ailleurs un site de présentation de la maison des ligues présente le fonctionnement.

La mission

Tout internaute doit pouvoir se connecter à l'un des sites afin de poser des questions et consulter les autres questions/réponses relatives au thème du site consulté. Pour cela il doit tout d'abord s'inscrire sur le site.

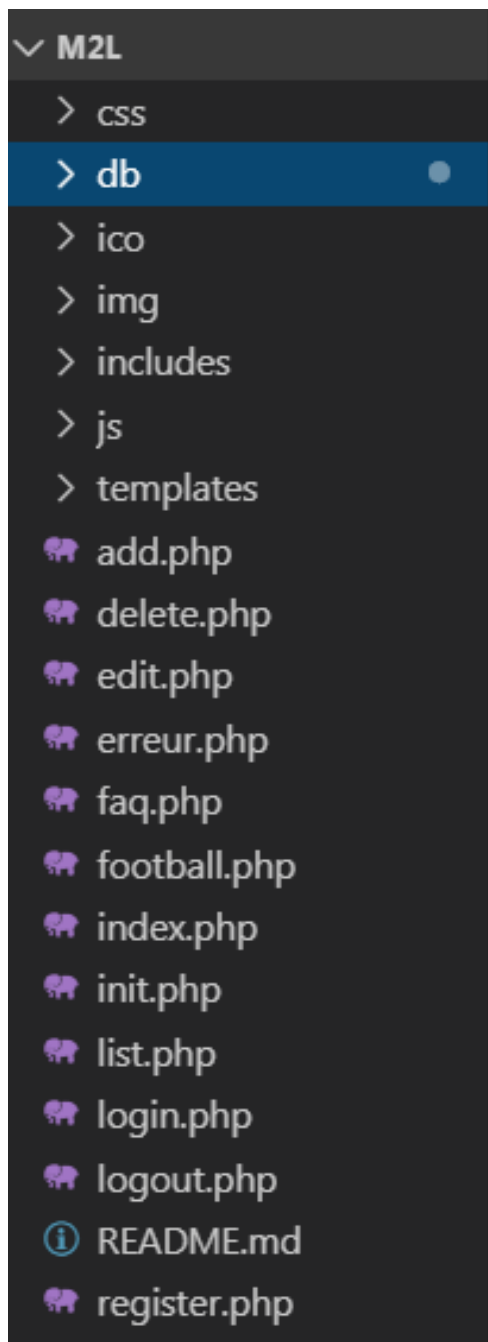
Les sites créés lors de la mission précédente devront être modifiés pour devenir interactifs.

Etapes

1. Création du projet

- Créez le projet « **m2l** » . Le chemin est « `*\xampp\htdocs\projets\m2l` »
- Créez un fichier PHP par **page Web** (Voir Annexes). Pour l'instant, ces fichiers sont vides.
 - index.php
 - faq.php
 - register.php
 - login.php
 - logout.php
 - list.php
 - add.php
 - edit.php
 - delete.php
- Créez un fichier « **init.php** » qui contiendra les paramètres de configuration du site.
- Créez une page d'erreur « **erreur.php** » qui affichera les messages d'erreur.
- Créez les dossiers suivants pour y ranger les différents types de fichiers
 - img
 - css
 - ico
 - includes
 - js
 - db
 - templates

Exemple de projet sous **VSCode**.



2. Création du menu de l'application

Créez un menu « **menu.php** » à insérer dans TOUTES les pages.

Il permet d'accéder à n'importe quelle page du site (Voir la carte du site en annexe)

Exemple de menu.

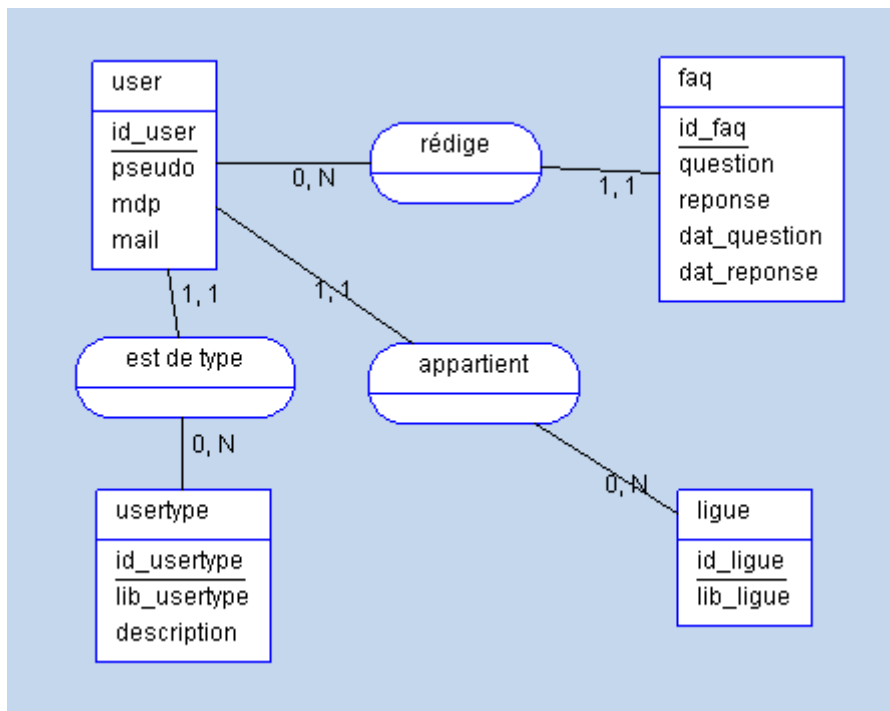
Bienvenue, jef

Menu

- [Maison des ligues](#)
- [Ligue de football](#)
- [Accueil de la FAQ](#)
- [Liste des questions](#)
- [Ajouter une question](#)
- [Connexion](#)
- [Déconnexion](#)
- [Inscription](#)

3. Création de la base de données

Définissez la base de données « **m2l** » à partir du MCD suivant.



Entité « faq »

Elle contient les messages (question + réponse). Un message est rédigé par un utilisateur et un seul.

id_faq	question	reponse	dat_question	dat_reponse	id_user
1	Est ce que le ballon est rond ?	Bien sûr que oui	2015-01-03 14:42:15.000000	2015-01-03 14:57:14.000000	1

Entité « user »

Elle contient les utilisateurs. Ils sont enregistrés via la page d'inscription. Le mot de passe doit être **haché**.

id_user	pseudo	mdp	mail	id_usertype	id_ligue
1	jef	\$2y\$10\$Xaes8DuWAdbRxfQqW.bekekmRJZmSGuQVPnQj6hUvu8...	jef@m2l.com	1	1
2	tintin	\$2y\$10\$1yRL8HPU9A9XnQYZczOBteyUXhB4ygY8yH72wQva.Hv...	tintin@m2l.com	1	1

Entité usertype

Elle contient les types d'utilisateur. Il y en a actuellement trois.

- « **utilisateur** » : il ne peut que poser une question.
- « **administrateur** » : il peut répondre à toutes les questions. Il peut aussi les poser, les modifier ou même les supprimer (question + réponse). Par contre, il ne peut agir que sur les messages de sa ligue.
- « **super-administrateur** » : il fait la même chose que l'administrateur mais sur toutes les ligues.

id_usertype	lib_usertype	description
1	utilisateur	Utilisateur de base
2	admin	Administrateur de ligue
3	superadmin	Administrateur de toutes les ligues

Entité ligue

Elle contient les différentes ligues. Une d'entre-elles est nommée « **toutes les ligues** » afin de pouvoir gérer les droits du super administrateur.

id_ligue	lib_ligue
1	Ligue de football
2	Ligue de basket
3	Ligue de volley
4	Ligue de handball
5	Toutes les ligues

Consignes pour la création de la base

- L'encodage est **UTF-8**.
- Le moteur est **innoDB** (à cause des clés étrangères).
- Les noms de la base, des tables et des rubriques sont en **minuscules sans accent**.
- Le script de création de la base est « **m2l.sql** ». Il contient toutes les instructions nécessaires pour créer et remplir la base avec un jeu d'essais minimum mais cohérent. Il sera stocké dans le dossier « **db** » du projet.

- Les clés primaires et étrangères sont en **bigint(11)**. Elles sont toujours **auto-incrémentées**.
- La base contient des **clés étrangères** (Voir ci-dessous).
- Les dates sont au format MySQL « **Date** » ou « **DateTime** » suivant les besoins.
- Les chaînes de caractères sont de type « **char()** » ou « **varchar()** » sauf « question » et « réponse » qui sont de type « **text** ».

Clés étrangères

FK_faq_id_user : relie faq(id_user) à user(id_user). Un utilisateur doit exister dans la table « user » pour pouvoir poser une question.

FK_user_id_ligue : relie user(id_ligue) à ligue(id_ligue). Un utilisateur appartient à une et une seule ligue.

FK_user_id_usertype : relie user(id_usertype) à usertype (id_usertype). Un utilisateur est d'un et un seul type.

Travail

Créez le **MLD** puis créez la **base MySQL**.

Créez un **jeu d'essais** qui vous permettra de démarrer les tests (questions, ligues, utilisateurs et types)

Générez ensuite le **script SQL**. Vous pouvez le stocker dans un dossier « **db** » de votre projet afin de centraliser le code en un seul endroit.

4. Création de la page d'accueil de la FAQ

Créez la page d'accueil de la FAQ « **index.php** ». Cette page doit proposer des liens vers les pages de connexion et d'inscription.



Cette page servira de **modèle** pour toutes les autres pages.

Insérez le menu et un bas de page qui se répétera sur TOUTES les pages.

5. Création de la page de connexion

Créez la page de connexion « **login.php** ».
La connexion est obligatoire pour accéder à la FAQ.

Connexion

Veuillez remplir le formulaire

Pseudo

Mot de passe

Envoyer

Réinitialiser

- Le pseudo est obligatoire
- Le mot de passe est obligatoire

Pseudo

Veuillez compléter ce champ.

Ces données seront plus tard comparées à celles qui sont dans la base de données.

Quand l'utilisateur sera connecté, prévoyez d'afficher un message au-dessus du menu confirmant qu'il est identifié.
Par exemple.

Bienvenue, jef

Menu

- [Maison des ligues](#)
- [Ligue de football](#)
- [Accueil de la FAQ](#)
- [Liste des questions](#)
- [Ajouter une question](#)
- [Connexion](#)
- [Déconnexion](#)
- [Inscription](#)

6. Création de la page d'inscription

Créez la page d'inscription « **register.php** ».

L'inscription est obligatoire pour se connecter et accéder à la FAQ.

Les données du formulaire alimenteront plus tard la table « **user** ».

Inscription à la FAQ

Veuillez fournir vos coordonnées

Pseudo

Mot de passe

Mail

@m2l.com

Ligue

Ligue de football ▼

Enregistrer

Pseudo

Veuillez compléter ce champ.

- Le pseudo est obligatoire
- Le mot de passe est obligatoire
- L'adresse mail est obligatoire
- La ligue figure dans la table « **ligue** ». On peut toutefois utiliser une liste « en dur » car les ligues ne changent pas fréquemment

Mail

Veuillez saisir une adresse courriel valide.

Champ obligatoire en HTML5

On peut forcer la saisie d'un champ en HTML5. Voici quelques liens.

http://www.w3schools.com/tags/tag_input.asp

<http://www.wufoo.com/html5/>

7. Création du script PHP de connexion

La connexion utilise le principe des « **sessions PHP** ».

1. On récupère la saisie du pseudo et du mot de passe
2. On vérifie dans la table « user » s'ils existent (le mot de passe est haché, voir étape 11)
3. S'ils existent, on crée une variable de session qui contient ce pseudo
4. Tant que cette variable de session existe, l'utilisateur est connecté

Le script doit être capable d'afficher des messages d'aide (message « **flash** ») afin de guider l'utilisateur.

Connexion

Veuillez remplir le formulaire

Pseudo

Mot de passe

Envoyer

Réinitialiser

Connexion

Essayez encore !

Pseudo

Veuillez compléter ce champ.

Envoyer

Réinitialiser

Toutes les pages à accès réservé devront vérifier si l'utilisateur est bien connecté.

8. Création du script PHP de déconnexion

La déconnexion utilise aussi le principe des « **sessions PHP** ».

Il suffit de détruire les données de session pour que l'utilisateur soit déconnecté.

La déconnexion ne doit pas forcément afficher une page HTML ou un message, elle peut simplement renvoyer sur la page d'accueil de la FAQ.

Vous pouvez créer au choix une icône, un lien ou un bouton pour déclencher la déconnexion.

Menu

- [Maison des ligues](#)
- [Ligue de football](#)
- [Accueil de la FAQ](#)
- [Liste des questions](#)
- [Ajouter une question](#)
- [Connexion](#)
- [Déconnexion](#)
- [Inscription](#)

9. Création du script PHP d'inscription

Ce script « **register.php** » est découpé en quatre étapes.

1. On récupère la saisie des données nécessaires à l'inscription
2. On vérifie si elles sont correctes (par exemple, le pseudo ne doit pas être déjà utilisé)
3. On enregistre les données dans la table « **user** » (le mot de passe est haché, voir plus bas)
4. On signale à l'utilisateur que l'inscription a réussi

Inscription à la FAQ

Veuillez fournir vos coordonnées

Pseudo

mabiche

Mot de passe

•••••

Mail

@m2l.com

Ligue

Ligue de football

Ligue de football

Ligue de basket

Ligue de volley

Ligue de handball

Exemple de message d'erreur

Mail

@m2l.com

Veuillez saisir une adresse courriel valide.

Exemple de message de confirmation

Inscription à la FAQ

mabiche, votre inscription a bien été prise en compte

Hachage

Le hachage (« hash » en anglais) consiste à coder une chaîne de façon à ce qu'on ne puisse pas retrouver la chaîne d'origine à partir de la version hachée. Cela évite de se faire piller ses mots de passe en cas d'intrusion dans la base de données.

Le principe et la mise en œuvre en PHP sont décrits sur ces sites :

<http://grunk.developpez.com/tutoriels/php/mots-de-passe-securises/>















<http://php.net/manual/en/function.password-hash.php>

10. Création du script PHP d'affichage de la liste des messages

Ce script « **list.php** » affiche la liste des messages. Dans cette page, on peut aussi ajouter un message, modifier ou supprimer un message.

Selon les règles décrites précédemment, seul l'administrateur ou le super-administrateur peuvent modifier ou supprimer un message.

Liste des questions de la FAQ

Nr	Auteur	Question	Réponse	Actions
1	jef	Est ce que le ballon est rond ?	Bien sûr que oui	 
2	jef	Quel est l'âge du Capitaine	Ce qu'il faut	 
3	jef	Est ce qu'on va gagner ?	Peut-être	 
6	jef	Est ça c'est du cochon ?		 
7	jef	Quel est le tél. de Nabila ?		 
8	jef	Que signifie les initiales P.S.G ?	Pas Sûr de Gagner	 
10	super	gras	C'est malin	 

 [Ajouter une question](#)

Icônes

Pour améliorer la présentation, vous pouvez utiliser les icônes libres « **silk** ».

<http://www.famfamfam.com/lab/icons/silk/>

URL paramétrée

Pour accéder à un message en particulier, vous pouvez paramétrer les URL qui servent de lien de la façon suivante.

<http://localhost/projets/m2l/edit.php?id=7>

edit.php : script PHP vers lequel l'URL pointe

?id=7 : paramètre qui sera reçu dans le script via la variable \$_GET['id']

Gestion des droits (rappel)

Utilisateur : il ne peut que poser des questions

Administrateur : il peut poser des questions, y répondre et supprimer des messages mais seulement pour ceux qui correspondent à sa ligue

Super-administrateur : il fait la même chose que l'administrateur mais sur toutes les ligues.

11. Création du script PHP de création d'un message (question)

Ce script « **add.php** » est découpé en quatre étapes.

1. On affiche un formulaire de saisie d'une nouvelle question
2. On récupère la saisie
3. On enregistre les données dans la table « **faq** » (Voir plus bas)
4. On signale à l'utilisateur que la question est enregistrée

Ajouter une question à la FAQ

Veuillez saisir votre question

Question

Enregistrer

 [Revenir à la liste](#)

Alimentation des rubriques

Les rubriques suivantes dans la table « **faq** » sont automatiquement alimentées.

- **Id_faq** : auto-incrémentée
- **reponse** : Null tant que l'administrateur n'a pas répondu
- **dat_question** : horodatage courant (date+heure)
- **dat_reponse** : Null tant que l'administrateur n'a pas répondu
- **id_user** : ID de l'utilisateur connecté qui a posé la question

12. Création du script PHP de modification d'un message (question + réponse)

Ce script « **edit.php** » est découpé en quatre étapes.

1. On affiche le message dans un formulaire de saisie
2. On récupère la saisie
3. On enregistre les données dans la table « **faq** »
4. On signale à l'utilisateur que la question est enregistrée

Question N° 1

Veuillez saisir les modifications de cette FAQ

Question

Est ce que le ballon est rond ?

Réponse

Bien sûr que oui

Enregistrer

 [Revenir à la liste](#)

Alimentation des rubriques

Les rubriques suivantes dans la table « **faq** » sont automatiquement alimentées.

- **Reponse** : la réponse de l'administrateur
- **Dat_reponse** : horodatage courant (date+heure)

Gestion des droits (rappel)

Seuls l'administrateur ou le super-administrateur peuvent répondre ou modifier des messages.

L'administrateur ne peut modifier que les messages de sa ligue.

13. Création du script PHP de suppression d'un message (question+réponse)

Ce script « **delete.php** » est découpé en quatre étapes.

1. On affiche le message dans un formulaire (toute saisie est impossible)
2. On attend que l'utilisateur clique sur le bouton de confirmation
3. On supprime le message dans la table « **faq** »
4. On signale à l'utilisateur que le message a bien été supprimé

Question N° 10

Veuillez cliquer sur le bouton pour supprimer cette question

Question

```
<b>gras</b>
```

Réponse

```
C'est malin
```

Supprimer

 [Revenir à la liste](#)

Gestion des droits (rappel)

Seuls l'administrateur ou le super-administrateur peuvent supprimer des messages.
L'administrateur ne peut supprimer que les messages de sa ligue.

14. Amélioration du script PHP d'affichage de la liste des messages (optionnel)

Améliorez le script « **list.php** » en proposant les fonctions suivantes :

- Pagination de la liste
- Tri sur un des critères suivants : par nr, par auteur ou anti chronologique (le plus récent en premier)
- Tri sur les entêtes de colonne
- Filtre n'affichant que les messages sans réponse ou d'un seul utilisateur

15. Création du moteur de recherche (optionnel)

Créez le script « **search.php** » qui permet de rechercher des mots-clés dans les messages.

En réponse, ce script affichera une page de « résultats ».

16. Création du script PHP de notification (optionnel)

Créez le script « **notify.php** » qui permet de notifier les utilisateurs de certains événements.

Par exemple :

- Une question vient d'être posée
- Une réponse vient d'être saisie
- Une réponse à une question de l'utilisateur vient d'être saisie

La notification peut se traduire par l'envoi de mails en PHP.

17. Réorganisation de la FAQ par catégorie (optionnel)

Le principe est de rassembler les messages par catégorie.

Le travail consiste donc à :

- Créer les catégories dans la base de données
- Modifier les messages existants pour qu'ils soient associés à une catégorie
- Modifier l'affichage de la liste des messages par catégorie (ou toutes catégories confondues)
- Modifier les formulaires de création et de modification des messages pour qu'ils prennent en compte la catégorie

18. Amélioration de la saisie des messages (optionnel)

Améliorez la saisie des messages (questions et réponses) avec un éditeur WYSIWYG, par exemple CKEDITOR.

<http://ckeditor.com/demo>

19. Création d'un back-office (optionnel)

Le principe est de gérer via une interface intégrée les données qui sont actuellement gérées via PHPMYAdmin.

- Les utilisateurs
- Les ligues
- Les types d'utilisateur

Les messages pourraient aussi être complètement gérés (dates ou code utilisateur par exemple).

Le travail consiste donc à créer un back-office permettant de gérer ces tables. L'accès serait réservé aux administrateurs.

Annexes

Web M2L – Gestion de la FAQ

