

INSTALLATION BDD

Nous vous fournissons d'un fichier .sql avec tous programmer à l'intérieur. Les choses dont vous devez vous occuper sont :

- Importation du fichier "cubic_market.sql" dans phpmyadmin
- Relation entre votre base de donnée et le vos fichier php

L'intégralité de la structure et des données nécessaires au projet est fournie via un dossier projetGroup. La mise en place requiert les interventions suivantes :

- Importation de la base de données : Procéder à l'importation du fichier cubic_market.sql via l'interface de gestion phpMyAdmin.
- Configuration de la liaison entre le serveur de base de données et les fichiers PHP de l'application (paramétriser le fichier de connexion).

La première étape consiste à initialiser la base de données sur le serveur cible. Pour ce faire, l'utilisateur devra :

1. Accéder à l'interface phpMyAdmin du serveur hôte.
2. Utiliser la fonctionnalité « Importer » pour charger le fichier source fourni.

The screenshot shows the phpMyAdmin interface with the following details:

- Header:** Shows "Serveur: 127.0.0.1".
- Navigation:** Shows a sidebar with databases like Nouvelle_base_de_données, cabinet_veterinaire, cubic_market, gestion_aeroport, information_schema, ipsi_db, livre_db, lol_database, mysql, performance_schema, phpmyadmin, projet_ecologique, reservation_salle, rpg_fantasy, spotify_songs, test, and todo.
- Main Tabs:** Bases de données, SQL, État, Comptes utilisateurs, Exporter, Importer (highlighted with a red box), Paramètres, Réplication, Variables, Jeux de caractères.
- Import Tab Content:**
 - Paramètres généraux:** Interclassement pour la connexion au serveur: utf8mb4_unicode_ci.
 - Paramètres d'affichage:** Langue (Language): Français - French; Thème: pmahomme.
 - Serveur de base de données:** Serveur: 127.0.0.1 via TCP/IP; Type de serveur: MariaDB; Connexion au serveur: SSL n'est pas utilisé; Version du serveur: 10.4.32-MariaDB - mariadb.org binary distribution; Version du protocole: 10; Utilisateur: root@localhost; Jeu de caractères du serveur: UTF-8 Unicode (utf8mb4).
 - Serveur Web:** Apache/2.4.58 (Win64) OpenSSL/3.1.3 PHP/8.2.12; Version du client de base de données: libmysql - mysqld 8.2.12; Extension PHP: mysqli, curl, mbstring.
 - phpMyAdmin:** Version: 5.2.1, dernière version stable: 5.2.3; Documentation, Site officiel, Contribuer, Obtenir de l'aide, Liste des changements, Licence.
- Bottom:** Console de requêtes SQL.

Ensuite, sélectionner le fichier "cubic_market.sql" via le bouton « Choisir un fichier ».

The screenshot shows the 'Importation dans le serveur courant' (Import into current server) page in phpMyAdmin. On the left, a sidebar lists various databases. The main area contains three sections: 'Fichier à importer' (File to import), 'Importation partielle' (Partial import), and 'Autres options' (Other options). The 'Format' dropdown is set to 'Console de requêtes SQL'. A red box highlights the 'Choisir un fichier' (Select a file) input field in the 'Fichier à importer' section.

Après avoir vérifié le fichier a bien été choisi, valider l'importation en cliquant sur le bouton « Importer » situer au bas de la page.

The screenshot shows the same 'Importation dans le serveur courant' page after the file has been selected. The 'Format' dropdown is now set to 'SQL'. The 'Importation partielle' section shows the 'Permettre l'interruption de l'importation pour respecter la limite de temps définie dans PHP' checkbox is checked. The 'Autres options' section has the 'Activer la vérification des clés étrangères' checkbox checked. The 'Format' dropdown is set to 'SQL'. A red box highlights the 'Importer' button at the bottom of the page.

Une fois l'importation du fichier sql terminés, l'étape suivante consiste à configurer la liaison entre le serveur de base de données et les fichiers PHP de l'application.

La gestion des accès s'effectue au sein du répertoire racine du projet.

Voici les étapes à suivre :

- Accédez au répertoire `/projetGroup/config/`.
- Ouvrez le fichier de configuration `db.php`.
- Saisissez le nom d'utilisateur associé à votre base de données.

- Saisissez le mot de passe associé à votre base de données.

Assurez-vous que l'utilisateur configuré dispose des privilèges suffisants pour ne pas avoir de problème de relation.

```
$user = "root";
$pwd = "";
```

Explication de l'arborescence

Voici l'arborescence :

```
cubic-market/
├── config/
│   └── db.php
└── public/
    ├── includes/
    │   ├── ajax_search.php
    │   └── autoloader.php
    ├── src/
    │   ├── css/
    │   │   ├── style.css
    │   │   ├── styleAddProduct.css
    │   │   ├── styleLogin.css
    │   │   └── styleShop.css
    │   ├── js/
    │   │   ├── app.js
    │   │   └── appSearch.js
    │   ├── svg/
    │   │   ├── hamburger.svg
    │   │   └── imgAdd.svg
    │   └── uploads/
    │       ├── background.jpg
    │       ├── choiceProduct.png
    │       └── product1.png
    ├── boutique.php
    ├── formAddProduct.php
    ├── index.php
    └── inscription.php
src/
├── Entities/
│   ├── Arme.php
│   ├── Armure.php
│   ├── Compte.php
│   ├── Grade.php
│   └── Produit.php
└── Managers/
    └── BoutiqueManager.php
```

Le dossier config/

Le dossier est là où le fichier est relié à la base de données.

- db.php : Contient les identifiants de connexion à la base de données (hôte, nom de la base, utilisateur, mot de passe).

Le dossier public/

C'est le seul dossier accessible par les utilisateurs via leur navigateur.

- Contient les fichiers (index.php, boutique.php, etc.) => Ce sont les pages web accessible (par un user, admin). L'index.php est la page principal (page qui permet de se connecter ou de s'inscrire).
- includes/ : Contient des scripts php.
 - autoloader.php : Permet de charger automatiquement les classes PHP sans faire des require_once ou require partout.
 - ajax_search.php : Effectue une recherche dynamique.
- src/ : Contient les fichiers "front-end" :
 - css / js : Mon design et mes scripts d'interactivité.
 - svg / uploads : Les icônes svg et les images téléchargées (produits, arrière-plan de la page).

Le dossier src/

C'est ici que se trouve toute la logique métier de ton application. Ce dossier n'est pas accessible par l'URL, ce qui protège ton code source.

- Entities/ : Ce sont des classes qui représentent mes données en base de données sous forme d'objets.
- Managers/ : Ce sont les classes qui font le lien entre mes Entities et la Base de données.
 - BoutiqueManager.php contient les requêtes SQL pour récupérer, envoyer et modifier des données.

Test de l'application

Voici des comptes “ADMIN” et “USER” pour tester notre application. Dans la page index.php, penser à rentrer les bonnes valeurs aux bons endroits.

COMPTE ADMIN :

- Pseudo -> Admin
- Mot de passe -> Admin1234

COMPTE USER :

- Pseudo -> Tom
- Mot de passe -> Admin1234