# Instructions d'installation et Authentification utilisateur

## 1. Installation du projet

Ce projet est une application de gestion de flotte de voitures d'entreprise développée en PHP avec une base de données MySQL.  
  
Étapes d'installation :  
1. Cloner le dépôt Git : `git clone https://github.com/votre-utilisateur/backend\_flotte\_voiture.git`  
2. Copier les fichiers dans le répertoire `htdocs` de XAMPP ou dans le dossier racine de votre serveur web.  
3. Créer la base de données MySQL en important le fichier `create\_database.sql` via phpMyAdmin.  
4. Configurer les paramètres de connexion dans `config/db.php` :  
 - Hôte : localhost  
 - Utilisateur : root  
 - Mot de passe : (laisser vide si non défini)  
 - Nom de la base : flotte\_voiture  
5. Lancer le serveur Apache et MySQL via XAMPP.  
6. Accéder à l'application via `http://localhost/backend\_flotte/index.php`

## 2. Authentification utilisateur

L'application intègre un système d'authentification sécurisé.  
  
Fonctionnalités :  
- Formulaire de connexion avec email et mot de passe.  
- Vérification des identifiants via la base de données.  
- Hachage des mots de passe avec `password\_hash()`.  
- Protection contre les attaques CSRF via jeton de session.  
- Gestion des rôles : `admin` et `employé`.  
- Redirection vers le tableau de bord en fonction du rôle.  
- Déconnexion sécurisée via `logout.php`.  
  
Fichiers concernés :  
- `login.php` : formulaire et traitement de connexion.  
- `logout.php` : destruction de la session.  
- `db.php` : connexion PDO sécurisée.  
- `users.php` : gestion des utilisateurs avec rôles et statuts.

## 3. Structuration pour examen et validation des processus de fonctionnement de l’API

3.1. Routes REST définies

Selon mon routeur principal, il y a :

|  |
| --- |
| $router->group('/api/routes', function($r) {  $r->resource('/users', 'usersController');  $r->add('GET', '/articles/{slug}', 'ArticleController@read');  $r->add('POST', '/auth', 'AuthController@login');  $r->add('POST', '/auth/handleForm', 'AuthController@handleForm');  }); |

Routes REST attendues pour /users :

| Méthode | URI | Action attendue |
| --- | --- | --- |
| GET | /api/routes/users | Liste des utilisateurs |
| GET | /api/routes/users/{id} | Détail d’un utilisateur |
| POST | /api/routes/users | Création d’un utilisateur |
| PUT | /api/routes/users/{id} | Mise à jour d’un utilisateur |
| DELETE | /api/routes/users/{id} | Suppression d’un utilisateur |

Endpoint pour /users :

| **Méthode** | **Endpoint** | **Description** | **JWT utilisé** | **Résultat attendu** |
| --- | --- | --- | --- | --- |
| GET | /users | Liste des utilisateurs | JWT actif | **200 OK** |
| GET | /users | Accès avec JWT inactif | JWT inactif | **403** "Votre compte est désactivé..." |
| GET | /users | Accès sans JWT | Aucun | 401 "Authentification requise" |
| POST | /users | Création d’un utilisateur | JWT admin actif | **201 Created** |
| POST | /users | Création avec JWT employé | JWT employé actif | **403** "Accès interdit : rôle insuffisant" |
| PUT | /users/{id} | Modification utilisateur | JWT admin actif | **200 OK** |
| PUT | /users/{id} | Modification avec JWT inactif | JWT admin inactif | **403** "Votre compte est désactivé..." |
| DELETE | /users/{id} | Suppression utilisateur | JWT admin actif | **204** No Content |
| DELETE | /users/{id} | Suppression sans JWT |  |  |

Prérequis :

|  |
| --- |
| * Base URL : http://localhost/api/routes/users * JWT actif : généré via /api/routes/auth avec un compte actif * JWT inactif : généré via /api/routes/auth avec un compte inactif * JWT admin/employé : selon le rôle du compte |

Structure de la collection Postman

|  |
| --- |
| 📁 **Flotte API Tests**  ├── **Authentification**  │ ├── POST /auth (login)  │ └── POST /auth/handleForm (formulaire)  ├── **Utilisateurs**  │ ├── GET /users (admin actif)  │ ├── GET /users (employé actif)  │ ├── GET /users (admin inactif)  │ ├── POST /users (admin)  │ ├── PUT /users/{id} (admin)  │ ├── DELETE /users/{id} (admin)  ├── **Sécurité**  │ ├── GET /users sans JWT  │ ├── GET /users avec JWT corrompu  │ ├── GET /users avec rôle insuffisant  │ └── GET /users avec statut inactif |

Variables d’environnement Postman

|  |  |
| --- | --- |
| **Nom** | **Valeur exemple** |
| base\_url | http://localhost/api/routes |
| jwt\_token | (sera rempli automatiquement après login) |