

# Projet PHP – Création d'un Site Web Innovant

---

Bienvenue dans ce défi de deuxième année en développement PHP ! Ce projet collaboratif vous invite à concevoir et développer un site web dynamique, en mettant en œuvre de nombreuses fonctionnalités modernes et en respectant les bonnes pratiques de développement.

---

## 1. Contexte et Objectifs

---

### Contexte

Dans un environnement numérique en constante évolution, la maîtrise du développement web et la capacité à concevoir des architectures robustes sont des compétences clés. Ce projet vous permettra de mettre en pratique vos connaissances en PHP et en base de données, tout en travaillant en équipe.

### Objectifs

- **Développer un site web en PHP** qui répond à un cahier des charges précis.
  - **Concevoir une architecture logicielle** propre et modulaire.
  - **Implémenter des mesures de sécurité** pour protéger l'application (prévention des injections SQL, XSS, etc.).
  - **Intégrer une interface utilisateur (UI) attractive** et ergonomique.
  - **Ajouter des fonctionnalités avancées** pour aller au-delà des exigences de base (exemples : Query Builder personnalisé, router avancé, gestion en temps réel, upload de fichiers, système de migrations BDD, etc.).
- 

## 2. Constitution des Groupes

---

- **Nombre de personnes par groupe** : 3
  - **Règles de constitution** : Libre – choisissez vos partenaires en fonction de vos affinités et complémentarités !
- 

## 3. Livrables et Modalités de Rendu

---

### Dépôt Git Public

- Votre code source devra être rendu sur [EEMI.tech](https://eemi.tech) via un dépôt Git public.

### README.md Obligatoire

Votre dépôt doit contenir un fichier `README.md` qui précise :

- **Les membres du groupe** : Indiquez le couple pseudo/nom de chaque étudiant (attention aux multiples pseudos).
- **Les contributions** : Précisez les fonctionnalités développées par chacun.
- **Les instructions de lancement** : Détaillez comment installer, configurer et démarrer le projet.
- **Le dump SQL (DDL)** : Fournissez le script de création de votre base de données.

**Important** : Chaque membre du groupe doit avoir contribué activement. En cas de non-participation d'un étudiant, celui-ci recevra la note de 0.

---

## 4. Cahier des Charges

---

### Base de Données

- **Utilisation de PDO** : Toute interaction avec la base doit se faire via PDO.
- **Schéma de la BDD** : Créez au minimum 4 tables en plus de la table `User` (hors table de relation Many-to-Many).

### Authentification et Gestion des Droits

- **Système d'authentification** : Implémentez les pages de login et d'inscription.
- **Gestion des rôles** : Mettez en place une gestion fine des droits (par exemple, un utilisateur ne peut voir que ses propres commandes, alors qu'un administrateur peut accéder à toutes les commandes).

### Pages Obligatoires

Votre site devra comprendre au minimum les pages suivantes :

- **Login** : Pour permettre aux utilisateurs de se connecter.
- **Register** : Pour l'enregistrement de nouveaux utilisateurs.
- **Profil** : Pour que chaque utilisateur puisse consulter et modifier ses informations personnelles.
- **Landing Page** : Une page d'accueil accessible sans authentification, pouvant afficher des données issues de la base (ex. : statistiques, annonces, etc.).

- **Pages supplémentaires** : Développez d'autres pages en fonction du thème choisi pour enrichir votre projet.

---

## 5. Critères d'Évaluation

---

Les projets seront notés selon le barème suivant :

- **7 pts – Fonctionnalités de Base** :
  - Respect du schéma de la BDD (nombre de tables, utilisation de PDO).
  - Mise en place des mesures de sécurité (validation des formulaires, protection contre les injections SQL et XSS, etc.).
- **3 pts – Interface Utilisateur (UI)** :
  - Esthétique, ergonomie et fluidité de l'interface.
- **5 pts – Architecture Logicielle** :
  - Organisation claire des fichiers, modularité et réutilisabilité des composants.
- **5 pts – Fonctionnalités Supplémentaires** :
  - Ajout d'outils ou de techniques avancées, tels que :
    - Query Builder personnalisé
    - Router avancé (tableau de routes, actions)
    - Système de migrations BDD
    - Gestion en temps réel
    - Upload de fichiers
    - ... (à valider selon le projet)
- **Malus** :
  - Code mort (commentaires inutiles), indentation incohérente ou illisible.
  - Failles de sécurité (injection SQL, XSS, élévation de privilèges).

**Attention** : En cas de triche détectée, la note du groupe sera automatiquement ramenée à 0.

---

## 6. Conseils et Astuces

---

- **Planifiez bien votre travail** : Répartissez les tâches de manière équilibrée afin que chacun puisse apporter sa contribution.
- **Testez régulièrement** : Vérifiez au fur et à mesure que chaque fonctionnalité fonctionne et que votre code reste sécurisé.
- **Documentez votre code** : Une bonne documentation facilitera la compréhension et la maintenance de votre projet.
- **Soyez créatifs** : N'hésitez pas à proposer des fonctionnalités innovantes qui apporteront une réelle valeur ajoutée à votre site.

---

## 7. Conclusion

---

Ce projet est une occasion unique de mettre en pratique vos compétences en développement web tout en travaillant en équipe. Faites preuve de rigueur, d'organisation et de créativité pour réaliser une application de qualité. Nous avons hâte de découvrir vos réalisations et de voir comment vous avez relevé ce défi !

Bonne chance à tous et amusez-vous bien !