

# Création d'un site e-commerce dynamique en PHP

## Description du projet :

Vous allez développer un site e-commerce dynamique en PHP, en utilisant une base de données SQL. Le site devra permettre aux utilisateurs de consulter des articles, de s'inscrire et de se connecter, de gérer un panier, et d'effectuer des opérations CRUD via une interface d'administration (back-office).

Les étudiants auront une certaine liberté pour définir le thème du site (ex. vêtements, accessoires, livres, etc.).

Le projet doit être développé localement sur un serveur comme **MAMP**, **XAMPP**, ou tout autre environnement local adapté.

## Fonctionnalités attendues

### 1. Pages front-office accessibles aux utilisateurs

#### a) Accueil

- Présente une vue d'ensemble du site (bannière, quelques produits en avant, description brève).

#### b) Qui sommes-nous ?

- Page statique présentant l'équipe ou le concept derrière le site.

#### c) Articles (catalogue des produits)

- Affiche la liste des produits disponibles (photo, nom, prix, description courte).
- Possibilité de cliquer sur un produit pour voir une fiche produit détaillée.

#### d) Inscription/Connexion

- Formulaire pour créer un compte utilisateur avec validation des données (ex. email valide, mots de passe correspondants).
- Formulaire de connexion pour permettre aux utilisateurs de se connecter.
- Authentification sécurisée (hachage des mots de passe via **password\_hash**).

#### e) Panier

- Les utilisateurs doivent pouvoir ajouter des articles dans leur panier depuis la page articles.
- Une page dédiée permettra de visualiser le contenu du panier : liste des articles avec leur quantité et leur prix total.
- Option pour supprimer un article du panier ou vider complètement le panier.

### 2. Back-office (administration)

Un espace réservé aux administrateurs pour gérer le site :

#### a) Connexion (authentification admin)

- Authentification via un formulaire réservé aux administrateurs.

#### b) Gestion des produits (CRUD)

- Ajouter un nouvel article (avec image, nom, description, prix, stock).
- Modifier les informations d'un article existant.

- Supprimer un article.
- Afficher la liste des articles avec leurs détails.

### c) Gestion des utilisateurs

- Visualiser la liste des utilisateurs inscrits.
- Supprimer un utilisateur si nécessaire.

## Contraintes techniques

### 1. Base de données SQL :

- Au minimum 5 tables :
  - **users** : pour gérer les utilisateurs (id, nom, email, mot de passe, rôle).
  - **items** : pour gérer les produits (id, nom, description, prix, stock, date de publication, image).
  - **orders** : pour gérer les commandes (id, id\_user, id\_item).
  - **invoice** : pour gérer la facture (id, id\_user, date de transaction, montant, adresse de facturation, ville, code postal)
  - **stock** : pour gérer les quantités (id, it\_item, quatité item en stock)
- Relation entre les tables selon le modèle relationnel (ex. clefs étrangères).

### 2. Langages et outils :

- PHP pour le back-end
- HTML/CSS, JS etc... pour le front-end (vous êtes plutôt libres. Vous pouvez également utiliser Bootstrap).
- MySQL pour la base de données.

### 3. Hébergement local :

- Utilisez un serveur local (MAMP, XAMPP etc...).

### 4. Normes :

- Validation des formulaires côté serveur et côté client.
- Code PHP structuré avec commentaires pour chaque fonction ou classe.

### 5. Sécurité :

- Hachage des mots de passe (password\_hash) et répétition du mot de passe.
- Sécurisation des adresses mail et évitez les doublons.
- Validation des entrées pour éviter les injections SQL (utiliser des requêtes préparées).

## Organisation du projet

### 1. Équipe

- Le projet sera réalisé en groupes de **2 à 3 étudiants**.

## 2. Livrables

- Code source complet à déposer sur **Gitea**.
- La date limite de rendu est le **3 mars à 23h59**. Seul le dernier commit avant cette date sera pris en compte. Chaque jour de retard entrainera des réductions de points.
- Préparer le dépôt Gitea et rédiger un fichier **README.md** pour expliquer comment exécuter le projet.

## 3. Orale

- Une présentation de 10-15 minutes à l'oral **prévu le 4 mars 2025 à partir de 8h45**.

Critères d'évaluation	Points attribués	Description
<b>1. Fonctionnalités principales (8 points)</b>		
Page d'accueil fonctionnelle et esthétique	1	Présentation claire et visuelle de la page d'accueil.
Affichage des produits et détails des articles	1,5	Liste des produits avec fiche produit détaillée (image, description, prix).
Panier fonctionnel	1,5	Ajout, suppression, et total du panier calculé automatiquement.
CRUD complet pour les produits dans le back-office	2	Ajouter, modifier, supprimer et afficher les produits dans l'interface d'administration.
Connexion sécurisée pour l'administration	1	Formulaire sécurisé pour authentifier un administrateur.
Gestion des utilisateurs ou commandes (optionnel)	1	Fonctionnalité supplémentaire de gestion des utilisateurs ou des commandes.
<b>2. Qualité du code (3 points)</b>		
Structuration et organisation	1,5	Code lisible, bien commenté, avec une bonne séparation des fichiers (ex. inclusion des fichiers).
Sécurité	1,5	Utilisation de bonnes pratiques (hachage des mots de passe, protection contre les injections SQL).
<b>3. Base de données (4 points)</b>		
Modèle relationnel cohérent	2	Schéma logique des tables respectant le modèle relationnel.
Tables bien implémentées	2	Types de données adaptés, relations correctement définies (ex. clés primaires/étrangères).
<b>4. Présentation orale (5 points)</b>		
Démonstration fluide des fonctionnalités	1,5	Navigation claire et sans bug entre les différentes pages et fonctionnalités du site.
Explication de l'organisation	1,5	Présentation de l'architecture, des fichiers et du modèle de base de données.
Maîtrise du sujet et réponses aux questions	1	Réponses claires et pertinentes aux questions posées par les évaluateurs.
Clarté et structure de la présentation	1	Présentation bien organisée et engageante (bonne communication et gestion du temps).

**Total : 20 points**