

# Technologies du Web

## Mini-Project : Création d'un Site E-Commerce avec Laravel

---

L'objectif de ce mini-projet est de concevoir un site e-commerce simple en utilisant le framework Laravel. Ce projet permettra de mieux comprendre l'architecture Modèle-Vue-Contrôleur (MVC), les opérations CRUD, l'authentification des utilisateurs, les fonctionnalités essentielles d'un site e-commerce et la collaboration efficace via GitHub. Le travail doit être réalisé en binôme (minimum) ou en trinôme (maximum), et tous les membres doivent appartenir au même groupe de travaux pratiques (TP).

### 1 Exigences du Projet

#### 1.1 Interface Administrateur (Opérations CRUD)

— **Fonctionnalités :**

L'administrateur doit pouvoir :

- ✓ Ajouter de nouveaux produits (**Créer**).
- ✓ Consulter la liste des produits (**Lire**).
- ✓ Modifier les détails d'un produit (**Mettre à jour**).
- ✓ Supprimer des produits (**Supprimer**).

— **Caractéristiques :**

- ✓ Un tableau de bord permettant à l'administrateur de gérer les produits.
- ✓ Des formulaires pour ajouter et modifier des produits (nom du produit, description, prix, image, etc.).
- ✓ Un tableau affichant tous les produits avec des options pour les modifier ou les supprimer.

#### 1.2 Interface Utilisateur

— **Fonctionnalités :**

Les utilisateurs doivent pouvoir :

- ✓ S'inscrire et se connecter.
- ✓ Parcourir une liste de produits.
- ✓ Ajouter des produits à un panier.
- ✓ Consulter leur panier et retirer des articles si nécessaire.
- ✓ Ajouter une barre de recherche pour filtrer les produits.
- ✓ Mettre en place un système de paiement avec une passerelle (ex. : Stripe ou PayPal).

#### 1.3 Exigences Techniques

— **Utilisation de l'architecture MVC de Laravel :**

- ✓ Modèles : Définir des modèles pour les utilisateurs, les produits et les articles du panier.
- ✓ Vues : Créer des templates Blade pour l'interface utilisateur.
- ✓ Contrôleurs : Gérer la logique des actions des utilisateurs (ex. : ajout au panier, gestion des produits).

- **Front-end :**
  - ✓ Peuvent utiliser n'importe quel framework CSS (ex. : Bootstrap, Tailwind CSS), Angular ou React, ou simplement du HTML/CSS/JavaScript pur.
- **Backend (Laravel)**
- **Base de données :**
  - ✓ Vous pouvez utiliser la base de données de votre choix. Exemple (MySQL).
- **Collaboration avec GitHub :**
  - ✓ Utiliser GitHub et Git pour le contrôle de version et la collaboration.
  - ✓ Invitez votre binôme en tant que collaborateur dès le démarrage afin que tout l'historique des modifications soit conservé.
  - ✓ Il ne suffit pas d'importer le projet sur GitHub à la fin ; tout le processus de développement doit être suivi et documenté dans le dépôt dès le début.

## Livrables

- **Code source** : Un projet Laravel bien structuré et documenté, respectant l'architecture MVC. (GitHub)
- **Dépôt GitHub** : Un dépôt partagé avec un historique de commits clair et des pull requests.
- **Rapport détaillé** : Un rapport expliquant en détail vos **choix de conception, défis rencontrés et solutions mises en œuvre**.
- **Présentation** : Une présentation de 10 minutes expliquant votre projet, ses principales fonctionnalités et les défis rencontrés lors du développement (sans PowerPoint, je veux voir le site directement).

## Calendrier

**Date de début du projet** : [12/03/2025]

**Date limite de soumission** : [02/04/2025]

## Critères d'Évaluation

Votre projet sera évalué selon les critères suivants :

- **Fonctionnalité et complétude** des fonctionnalités.
- **Respect de l'architecture MVC**.
- **Qualité du code, organisation et documentation**.
- **Utilisation efficace de GitHub** (historique des commits, pull requests, suivi des tâches).
- **Créativité et expérience utilisateur**.