

Cahier des charges pour le projet E-commerce

Contexte du projet

Ce projet consiste en la conception et le développement d'une application de commerce électronique permettant aux utilisateurs de naviguer, découvrir, et acheter des produits physiques ou numériques. L'application inclut des fonctionnalités de gestion des utilisateurs, des paniers, des commandes, et des avis, avec une interface moderne et intuitive.

Objectifs du projet

1. Permettre aux utilisateurs de créer un compte et de gérer leurs informations personnelles.
 2. Offrir une expérience de navigation fluide pour consulter des produits et leurs détails.
 3. Implémenter un système de gestion de paniers et de commandes.
 4. Assurer la gestion des avis pour les produits afin d'améliorer la confiance des utilisateurs.
 5. Mettre en place un système de facturation.
-

Fonctionnalités principales

Gestion des utilisateurs

- Création, modification et suppression d'un compte utilisateur.
- Connexion et déconnexion sécurisées.
- Gestion des adresses (facturation et livraison).

Produits

- Gestion des produits physiques et numériques via des entités spécifiques (**PhysicalProduct** et **DigitalProduct**).
- Organisation des produits par catégories (**Category**) et par tags (**Tag**).
- Affichage des détails d'un produit (description, prix, disponibilité).

Panier et commandes

- Ajout, modification, et suppression d'articles dans un panier.
- Conversion d'un panier en commande.
- Gestion des articles d'une commande (**OrderItem**).
- Suivi des états de commande (en cours, expédiée, livrée).

Avis

- Ajout d'avis sur les produits par les utilisateurs.
- Affichage des avis pour chaque produit.

Facturation

- Génération et gestion des factures associées aux commandes.
- Liaison des factures à un utilisateur.

Modélisation des données

Entités principales et relations

Relations ManyToMany

1. **Product ⇔ Category**
 - Un produit peut appartenir à plusieurs catégories, et une catégorie peut contenir plusieurs produits.
2. **Product ⇔ Tag**
 - Un produit peut avoir plusieurs tags, et un tag peut être associé à plusieurs produits.

Relations OneToMany

1. **User ⇔ Review**
 - Un utilisateur peut avoir plusieurs avis, mais chaque avis appartient à un seul utilisateur.
2. **Product ⇔ Review**
 - Un produit peut avoir plusieurs avis, mais chaque avis appartient à un seul produit.
3. **Order ⇔ OrderItem**
 - Chaque article de commande est lié à une seule commande, mais une commande peut avoir plusieurs articles.

4. **Cart** ⇔ **CartItem**

- Chaque élément de panier est associé à un seul panier, mais un panier peut avoir plusieurs éléments.

5. **User** ⇔ **Order**

- Un utilisateur peut passer plusieurs commandes, mais chaque commande appartient à un seul utilisateur.

6. **User** ⇔ **Cart**

- Un utilisateur peut avoir plusieurs paniers, mais chaque panier appartient à un seul utilisateur.

7. **User** ⇔ **Invoice**

- Un utilisateur peut avoir plusieurs factures, mais chaque facture est associée à un seul utilisateur.

8. **User** ⇔ **Address**

- Un utilisateur peut avoir plusieurs adresses, mais chaque adresse appartient à un seul utilisateur.

Contraintes techniques

1. **Technologie principale** : Symfony (framework PHP).
2. **Base de données** : MySQL ou MariaDB.
3. **Sécurité** : Gestion des sessions et des données sensibles (hashage des mots de passe, protection CSRF).
4. **Versionnement** : Utilisation de Git pour la gestion du code source.
5. **Déploiement** : Intégration avec Docker pour un environnement de développement cohérent.

Livrables

1. Code source complet et documenté.
2. Schéma de base de données et fichier de migration.
3. Documentation utilisateur et technique.
4. Rapport final présentant les fonctionnalités implémentées et les tests réalisés.

Planning prévisionnel

1. **Phase 1 : Analyse et conception** (2 semaines)

- Compréhension des besoins.
- Création du schéma de base de données.

2. Phase 2 : Développement (4 semaines)

- Implémentation des entités, relations et contrôleurs.
- Intégration du front-end.

3. Phase 3 : Tests et validation (2 semaines)

- Tests unitaires et fonctionnels.

4. Phase 4 : Livraison (1 semaine)

- Documentation et déploiement final