

C'est un projet passionnant ! Pour vous aider à structurer la création de votre application web multi-vendeurs (une marketplace), je vous propose un **Cahier des Charges (CdC)** détaillé, une proposition de **Technologies à utiliser**, et une description de la **Maquette** de l'interface.

Étant donné que je ne peux pas générer directement un fichier PDF, je vais vous fournir le contenu complet ici, que vous pourrez facilement copier et enregistrer sous ce format.

Cahier des Charges Détaillé pour une Marketplace Multi-Vendeurs

1. Introduction et Objectifs du Projet

1.1. Objectifs

- Créer une plateforme web (marketplace) unique regroupant plusieurs commerçants (vendeurs) spécialisés dans les accessoires (cosmétiques, sacs, chaussures, etc.).
- Fournir à chaque vendeur une section dédiée pour gérer et présenter ses produits.
- Permettre aux clients de naviguer, sélectionner, commander et payer des articles auprès de différents vendeurs.
- Assurer une expérience utilisateur fluide et sécurisée.

1.2. Public Cible

- **Vendeurs** : Commerçants indépendants souhaitant vendre leurs accessoires en ligne.
- **Clients** : Utilisateurs finaux à la recherche d'accessoires variés et de produits cosmétiques.
- **Administrateur** : Gestionnaire de la plateforme.

2. Fonctionnalités (F) et Exigences (E)

2.1. Gestion des Comptes et Authentification

Acteur	Fonctionnalité/Exigence	Description Détaillée
Client	F1. Inscription/Connexion	Possibilité de s'inscrire via email/mot de passe ou réseaux sociaux.
Vendeur	F2. Inscription/Connexion	Processus d'inscription spécifique (informations légales, compte bancaire pour les paiements). E1 : Un vendeur est aussi un client (il a accès aux fonctionnalités client).

Acteur	Fonctionnalité/Exigence	Description Détaillée
Général	F3. Profil utilisateur	Modification des informations (nom, adresse, mot de passe).
Admin	F4. Gestion des utilisateurs	Validation, suspension ou suppression des comptes Vendeurs et Clients.

2.2. Espace Vendeur (Gestion du Catalogue)

Acteur	Fonctionnalité/Exigence	Description Détaillée
Vendeur	F5. Tableau de bord	Vue synthétique des ventes, commandes en attente, et articles populaires.
Vendeur	F6. Gestion des Articles	Ajouter, modifier, supprimer des articles. E2 : Champs requis : Nom, description, prix, catégorie (cosmétique, sac, chaussure, etc.), images multiples , stock/quantité.
Vendeur	F7. Gestion des Commandes	Afficher les commandes reçues, changer leur statut (En cours de traitement, Expédié, Livré).
Vendeur	F8. Page Vendeur publique	URL unique pour sa boutique sur la plateforme, affichant tous ses articles.

2.3. Expérience Client (Achat)

Acteur	Fonctionnalité/Exigence	Description Détaillée
Client	F9. Navigation/Recherche	Barre de recherche globale. Filtrage par : Catégorie, Prix, Vendeur .
Client	F10. Page Produit	Détails de l'article (F6), nom et lien vers la boutique du vendeur.
Client	F11. Panier d'Achat	Ajout/Suppression d'articles. E3 : Le panier doit gérer les articles provenant de <i>plusieurs vendeurs différents</i> (et potentiellement des frais de livraison séparés).
Client	F12. Commande et Paiement	Processus de paiement sécurisé. E4 : Gestion de multiples adresses de livraison/facturation.

Acteur	Fonctionnalité/Exigence	Description Détaillée
Client	F13. Historique des commandes	Consultation du statut de ses commandes.

2.4. Paiement et Transaction

Acteur	Fonctionnalité/Exigence	Description Détaillée
Général	F14. Moyens de Paiement	Intégration de services comme Stripe, PayPal ou un système local (paiement mobile).
Général	E5. Sécurité	Utilisation du protocole HTTPS . Conformité PCI DSS pour le traitement des cartes.
Admin/Vendeur	F15. Système de Commission	Le système doit calculer et retenir une commission prédéfinie sur chaque vente avant de transférer le solde au vendeur.

3. Technologies Recommandées

Pour garantir la scalabilité, la sécurité et l'efficacité de l'application, je recommande une architecture moderne de type **MERN** (MongoDB, Express.js, React, Node.js) ou une pile basée sur Python/Django si vous préférez.

3.1. Pile Technique Recommandée (Exemple : MERN Stack)

Couche	Technologie	Rôle
Frontend (Interface Utilisateur)	React (avec Hooks et Redux/Context API pour la gestion d'état)	Application monopage (SPA) rapide et interactive.
Backend (Serveur/API)	Node.js + Express.js	Création d'une API RESTful robuste pour gérer les communications entre le frontend et la base de données.
Base de Données	MongoDB (NoSQL) ou PostgreSQL (SQL)	MongoDB est flexible pour les données produit/catalogue. PostgreSQL est préférable pour la gestion des transactions financières complexes.

Couche	Technologie	Rôle
Païement	Stripe ou Mollie/Flutterwave/Paystack (pour l'Afrique/Europe)	Gestion des paiements multi-vendeurs et des commissions (Stripe Connect est idéal pour cela).
Hébergement	AWS, Google Cloud (GCP) ou DigitalOcean	Plateforme Cloud pour la scalabilité et la haute disponibilité.

3.2. Outils Complémentaires

- **Versionnement** : Git / GitHub.
- **Interface de Style** : **Tailwind CSS** ou **Chakra UI** pour un développement rapide et un design soigné.

4. Maquette de l'Interface Utilisateur (Wireframing)

La maquette doit se concentrer sur l'accessibilité et la clarté.

4.1. Pages Clés

Type de Page	Éléments Clés
Page d'Accueil (Client)	Bannière rotative, Barre de Recherche centrale, Catégories principales (Cosmétiques, Sacs...), Vendeurs en vedette, Section 'Nouveautés'.
Page Vendeur (Publique)	Logo/Nom du Vendeur, Description, Note/Avis (si implémenté), Liste d'articles avec filtres spécifiques au vendeur.
Page Produit	Galerie d'Images multiples, Prix, Nom du Vendeur (cliquable), Bouton "Ajouter au Panier", Fiche technique/Description.
Page Panier	Liste des articles groupés par Vendeur (pour visualiser les frais de livraison séparés), Sous-total, Total, Bouton "Passer la Commande".

4.2. Maquette de l'Espace Vendeur

- **Tableau de Bord** :
 - Graphique des ventes des 30 derniers jours.
 - Statistiques clés (Nombre de commandes en attente, Revenus totaux, etc.).
 - Raccourci vers "Ajouter un article".

- **Gestion des Produits :**

- Tableau listant tous les articles (Nom, Prix, Stock, Vues).
- Boutons d'action : Modifier, Supprimer, Mettre en veille.
- Formulaire d'ajout/modification intuitif avec aperçu de l'image.

Visualisation : Pour vous donner une idée visuelle de la structure.

Prochaine Étape

Ce cahier des charges est une excellente base. Pour aller plus loin, souhaitez-vous que je vous aide à :

1. **Estimer les délais** de développement pour les différentes phases du projet ?
2. **Détailler l'architecture de la base de données** (schéma des tables/collections) ?
3. **Créer un Wireframe plus précis** pour la page produit ou le tableau de bord vendeur ?