

Projet de fin d'études

Titre du projet : Développement d'une plateforme e-commerce unifiée pour la gestion des boutiques en ligne et locales

Établissement : Youcode - UM6P

Étudiant(e) : Yassir ZBIDA

Encadrant(e) : Saad HAIMEUR

Date : 14/01/2025

1. Contexte et objectifs

Contexte

Dans un monde de plus en plus tourné vers le commerce en ligne, les entreprises doivent gérer à la fois leurs boutiques en ligne (via des plateformes comme WooCommerce ou Shopify) et leurs magasins physiques. Une gestion centralisée des produits, des stocks, des commandes et du support client est essentielle pour optimiser les processus et améliorer l'efficacité opérationnelle. Ce projet vise à concevoir une solution unifiée qui répond aux besoins des petites et moyennes entreprises (PME) souhaitant harmoniser la gestion de leurs activités en ligne et locales.

Problématique

Les petites et moyennes entreprises (PME) qui opèrent à la fois en ligne et en magasin physique rencontrent souvent des difficultés pour gérer leurs activités de manière cohérente et efficace. Les principaux problèmes incluent :

1. **Manque de synchronisation des stocks :** Les stocks en ligne et en magasin sont souvent gérés séparément, ce qui entraîne des erreurs de disponibilité et des pertes de ventes.

2. **Gestion fragmentée des commandes** : Les commandes en ligne et en magasin sont traitées via des systèmes distincts, ce qui complique le suivi et la gestion globale.
3. **Absence d'un tableau de bord unifié** : Les entreprises doivent basculer entre plusieurs outils pour gérer leurs activités, ce qui réduit leur productivité et augmente le risque d'erreurs.
4. **Difficulté à générer des rapports consolidés** : L'absence d'une vue centralisée des performances rend difficile l'analyse des ventes et la prise de décisions éclairées.
5. **Support client inefficace** : La gestion des réclamations et des tickets est souvent dispersée entre plusieurs canaux, ce qui nuit à la satisfaction client.

Ce projet vise à résoudre ces problèmes en proposant une plateforme centralisée qui synchronise les données entre les boutiques en ligne et les magasins physiques, tout en offrant un tableau de bord unique pour une gestion simplifiée et efficace.

Objectifs

- Concevoir une plateforme e-commerce centralisée et fonctionnelle.
- Développer un tableau de bord unique pour l'administration des produits, des stocks, des commandes, des paiements et du support client.
- Synchroniser les données entre les boutiques en ligne (WooCommerce, Shopify) et les magasins physiques.
- Automatiser la génération et la gestion des factures.
- Offrir une interface utilisateur intuitive et adaptée à tous les appareils (responsive).
- Intégrer un système de point de vente (POS) pour la gestion des magasins physiques.

2. Description fonctionnelle

2.1. Tableau de bord (Dashboard)

Le tableau de bord doit permettre :

- Une vue synthétique des performances (chiffre d'affaires, commandes, stocks, etc.).
- Une gestion centralisée des modules (produits, stocks, commandes, support client, factures).
- Une synchronisation en temps réel avec WooCommerce et Shopify pour afficher les données des boutiques en ligne.
- Une gestion des ventes en magasin via un système de point de vente (POS).

2.2. Gestion des produits (CRUD)

Fonctionnalités :

- **Synchronisation WooCommerce/Shopify** : Récupération et mise à jour automatique des produits.
- **Créer** : Ajouter des produits avec des champs (nom, description, prix, catégorie, images, etc.).
- **Lire** : Consulter une liste paginée des produits avec recherche et tri.
- **Mettre à jour** : Modifier les informations des produits.
- **Supprimer** : Supprimer des produits avec une confirmation.

2.3. Gestion des stocks

- Mise à jour automatique des stocks après une commande en ligne ou en magasin.
- Alertes sur les niveaux de stock bas.
- Fonction pour ajuster manuellement les niveaux de stock.
- Synchronisation bidirectionnelle avec WooCommerce et Shopify.

2.4. Gestion des commandes

- Historique des commandes (en attente, validées, livrées, annulées).
- Détails de commande : produits achetés, quantité, prix total, coordonnées client.
- Modification du statut des commandes (manuelle ou automatisée).
- Intégration des API WooCommerce et Shopify pour la synchronisation en temps réel.

2.5. Factures et paiements

- Génération automatique des factures PDF après validation d'une commande.
- Archivage des factures pour consultation ultérieure.
- Fonction de téléchargement des factures.
- Prise en charge des systèmes de paiement WooCommerce et Shopify.

2.6. Système de support client

- Les clients peuvent soumettre des tickets via une interface dédiée.
- Les administrateurs peuvent répondre aux tickets via le tableau de bord.
- Suivi des tickets ouverts, en cours de traitement et résolus.

2.7. Point de vente (POS) pour magasins physiques

- Barcode scanning pour les produits.
 - Gestion des ventes en magasin (encaissement, reçus).
 - Synchronisation des ventes avec le tableau de bord central.
-

3. Spécifications techniques

3.1. Technologies proposées

- **Front-end** : React.js (interface utilisateur responsive).
- **Back-end** : Laravel (gestion des API et logique métier).
- **Base de données** : PostgreSQL (gestion des données relationnelles).
- **Serveur** : Hébergement Hostinger.
- **Intégrations** :
 - API REST WooCommerce.
 - API GraphQL Shopify.

3.2. Sécurité

- Authentification sécurisée (OAuth2, JWT).
- Protection des données clients (chiffrement SSL/TLS).
- Règles de permissions (administrateurs, modérateurs, utilisateurs).

3.3. Accessibilité et design

- Interface utilisateur intuitive et responsive (adaptée aux appareils mobiles et desktop).
-

4. Livrables attendus

- Une plateforme fonctionnelle avec tableau de bord intégré.
 - Documentation technique et utilisateur.
 - Présentation du projet (PowerPoint, rapport écrit).
 - Dépôt Git contenant le code source.
-

5. Planning prévisionnel

1. **Phase 1 : Conception et planification** (2 semaines)
 - Définition des cas d'utilisation et des wireframes.
 - Choix des technologies et des API.
2. **Phase 2 : Développement** (8 semaines)
 - Développement des modules principaux (produits, stocks, commandes, POS).
 - Intégration des API WooCommerce et Shopify.
3. **Phase 3 : Tests et validation** (2 semaines)
 - Tests unitaires, tests d'intégration, et tests utilisateurs.
4. **Phase 4 : Déploiement** (1 semaine)
 - Mise en production et formation des utilisateurs.
5. **Phase 5 : Maintenance et améliorations** (en continu)
 - Correction des bugs et ajout de nouvelles fonctionnalités.