**Cahier** **des** **charges** **simplifié**

# 1. Présentation du projet

• Nom du projet : MEDAMINE-STORE33(eCommerce-website)

• Brève description : Développement d’un site e-commerce moderne permettant la vente en ligne de produits variés avec une interface utilisateur intuitive.

• Objectif principal de l’application : Permettre aux utilisateurs de parcourir des produits, les ajouter au panier, et finaliser leur commande en ligne.

# 2. Problèmes à résoudre / besoins exprimés

• Difficultés ou besoins résolus :

- Besoin d’une plateforme de vente accessible 24/7.

- Digitalisation d’une activité commerciale traditionnelle.

- Simplification du processus d’achat pour les clients.

• Attentes de l’utilisateur final :

- Navigation simple et fluide.

- Paiement sécurisé.

- Visualisation rapide des produits.

- Gestion efficace des commandes et des livraisons.

# 3. Utilisateurs cibles

• Administrateur :

- Gère les produits, les commandes, les utilisateurs et le contenu du site.

• Client :

- Parcourt les produits, passe des commandes, suit ses livraisons, et gère son compte utilisateur.

**4. Fonctionnalités attendues**

• Client :

- Créer un compte / se connecter.

- Rechercher et filtrer des produits.

- Ajouter des produits au panier.

- Passer une commande.

- Suivre l’état de ses commandes.

- Gérer son profil utilisateur.

• Administrateur :

- Ajouter / modifier / supprimer des produits.

- Gérer les catégories.

- Gérer les utilisateurs.

- Suivre les commandes.

- Générer des rapports (commandes, ventes, etc.)

# 5. Contraintes techniques

• Technologies imposées :

- Backend : Laravel

- Frontend : React.js

• Base de données prévue :

- MySQL

• Hébergement envisagé :

- Hébergement mutualisé ou VPS (type OVH, Hostinger, etc.)

• Mobilité / Responsiveness :

- Le site devra être responsive : compatible avec ordinateurs, tablettes et smartphones.

Liste des fonctionnalités attendues

**6. Présentation générale**

But de l’application :  
Créer une plateforme e-commerce permettant aux utilisateurs de consulter des produits, passer commande, et aux administrateurs de gérer le catalogue et les ventes.  
  
Utilisateurs cibles :  
- Visiteurs (non connectés)  
- Clients (connectés)  
- Administrateurs

**7. Liste des fonctionnalités par type d’utilisateur**

**-Visiteur (non connecté)**

-Consulter la liste des produits

-Rechercher un produit par nom ou catégorie

-Consulter les détails d’un produit

-S’inscrire / Créer un compte

-Se connecter

**-Client (connecté)**

-Ajouter un produit au panier

-Modifier/supprimer un produit du panier

-Passer une commande

-Consulter l’historique des commandes

-Modifier ses informations personnelles

-Se déconnecter

**-Administrateur**

-Ajouter, modifier ou supprimer un produit

-Gérer les catégories de produits

-Gérer les commandes (valider, annuler, suivre)

-Gérer les utilisateurs (clients)

-Accéder aux statistiques de vente

**8. User Stories (optionnel mais recommandé)**

-En tant que visiteur, je veux voir les produits disponibles, afin de choisir ce qui m’intéresse.

En tant que client, je veux ajouter des produits à mon panier, afin de les commander plus tard.

En tant que client, je veux suivre mes commandes passées, afin de savoir où en est ma livraison.

En tant qu’administrateur, je veux ajouter de nouveaux produits, afin de mettre à jour le catalogue.

En tant qu’administrateur, je veux consulter les commandes, afin de gérer les ventes efficacement.

**9. Priorisation des fonctionnalités**

|  |  |
| --- | --- |
| Fonctionnalité | Priorité |
| Consulter les produits | Essentielle |
| Rechercher un produit | Essentielle |
| S’inscrire / Se connecter | Essentielle |
| Ajouter au panier / Commander | Essentielle |
| Modifier ses informations personnelles | Secondaire |
| Consulter l’historique des commandes | Secondaire |
| Gérer les produits (admin) | Essentielle |
| Gérer les commandes (admin) | Essentielle |
| Statistiques de vente (admin) | Bonus |
| Gestion des utilisateurs (admin) | Secondaire |

**Planification du projet.**

* **Liste des tâches à réaliser**

## Phase 1 : Analyse

- Recueil des besoins fonctionnels (produits, utilisateurs, commandes, etc.)

- Rédaction du cahier des charges

- Définition des rôles utilisateurs (admin, client)

## Phase 2 : Conception

- Conception de la base de données (MCD/MPD)

- Définition des relations entre entités (modèles Laravel)

- Création de wireframes / maquette UI avec Figma

## Phase 3 : Développement Back-end (Laravel)

- Création du projet Laravel

- Mise en place des migrations, modèles et seeders

- Gestion de l’authentification : Laravel Breeze

- Création des contrôleurs (Produits, Clients, Commandes)

- Création des API REST pour chaque entité

## Phase 4 : Développement Front-end (React.js)

- Initialisation du projet React.js via vite

- Création des pages (Accueil, Produits, Panier, Connexion, Admin)

- Connexion à l’API Laravel avec Axios

- Intégration de la gestion du panier et paiement simulé

- Espace client (commandes passées)

- Espace admin (gestion produits / clients)

## Phase 5 : Tests

- Tests unitaires Laravel

- Tests de validation des formulaires (React.js)

- Tests fonctionnels des API

## Phase 6 : Déploiement

- Déploiement back-end : Laravel,php

- Déploiement front-end : React.js

- Configuration SSL, domaine et variables d’environnement

# 2. Estimation de la durée (en jours)

|  |  |
| --- | --- |
| Tâches | Durée estimée |
| Analyse | 1 jour |
| Conception | 1,5 jours |
| Développement Back-end | 3 jours |
| Développement Front-end | 3 jours |
| Tests | 1 jour |
| Déploiement | 1 jour |
| Total estimé | 10,5 jours |

# 3. Estimation des coûts (simplifiée)

|  |  |
| --- | --- |
| Ressources | Coût approximatif |
| Hébergement Laravel (Render) | Gratuit / 7€/mois |
| Nom de domaine (.store) | ~10€/an |
| Browser-navigateur (Front-end) | Gratuit |
| Outils (Figma, GitHub Free) | Gratuit |
| Total estimé (version basique) | ~10 à 20€ |

# 4. Outils nécessaires au développement

- Langages & Frameworks : PHP, Laravel, JavaScript, React.js

- IDE : VS Code

- Design UI : Figma

- Contrôle de version : Git, GitHub

- Gestion API : Postman

- Base de données : MySQL / PHP myadmine

- Client HTTP : Axios

# 5. Outils nécessaires à l’exploitation

- Plateforme d’hébergement Back :

- Plateforme d’hébergement Front : Opera Navigation

- Nom de domaine : Google Domains

- Surveillance / Logs : Logtail, Laravel Telescope

- Sauvegarde base de données : export manuel- fichier json

- Monitoring disponibilité: UptimeRobot (freemium)

**Tableau des utilisateurs et rôles**

Ce tableau présente les différents types d'utilisateurs de l'application e-commerce, leur description, leurs rôles et permissions, ainsi que leur lien avec les technologies Laravel et React.

|  |  |  |  |
| --- | --- | --- | --- |
| Type d'utilisateur | Description | Rôles et permissions | Lien avec les technologies |
| Client | Utilisateur qui achète des produits via la plateforme | - Créer un compte / se connecter - Consulter les produits - Ajouter au panier - Passer une commande - Suivre les commandes - Gérer son profil | Laravel : accès aux routes publiques via middleware 'auth:client' React : affichage des pages utilisateur après connexion |
| Administrateur | Personne en charge de la gestion du site, des produits, et des utilisateurs | - Gérer les produits (CRUD) - Gérer les catégories - Gérer les utilisateurs - Gérer les commandes - Accéder aux statistiques/rapports | Laravel : middleware 'auth:admin' + policies pour les ressources React : routes conditionnelles pour le dashboard admin |