Sujet : Développement d’un mini-clone de Shopify avec Laravel

Titre :

"e-Shop" — Plateforme de création de boutiques en ligne pour vendeurs indépendants

Contexte :

ShopFusion est une plateforme qui permet à des commerçants de créer leur propre boutique en ligne, de gérer leurs produits, leurs commandes et leurs clients.

L’application que vous allez développer est un prototype de base, mais structuré de façon professionnelle, suivant l’architecture MVC de Laravel.

---

Fonctionnalités principales à implémenter :

**1. Authentification multi-rôle (Admin / Vendeur / Client)**

Enregistrement et connexion.

Dashboard spécifique selon le rôle.

**2. Gestion des boutiques**

Un utilisateur "vendeur" peut créer une ou plusieurs boutiques.

Chaque boutique a un nom, un logo, une description, une URL (slug).

**3. Gestion des produits**

Chaque boutique peut gérer ses produits.

Produits : nom, description, image, prix, stock, catégorie.

Categorie: libelle, description, slup

**4. Catalogue côté client**

Affichage public des boutiques et de leurs produits.

Ajout au panier.

**5. Panier et commandes**

Le client peut :

ajouter des produits au panier,

passer une commande,

consulter ses commandes.

**6. Backoffice vendeur**

Voir les commandes reçues.

Gérer l’état d’une commande : En attente, En cours, Expédiée, Annulée.

**7. Tableau de bord (statistiques)**

Vendeur : ventes par produit, total des revenus, commandes en cours.

Admin : utilisateurs inscrits, nombre de boutiques, etc.

Bonus (optionnel ou pour approfondir)

Upload d’images avec stockage Laravel.

Utilisation d’API pour paiement (simulée ou Stripe sandbox).

Notifications par mail ou tableau

Migration  
Parfait ! Voici les **migrations Laravel** pour les modèles suivants d’un mini-clone de Shopify :

* Boutiques (shops)
* Catégories (categories)
* Produits (products)
* Panier (carts, cart\_items)
* Commandes (orders, order\_items)

**✅ 1. Migration : shops**

php artisan make:migration create\_shops\_table

Schema::create('shops', function (Blueprint $table) {

$table->id();

$table->foreignId('user\_id')->constrained()->onDelete('cascade');

$table->string('name');

$table->string('slug')->unique();

$table->string('logo')->nullable();

$table->text('description')->nullable();

$table->timestamps();

});

**✅ 2. Migration : categories**

php artisan make:migration create\_categories\_table

Schema::create('categories', function (Blueprint $table) {

$table->id();

$table->string('name');

$table->foreignId('shop\_id')->constrained()->onDelete('cascade');

$table->timestamps();

});

**✅ 3. Migration : products**

php artisan make:migration create\_products\_table

Schema::create('products', function (Blueprint $table) {

$table->id();

$table->foreignId('shop\_id')->constrained()->onDelete('cascade');

$table->foreignId('category\_id')->nullable()->constrained()->onDelete('set null');

$table->string('name');

$table->text('description')->nullable();

$table->string('image')->nullable();

$table->decimal('price', 10, 2);

$table->integer('stock')->default(0);

$table->timestamps();

});

**✅ 4. Migration : carts**

php artisan make:migration create\_carts\_table

Schema::create('carts', function (Blueprint $table) {

$table->id();

$table->foreignId('user\_id')->constrained()->onDelete('cascade');

$table->timestamps();

});

**✅ 5. Migration : cart\_items**

php artisan make:migration create\_cart\_items\_table

Schema::create('cart\_items', function (Blueprint $table) {

$table->id();

$table->foreignId('cart\_id')->constrained()->onDelete('cascade');

$table->foreignId('product\_id')->constrained()->onDelete('cascade');

$table->integer('quantity')->default(1);

$table->timestamps();

});

**✅ 6. Migration : orders**

php artisan make:migration create\_orders\_table

Schema::create('orders', function (Blueprint $table) {

$table->id();

$table->foreignId('user\_id')->constrained()->onDelete('cascade');

$table->foreignId('shop\_id')->constrained()->onDelete('cascade');

$table->decimal('total', 12, 2);

$table->enum('status', ['pending', 'paid', 'shipped', 'cancelled'])->default('pending');

$table->timestamps();

});

**✅ 7. Migration : order\_items**

php artisan make:migration create\_order\_items\_table

Schema::create('order\_items', function (Blueprint $table) {

$table->id();

$table->foreignId('order\_id')->constrained()->onDelete('cascade');

$table->foreignId('product\_id')->constrained()->onDelete('cascade');

$table->integer('quantity');

$table->decimal('unit\_price', 10, 2);

$table->timestamps();

});