

Atelier 9:

Mise à jour et suppression d'un produit dans Ecomer

1- Récupérer le produit par id dans la méthode 'show', et l'envoie vers la view 'showproduit.blade.php' :

```
public function show(string $id)
{
    $product = Produit::findOrFail($id);
    return view('showproduit', ['product' => $product]);
}
```

2- Création de view 'showproduit' :

```
● houdaifa@HoudaifaPC:~/everything/laravel/Ecommerce$ php artisan make:view showproduit
INFO View [resources/views/showproduit.blade.php] created successfully.

↳ houdaifa@HoudaifaPC:~/everything/laravel/Ecommerce$
```

3- Showproduit :

```
<a href="/" class="back-link">← Retour aux produits</a>

<div class="product-detail-container">
    <div>
        nom }}" class="product-image">
    </div>

    <div class="product-details">
        <h1>{{ $product->nom }}</h1>
        <p class="category">Catégorie: <strong>{{ $product->categorie }}</strong></p>

        @if ($product->sold > 0)
            <div class="discount-badge">Solde: {{ $product->sold }}% de réduction</div>
        @endif

        <p class="price">{{ $product->prix }} DH</p>

        <p class="description">{{ $product->description ?? 'Aucune description disponible.' }}</p>

        <div class="detail-actions">
            <a href="{{ route('articles.edit', $product->id) }}" class="btn btn-edit">
                Modifier
            </a>
            <form action="{{ route('articles.destroy', $product->id) }}" method="POST" style="flex: 1;" onsubmit="return confirm('Voulez-vous vraiment supprimer ce produit ?')">
                @csrf
                @method('DELETE')
                <button type="submit" class="btn btn-delete" style="width: 100%;">
                    Supprimer
                </button>
            </form>
            <a href="/" class="btn btn-back">
                Retour
            </a>
        </div>
    </div>
</div>
```

4- Récupérer le produit par id dans la méthode ‘edit’, et l’envoie vers la view ‘editproduit.blade.php’ :

```
    */
    public function edit(string $id)
    {
        $product = Produit::findOrFail($id);
        return view('editproduit', ['product' => $product]);
    }
}
```

5- Creation de view ‘editproduit’ :

```
● houdaifa@HoudaifaPC:~/everything/laravel/Ecommerce$ php artisan make:view editproduit
  INFO View [resources/views/editproduit.blade.php] created successfully.

◊ houdaifa@HoudaifaPC:~/everything/laravel/Ecommerce$
```

6- Editproduit :

```
<section class="edit-product">
    </div>
    @endif

    <form action="{{ route('articles.update', $product->id) }}" method="POST" enctype="multipart/form-data">
        @csrf
        @method('PUT')

        <div class="form-group">
            <label for="nom">Nom du produit</label>
            <input
                type="text"
                id="nom"
                name="nom"
                value="{{ old('nom', $product->nom) }}"
                required
                placeholder="Ex: Stylo bleu"
            >
            @error('nom')
                <div class="error">{{ $message }}</div>
            @enderror
        </div>

        <div class="form-group">
            <label for="description">Description</label>
            <textarea
                id="description"
                name="description"
                placeholder="Détails du produit..."
                >{{ old('description', $product->description) }}</textarea>
            @error('description')
                <div class="error">{{ $message }}</div>
            @enderror
        </div>

        <div class="form-group">
            <label for="prix">Prix (DH)</label>
            <input
                type="number"
                id="prix"
            >
        </div>
    </form>
```

7- Dans 'editproduit', on utilise la méthode 'put' → on va utiliser la méthode update : on va valider, enregistrer la nouvelle image (si elle existe) dans cloudinary, et mise à jour du produit par des nouvelles informations :

```
public function update(Request $request, string $id)
{
    $product = Produit::findOrFail($id);

    $validated = $request->validate([
        'nom' => 'required|string|max:255',
        'description' => 'nullable|string',
        'prix' => 'required|numeric|min:0',
        'categorie' => 'required|string|max:255',
        'image' => 'nullable|image|mimes:jpeg,png,jpg,gif|max:2048',
        'solde' => 'nullable|integer|min:0|max:100',
    ]);

    try {
        if ($request->hasFile('image')) {
            $cloudinary = new Cloudinary([
                'cloud' => [
                    'cloud_name' => env('CLOUDINARY_CLOUD_NAME'),
                    'api_key' => env('CLOUDINARY_API_KEY'),
                    'api_secret' => env('CLOUDINARY_API_SECRET'),
                ],
            ]);

            $result = $cloudinary->uploadApi()->upload(
                $request->file('image')->getRealPath(),
                ['folder' => 'produits']
            );

            $validated['image'] = $result['secure_url'];
        }

        $product->update($validated);
        return redirect('/')->with('success', 'Produit mis à jour avec succès.');
    } catch (\Exception $e) {
        return back()->withErrors(['error' => 'Erreur lors de la mise à jour: ' . $e->getMessage()]);
    }
}
```

8- Recuperer le produit par id dans la méthode destroy, et le supprimer :

```
public function destroy(string $id)
{
    try {
        $product = Produit::findOrFail($id);
        $product->delete();
        return redirect('/')->with('success', 'Produit supprimé avec succès.');
    } catch (\Exception $e) {
        return back()->withErrors(['error' => 'Erreur lors de la suppression: ' . $e->getMessage()]);
    }
}
```

9- Modification dans Home.blade.php: les liens pour voir et modifier le produit, et le button pour le supprimer (methode ‘delete’) :

```
@if (count($products) > 0)
    <div class="products-section">
        <h3>Tous nos Produits</h3>
        <div class="products-container">
            @foreach ($products as $product)
                <div class="product-card">
                    nom }}" class="product-img">
                    <div class="product-info">
                        <h3>{{ $product->nom }}</h3>
                        <p>{{ $product->description ?? '' }}</p>
                        <div class="product-price">{{ $product->prix }} DH</div>

                    <!-- Action Buttons -->
                    <div class="product-actions">
                        <a href="{{ route('articles.show', $product->id) }}" class="btn btn-view">
                            Voir
                        </a>
                        <a href="{{ route('articles.edit', $product->id) }}" class="btn btn-edit">
                            Modifier
                        </a>
                        <form action="{{ route('articles.destroy', $product->id) }}" method="POST" style="display:inline;">
                            @csrf
                            @method('DELETE')
                            <button type="submit" class="btn btn-delete">
                                Supprimer
                            </button>
                        </form>
                    </div>
                </div>
            @endforeach
        </div>
        <!-- Pagination Links -->
        <div class="pagination">
            {{ $products->links('vendor.pagination.custom') }}
        </div>
    </div>
@endif
```