69. Mejora el servicio del carrito y permite eliminar sus productos con Laravel

Modificar `ProductCartController`:

```php
<?php

namespace App\Http\Controllers;

use App\Models\Cart;
use App\Models\Product;
use App\Services\CartService;
use Illuminate\Http\Request;
use Illuminate\Support\Facades\Cookie;

class ProductCartController extends Controller {
    public $cartService;
    public function __construct(CartService $cartService) {
        $this->cartService = $cartService;
    }

    /**
    * Store a newly created resource in storage.
    * @param \Illuminate\Http\Request $request
    * @param \App\Product $product
    * @return \Illuminate\Http\Response
    */
    public function store(Request $request, Product $product) {
        $cart = $this->cartService->getFromCookieOrCreate();
        $quantity = $cart->products()
                ->find($product->id)
                ->pivot
                ->quantity ?? 0;
        $cart->products()->syncWithoutDetaching([
            $product->id => ['quantity' => $quantity + 1],
        ]);
        $cookie = $this->cartService->makeCookie($cart);

        return redirect()->back()->cookie($cookie);
    }

    /**
    * Remove the specified resource from storage.
    * @param \App\Product $product
    * @param \App\Cart $cart
    * @return \Illuminate\Http\Response
    */
    public function destroy(Product $product, Cart $cart) {
        // Envía el id del elemento que queremos borrar de la relación
        $cart->products()->detach($product->id);
        $cookie = $this->cartService->makeCookie($cart);

        return redirect()->back()->cookie($cookie);
    }
}
?>
```

Modificar `CartService`:

```php
<?php
namespace App\Services;

use App\Models\Cart;
use Illuminate\Support\Facades\Cookie;

class CartService {
    protected $cookieName = 'cart';

    public function getFromCookieOrCreate() {
        $cartId = Cookie::get($this->cookieName);
        $cart = Cart::find($cartId);

        return $cart ?? Cart::create();
    }

    public function makeCookie(Cart $cart) {
        return Cookie::make($this->cookieName, $cart->id, 7 * 24 * 60);
    }
}
?>
```

Modificar `product-card.blade.php`:

```php
<div class="card">
    <img class="card-img-top" src="{{ asset($product->images->first()->path) }}"
height="500">
    <div class="card-body">
        <h4 class="text-right"><strong>${{ $product->price }}</strong></h4>
        <h5 class="card-title">{{ $product->title }}</h5>
        <p class="card-text">{{ $product->description }}</p>
        <p class="card-text"><strong>{{ $product->stock }} left</strong></p>

        @if (isset($cart))
        <form
            class="d-inline"
            method="POST"
            action="{{ route('products.carts.destroy', ['cart' => $cart->id,
'product' => $product->id]) }}"
        >
            @csrf
            @method('DELETE')
            <button type="submit" class="btn btn-warning">Remove From
Cart</button>
        </form>

        @else
        <form
        class="d-inline"
        method="POST"
        action="{{ route('products.carts.store', ['product' => $product->id]) }}"
        >
            @csrf
            <button type="submit" class="btn btn-success">Add To Cart</button>
        </form>

        @endif
```

```
        </div>
    </div>
```