Session Summary — Stock Management Implementation

Date: 2026-01-12

Overview

This report summarizes the work performed to add stock management features to the Ecommerce project and the subsequent tasks (tests, admin UI, migration and logging).

What I implemented

- **Cart validation and model fixes**
  - `Temporale` model updated (fillable, relation)
  - `Customer/CartController` validated additions and updates against variant qty
- **Order flow & stock updates**
  - `Store::store()` wrapped in transaction and validates stock
  - Stock decremented on order details save and logged in `stock_transactions`
- **Migrations & models**
  - Added `alert_qty` to `product_family_attributes`
  - Added `stock_transactions` table and `StockTransaction` model
- **Admin tools**
  - StockController endpoints for adjustments, transactions, and a low-stock view
  - Blade view: `admin/stock/low_stock.blade.php`
- **Tests**
  - Added `tests/Feature/StockManagementTest.php` with cart/checkout tests

Files changed

- Migrations: `2026_01_12_000001_add_alert_qty…`, `2026_01_12_000002_create_stock_transactions_table.php`
- Models: `StockTransaction`, `ProductFamilyAttribute`, `Temporale`
- Controllers: `StockController`, updated `CartController`
- Services: `Store` trait updated
- Views: `admin/stock/low_stock.blade.php`
- Tests: `StockManagementTest.php`

How to run & validate locally

1. `php artisan migrate`
2. `php artisan test --filter=StockManagementTest`
3. Manual checks: add to cart, checkout, visit `/stock/low`

Next recommended tasks

- Add authorization middleware to admin routes
- Expose low-stock alerts on admin dashboard
- Add tests for the stock adjust endpoint and update README

Generated by: GitHub Copilot (Raptor mini (Preview))