**Angular E-Commerce Project Roadmap & Technical Architecture**

**📅 Project Overview**

A full-fledged E-Commerce web application built with **Angular 20.1.3** as the frontend framework and **.NET (ASP.NET Core)** as the backend, with **Microsoft SQL Server** as the database. The goal is to support robust admin/user functionalities, product management, cart/wishlist, order processing, and secure payment flows.

**🚀 Key Objectives**

* Build a scalable, maintainable Angular frontend using modular architecture.
* Touch every major Angular feature to ensure deep learning.
* Integrate secure authentication mechanisms.
* Ensure responsive UI/UX with Angular Material / TailwindCSS.
* Connect to backend APIs (RESTful) for CRUD operations.

**Project Architecture Overview**

We are following a **Modular Feature-based Architecture** to ensure scalability, maintainability, and separation of concerns. This architecture aligns with Angular's best practices:

**Key Architectural Features:**

* **Feature Modules**: Each business domain (e.g., User, Product, Admin) has its own Angular module.
* **Shared Module**: Reusable components, pipes, and directives.
* **Core Module**: Singleton services like AuthService, TokenService, etc.
* **Routing Modules**: Each feature module has its own routing file.
* **Lazy Loading**: Improves performance by loading modules only when needed.
* **State Management**: Using services and RxJS for lightweight state handling.
* **Environment Separation**: dev/prod environments configured.

**/src**

**/app**

**/core → Global services, guards, interceptors**

**/shared → Reusable components, directives, pipes**

**/auth → Login, Register, Google OAuth**

**/layout → Header, Footer, Navigation**

**/user → Profile, Address, Wishlist, Cart**

**/product → Categories, Products, Variants**

**/orders → Orders, History, Tracking**

**/payment → Payment flows (UPI, card, wallet)**

**/admin → Role mgmt, category admins, dashboard**

**/contact → Contact us, feedback**

**/logs → Email/SMS logs, system monitoring**

**Angular Concepts by Module**

**Module 1: User Management**

* **Angular Topics Used:**
  + Reactive Forms (Login, Signup)
  + Template-driven Forms (Contact form)
  + Route Guards (AuthGuard, RoleGuard)
  + Interceptors (JWT token injection)
  + Social Login (Google OAuth2 via External SDK)
  + Custom Validators (email, password strength)
  + Lazy Loading Module: UserModule
* **Features Covered:**
  + User Registration & Login (password + Google)
  + Role-based Routing
  + Profile Management
  + Multi-address UI with CRUD

**Module 2: Product & Category Management**

* **Angular Topics Used:**
  + Components (ProductCard, CategoryTile, ProductDetails)
  + Input/Output (passing data across child-parent)
  + Services for API Integration
  + Custom Pipes (Filter by price, category)
  + Angular Material Grid or Tailwind UI Cards
  + Lazy Loading Module: ProductModule
* **Features Covered:**
  + Category/Subcategory/Product Management
  + Image Upload via Angular File Upload
  + Tag management
  + Category-based Admin Access

**Module 3: Wishlist & Cart Management**

* **Angular Topics Used:**
  + Services with RxJS Subjects for Cart/Wishlist State
  + Route Guards for only logged-in access
  + Inter-Component Communication
  + Template-driven forms (update quantity)
  + Local Storage (optional fallback for guest user cart)
  + Lazy Loading Module: CartModule
* **Features Covered:**
  + Add/Remove from Cart/Wishlist
  + Cart-Wishlist Interoperability
  + Quantity tracking
  + User-specific persistent state

**Module 4: Order Management**

* **Angular Topics Used:**
  + Lifecycle Hooks (ngOnInit, ngOnDestroy)
  + Dynamic Forms (selecting address, payment)
  + Table UI for Order History (Angular Material/Table)
  + Custom Pipes (StatusPipe)
  + Lazy Loading Module: OrderModule
* **Features Covered:**
  + Placing an Order
  + Order History UI
  + Order Detail View
  + Track Status

**Module 5: Payment & Transaction Management**

* **Angular Topics Used:**
  + Reactive Forms (payment forms)
  + External SDK integration (Razorpay/Stripe/PhonePe SDK)
  + Error Handling (global via Interceptor)
  + Lazy Loading Module: PaymentModule
* **Features Covered:**
  + Credit/Debit/UPI-based Checkout
  + Razorpay or PhonePe/Google Pay Integration
  + Payment Status Feedback
  + Retry Mechanism

**Module 6: Admin & Role Management**

* **Angular Topics Used:**
  + Dynamic Routing with Role Guards
  + Admin Dashboard (Charts via Chart.js or Angular Material)
  + Lazy Loading Module: AdminModule
  + Table Management (CRUD Table UI)
* **Features Covered:**
  + Admin Panel with CRUD features
  + Superadmin vs. Category Admin routing
  + User Role Management

**Module 7: Notification, Logs, Communication**

* **Angular Topics Used:**
  + Toast Notifications (ngx-toastr or Angular Material Snackbar)
  + HTTP Logging Interceptor
  + Dynamic Email/SMS Templates using string interpolation
  + Lazy Loading Module: NotificationModule
* **Features Covered:**
  + View Logs (Error, Access)
  + Contact Us Form
  + Admin SMS/Email Template Management

**Module 8: Reporting & Analytics**

* **Angular Topics Used:**
  + Charting Library (ng2-charts, ApexCharts)
  + Data Filtering (using Pipes and Dropdowns)
  + Role Guard
  + Lazy Loading Module: ReportModule
* **Features Covered:**
  + Sales Analytics
  + User Reports
  + Product Click Reports
  + Export Options (CSV, PDF)