**1. Phần Quản Trị (Admin)**

**1.1. Quản Lý Người Dùng**

* **Đăng nhập/Đăng xuất Admin:** Xác thực và phân quyền truy cập.
* **Quản lý tài khoản người dùng:**
  + Xem danh sách người dùng.
  + Thêm, sửa, xóa người dùng.
  + Phân quyền (admin, nhân viên, v.v.).
* **Quản lý vai trò và quyền hạn:** Tạo và quản lý các vai trò với quyền truy cập cụ thể.

**1.2. Quản Lý Sản Phẩm**

* **Thêm/Sửa/Xóa sản phẩm:** Tạo mới, chỉnh sửa thông tin hoặc xóa sản phẩm.
* **Quản lý danh mục sản phẩm:** Tạo, sửa, xóa danh mục và phân loại sản phẩm.
* **Quản lý thương hiệu:** Thêm, sửa, xóa thông tin thương hiệu.
* **Quản lý tồn kho:** Theo dõi và cập nhật số lượng tồn kho.
* **Quản lý hình ảnh sản phẩm:** Tải lên, sửa đổi và xóa hình ảnh sản phẩm.

**1.3. Quản Lý Đơn Hàng**

* **Xem danh sách đơn hàng:** Theo dõi trạng thái đơn hàng (đang xử lý, đã giao, đã hủy, v.v.).
* **Cập nhật trạng thái đơn hàng:** Thay đổi trạng thái đơn hàng như xác nhận, vận chuyển, hoàn thành.
* **Quản lý trả hàng và hoàn tiền:** Xử lý yêu cầu trả hàng, hoàn tiền từ khách hàng.

**1.4. Quản Lý Khách Hàng**

* **Xem thông tin khách hàng:** Chi tiết về khách hàng, lịch sử mua hàng.
* **Quản lý hỗ trợ khách hàng:** Xử lý các yêu cầu, thắc mắc hoặc khiếu nại.

**1.5. Quản Lý Khuyến Mãi**

* **Tạo và quản lý mã giảm giá:** Thêm, sửa, xóa mã giảm giá, thiết lập điều kiện sử dụng.
* **Quản lý chương trình khuyến mãi:** Tạo các chương trình khuyến mãi như mua 1 tặng 1, giảm giá theo phần trăm.

**1.6. Quản Lý Nội Dung**

* **Quản lý trang tĩnh:** Chỉnh sửa nội dung các trang như giới thiệu, liên hệ, chính sách.
* **Quản lý blog hoặc bài viết:** Thêm, sửa, xóa bài viết để thu hút khách hàng và SEO.

**1.7. Báo Cáo và Thống Kê**

* **Báo cáo doanh số:** Theo dõi doanh thu theo thời gian, sản phẩm, danh mục.
* **Báo cáo tồn kho:** Kiểm tra số lượng hàng tồn kho, sản phẩm bán chạy.
* **Phân tích khách hàng:** Thống kê về hành vi mua hàng, khách hàng tiềm năng.

**1.8. Cấu Hình Hệ Thống**

* **Cài đặt cổng thanh toán:** Thiết lập các phương thức thanh toán như PayPal, Stripe, chuyển khoản ngân hàng.
* **Cài đặt vận chuyển:** Thiết lập các phương thức vận chuyển, phí vận chuyển.
* **Cài đặt bảo mật:** Quản lý SSL, bảo mật API, xác thực đa yếu tố (MFA).

**1.9. Quản Lý Thông Báo**

* **Quản lý mẫu email/SMS:** Tạo và chỉnh sửa các mẫu thông báo gửi tới khách hàng.
* **Thiết lập thông báo hệ thống:** Cấu hình các thông báo cho nhân viên quản trị.

**1.10. Bảo Mật và Giám Sát**

* **Nhật ký hoạt động:** Theo dõi và ghi lại các hoạt động của người dùng admin.
* **Quản lý truy cập:** Giới hạn quyền truy cập theo vai trò và địa chỉ IP.

**2. Trang Đích (Landing Page) Cho Khách Hàng**

**2.1. Đăng Ký và Đăng Nhập Người Dùng**

* **Đăng ký tài khoản:** Tạo tài khoản mới với email, mật khẩu.
* **Đăng nhập/Đăng xuất:** Xác thực người dùng.
* **Khôi phục mật khẩu:** Tính năng đặt lại mật khẩu qua email.

**2.2. Hiển Thị Danh Mục và Sản Phẩm**

* **Danh mục sản phẩm:** Hiển thị các danh mục chính và phụ.
* **Trang danh sách sản phẩm:** Hiển thị sản phẩm theo danh mục, sắp xếp, phân trang.
* **Trang chi tiết sản phẩm:** Mô tả chi tiết, hình ảnh, đánh giá, liên quan.

**2.3. Giỏ Hàng và Thanh Toán**

* **Giỏ hàng:** Thêm, sửa, xóa sản phẩm trong giỏ hàng.
* **Quy trình thanh toán:** Nhập thông tin giao hàng, chọn phương thức thanh toán.
* **Xác nhận đơn hàng:** Hiển thị thông tin đơn hàng và trạng thái.

**2.4. Quản Lý Tài Khoản Người Dùng**

* **Thông tin cá nhân:** Cập nhật thông tin cá nhân, địa chỉ giao hàng.
* **Lịch sử đơn hàng:** Xem lại các đơn hàng đã đặt.
* **Danh sách yêu thích (Wishlist):** Lưu trữ sản phẩm yêu thích.

**2.5. Tìm Kiếm và Bộ Lọc**

* **Tìm kiếm sản phẩm:** Tính năng tìm kiếm theo từ khóa.
* **Bộ lọc nâng cao:** Lọc theo giá, danh mục, thương hiệu, đánh giá.

**2.6. Đánh Giá và Bình Luận**

* **Đánh giá sản phẩm:** Cho phép người dùng đánh giá và đưa ra số sao.
* **Bình luận:** Cho phép người dùng bình luận về sản phẩm.

**2.7. Nội Dung và SEO**

* **Trang tĩnh:** Giới thiệu, liên hệ, chính sách bảo mật, điều khoản sử dụng.
* **Blog hoặc bài viết:** Cung cấp thông tin hữu ích, tăng cường SEO.
* **Tối ưu hóa SEO:** Sử dụng URL thân thiện, meta tags, schema markup.

**2.8. Responsive Design**

* **Thiết kế đáp ứng:** Đảm bảo giao diện hiển thị tốt trên mọi thiết bị (desktop, tablet, mobile).

**2.9. Bảo Mật và Hiệu Suất**

* **Bảo mật dữ liệu:** Sử dụng HTTPS, bảo vệ chống lại các lỗ hổng bảo mật phổ biến.
* **Tối ưu hóa tốc độ tải trang:** Sử dụng lazy loading, tối ưu hóa hình ảnh, sử dụng CDN.

**2.10. Thông Báo và Email**

* **Thông báo qua email:** Xác nhận đơn hàng, cập nhật trạng thái đơn hàng, khuyến mãi.
* **Thông báo đẩy (Push Notifications):** Cập nhật thông tin nhanh đến người dùng.

**2.11. Tích Hợp Mạng Xã Hội**

* **Chia sẻ sản phẩm:** Cho phép người dùng chia sẻ sản phẩm lên các mạng xã hội.
* **Đăng nhập qua mạng xã hội:** Đăng nhập bằng Facebook, Google, v.v.

**3. Tính Năng Chung**

**3.1. Thiết Kế Kiến Trúc**

* **API RESTful:** Thiết kế API rõ ràng, tuân thủ nguyên tắc REST.
* **Quản lý phiên bản API:** Sử dụng versioning cho API để dễ dàng mở rộng và bảo trì.
* **Bảo mật API:** Sử dụng JWT hoặc OAuth2 cho xác thực và phân quyền.

**3.2. Cơ Sở Dữ Liệu**

* **Thiết kế schema hiệu quả:** Sử dụng các bảng liên quan, chỉ mục hợp lý để tối ưu truy vấn.
* **Sao lưu và phục hồi dữ liệu:** Thiết lập các cơ chế sao lưu định kỳ.

**3.3. Kiểm Thử và Đảm Bảo Chất Lượng**

* **Unit Testing:** Viết các bài kiểm thử đơn vị cho cả backend và frontend.
* **Integration Testing:** Kiểm thử tích hợp giữa các module.
* **End-to-End Testing:** Kiểm thử toàn diện từ đầu đến cuối quy trình mua hàng.

**3.4. Triển Khai và CI/CD**

* **Continuous Integration/Continuous Deployment:** Thiết lập pipeline CI/CD để tự động hóa quá trình build, test và deploy.
* **Containerization:** Sử dụng Docker để đóng gói ứng dụng, dễ dàng triển khai trên các môi trường khác nhau.
* **Orchestration:** Sử dụng Kubernetes hoặc các công cụ tương tự để quản lý container.

**3.5. Giám Sát và Logging**

* **Giám sát hệ thống:** Sử dụng các công cụ như Prometheus, Grafana để theo dõi hiệu suất và tình trạng hệ thống.
* **Logging:** Thu thập và quản lý log để dễ dàng phát hiện và khắc phục sự cố.

**3.6. Tài Liệu và Hướng Dẫn**

* **Tài liệu API:** Sử dụng Swagger hoặc OpenAPI để tạo tài liệu API tự động.
* **Hướng dẫn sử dụng:** Tạo các tài liệu hướng dẫn cho người dùng và quản trị viên.

**3.7. Tính Năng Mở Rộng**

* **Tích hợp với các dịch vụ bên ngoài:** Ví dụ như tích hợp với hệ thống quản lý kho, ERP, CRM.
* **Hỗ trợ đa ngôn ngữ và đa tiền tệ:** Cung cấp trải nghiệm cho người dùng quốc tế.

**4. Các Nguyên Tắc Nâng Cao Trình Code**

**4.1. Kiến Trúc Modular và Tách Rời**

* **Separation of Concerns:** Phân tách rõ ràng giữa các tầng (presentation, business logic, data access).
* **Reusable Components:** Tạo các component tái sử dụng được trong Angular.
* **Microservices (nếu cần):** Xem xét sử dụng kiến trúc microservices để tăng khả năng mở rộng.

**4.2. Sử Dụng Design Patterns**

* **Singleton, Factory, Repository:** Áp dụng các mẫu thiết kế phù hợp để giải quyết các vấn đề phổ biến.
* **Dependency Injection:** Sử dụng DI để quản lý phụ thuộc, dễ dàng kiểm thử.

**4.3. Quản Lý Phụ Thuộc và Phiên Bản**

* **Version Control:** Sử dụng Git để quản lý mã nguồn, tuân thủ các quy tắc commit rõ ràng.
* **Dependency Management:** Quản lý các thư viện và phiên bản một cách hợp lý để tránh xung đột.

**4.4. Tối Ưu Hiệu Suất**

* **Caching:** Sử dụng caching cho các dữ liệu thường xuyên truy cập để giảm tải hệ thống.
* **Lazy Loading:** Tải các module Angular khi cần thiết để giảm thời gian tải trang.
* **Database Optimization:** Tối ưu các truy vấn SQL, sử dụng indexing hợp lý.

**4.5. Bảo Mật**

* **Xác thực và phân quyền:** Đảm bảo chỉ người dùng có quyền mới có thể truy cập các tính năng nhất định.
* **Bảo vệ dữ liệu nhạy cảm:** Sử dụng mã hóa cho thông tin quan trọng như mật khẩu, thông tin thanh toán.
* **Kiểm tra và phòng chống các lỗ hổng bảo mật:** Sử dụng các công cụ kiểm tra bảo mật và áp dụng các best practices.

**4.6. Tài Liệu và Hướng Dẫn**

* **Code Documentation:** Viết tài liệu cho mã nguồn để dễ dàng hiểu và bảo trì.
* **Comment Code:** Đặt các chú thích rõ ràng cho các phần code phức tạp.
* **README và Hướng dẫn Setup:** Cung cấp hướng dẫn chi tiết về cách thiết lập môi trường phát triển và triển khai.

**5. Công Cụ và Công Nghệ Đề Xuất**

**5.1. Backend (Spring Boot)**

* **Spring Security:** Quản lý bảo mật và phân quyền.
* **Spring Data JPA:** Quản lý truy cập dữ liệu.
* **Hibernate:** ORM framework.
* **Swagger/OpenAPI:** Tạo tài liệu API.
* **Lombok:** Giảm boilerplate code.

**5.2. Frontend (Angular)**

* **Angular Material hoặc Bootstrap:** Thiết kế giao diện người dùng.
* **RxJS:** Quản lý luồng dữ liệu không đồng bộ.
* **NgRx (nếu cần):** Quản lý trạng thái ứng dụng.
* **TypeScript:** Sử dụng TypeScript để tăng tính an toàn của mã nguồn.

**5.3. Công Cụ Hỗ Trợ**

* **Postman:** Kiểm thử API.
* **Docker:** Đóng gói ứng dụng.
* **Jenkins/GitHub Actions:** Thiết lập CI/CD.
* **Prometheus & Grafana:** Giám sát hệ thống.
* **ELK Stack (Elasticsearch, Logstash, Kibana):** Quản lý và phân tích log.

**6. Các Bước Triển Khai và Phát Triển**

1. **Phân Tích Yêu Cầu:**
   * Xác định rõ các yêu cầu chức năng và phi chức năng.
   * Tạo tài liệu yêu cầu chi tiết.
2. **Thiết Kế Kiến Trúc:**
   * Lên sơ đồ kiến trúc tổng thể.
   * Thiết kế database schema.
3. **Phát Triển Backend:**
   * Thiết lập dự án Spring Boot.
   * Phát triển các API RESTful.
   * Tích hợp bảo mật và phân quyền.
4. **Phát Triển Frontend:**
   * Thiết lập dự án Angular.
   * Xây dựng các component và module.
   * Kết nối với backend qua API.
5. **Kiểm Thử:**
   * Viết và chạy các bài kiểm thử.
   * Sửa lỗi và tối ưu mã nguồn.
6. **Triển Khai:**
   * Sử dụng Docker để đóng gói ứng dụng.
   * Thiết lập CI/CD để tự động hóa quá trình deploy.
   * Triển khai lên các dịch vụ cloud như AWS, Azure, hoặc Heroku.
7. **Giám Sát và Bảo Trì:**
   * Theo dõi hiệu suất hệ thống.
   * Cập nhật và nâng cấp khi cần thiết.
   * Xử lý các vấn đề phát sinh.