**E-Ticaret Uygulaması Adımları**

1. **Proje Yapısının Belirlenmesi**
2. **Kullanıcı Yönetimi**
3. **Ürün Kataloğu**
4. **Ürün Arama ve Filtreleme**
5. **Alışveriş Sepeti ve Ödeme İşlemleri**
6. **Sipariş Yönetimi**
7. **Yönetici Paneli**
8. **Güvenlik ve Performans**
9. **Test ve Devreye Alma**

**1. Proje Yapısının Belirlenmesi**

**Teknolojiler:**

* **Backend:** ASP.NET Core
* **Frontend:** React, Angular veya Blazor
* **Veritabanı:** SQL Server veya PostgreSQL
* **Diğer:** Docker (geliştirme ve dağıtım için), Git (versiyon kontrol)

**Adımlar:**

* Proje için temel bir klasör yapısı oluşturun.
* Backend ve frontend için ayrı klasörler yapın (backend, frontend).
* Dockerfile ve docker-compose.yml dosyalarını oluşturun.

**2. Kullanıcı Yönetimi**

**Özellikler:**

* Kullanıcı kaydı ve kimlik doğrulama (login, logout)
* Kullanıcı profili yönetimi (bilgi güncelleme, şifre değiştirme)

**Teknolojiler:**

* ASP.NET Core Identity (kullanıcı yönetimi için)
* JWT (JSON Web Token) ile kimlik doğrulama

**Adımlar:**

* ASP.NET Core Identity'yi projeye ekleyin ve konfigüre edin.
* Kullanıcı kayıt ve giriş API'lerini oluşturun.
* JWT ile kimlik doğrulama mekanizması kurun.
* React veya Angular'da kullanıcı arayüzünü oluşturun.

**3. Ürün Kataloğu**

**Özellikler:**

* Ürün ekleme, düzenleme ve silme
* Ürün detayları ve açıklamaları
* Ürün resimleri

**Adımlar:**

* Ürün modelini ve veritabanı tablolarını oluşturun.
* Ürün yönetim API'lerini yazın.
* Ürün listeleme ve detay sayfalarını frontend'de oluşturun.
* Ürün resimleri için dosya yükleme işlevini ekleyin.

**4. Ürün Arama ve Filtreleme**

**Özellikler:**

* Ürün arama fonksiyonu
* Fiyat, kategori ve diğer özelliklere göre filtreleme

**Adımlar:**

* Arama ve filtreleme API'lerini yazın.
* Frontend'de arama ve filtreleme arayüzünü oluşturun.

**5. Alışveriş Sepeti**

**Özellikler:**

* Kullanıcıların ürünleri sepete eklemesi ve sepeti yönetmesi
* Sipariş özeti ve onayı**.**
* Sepet yönetimi API'lerini yazın.
* Frontend'de sepet ve ödeme sayfalarını oluşturun.

**6. Sipariş Yönetimi**

**Özellikler:**

* Siparişlerin oluşturulması ve yönetilmesi
* Sipariş durumu takibi
* Kullanıcı sipariş geçmişi

**Adımlar:**

* Sipariş modelini ve veritabanı tablolarını oluşturun.
* Sipariş yönetim API'lerini yazın.
* Kullanıcıların sipariş geçmişini görüntülemesi için arayüz oluşturun.

**7. Yönetici Paneli**

**Özellikler:**

* Ürün yönetimi (ekleme, düzenleme, silme)
* Sipariş yönetimi
* Kullanıcı yönetimi

**Adımlar:**

* Yönetici rolleri ve yetkilerini belirleyin.
* Yönetici paneli için API'leri yazın.
* Frontend'de yönetici panelini oluşturun.

**8. Güvenlik ve Performans**

**Özellikler:**

* Kullanıcı verilerinin korunması
* SQL enjeksiyonuna karşı koruma
* Veritabanı optimizasyonları

**Adımlar:**

* ASP.NET Core'da güvenlik özelliklerini etkinleştirin.
* API'leri güvenli hale getirin.
* Performans izleme ve iyileştirme araçları kullanın.

**9. Test ve Devreye Alma**

**Özellikler:**

* Unit test ve integration test yazımı
* CI/CD pipeline oluşturma
* Docker ile deployment

**Adımlar:**

* Unit ve integration testlerini yazın.
* GitHub Actions veya Azure DevOps gibi araçlarla CI/CD pipeline oluşturun.
* Docker ile uygulamanızı konteynerize edin ve deploy edin.