

**ANKARA ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**



BLM4531 PROJE RAPORU

Mini E-Ticaret UYGULAMASI

Muhammed Eren Köseoğlu

22290604

Github:https://github.com/Mamyy25/mini_eticaretuygulamasi

Youtube:<https://youtu.be/LphBxx3dJVo>

1. Projenin Genel Özeti ve Amacı

Bu projenin temel amacı; günümüz web teknolojileri kullanılarak, kullanıcıların ürünleri inceleyip sipariş verebildiği, yöneticilerin ise sistem içeriğini yönetebildiği, istemci-sunucu (client-server) mimarisine dayalı işlevsel bir e-ticaret platformu geliştirmektir.

Proje, "Mini E-Ticaret Uygulaması" adı altında, modern yazılım geliştirme prensiplerine sadık kalınarak tasarlanmıştır. Sistem, sadece bir ön yüz (frontend) uygulaması değil, aynı zamanda veri tutarlığını sağlayan bir veritabanı yapısı ve dış dünya ile iletişim kurabilen RESTful API servislerini de barındıran tam kapsamlı bir ağ uygulamasıdır.

Projenin akademik hedefi; ağ üzerinden veri alışverişinin sağlanması, kullanıcı oturum yönetiminin (session management) kurgulanması ve dinamik içerik yönetim sistemlerinin çalışma mantığının uygulamalı olarak gösterilmesidir.

2. Geliştirilme Süreci

Proje, planlı bir takvim çerçevesinde 8 haftalık bir süreçte, modüler bir yapı izlenerek geliştirilmiştir. Süreç, temel mimarinin kurulmasından başlayarak güvenlik testlerinin yapılmasıyla son bulmuştur.

2.1. Planlama ve Veritabanı Mimarisi (1-2. Hafta)

- Proje Yapısının Kurulması:** ASP.NET Core MVC projesi oluşturulmuş, genel klasör yapısı ve gerekli kütüphaneler projeye dahil edilmiştir.
- Veritabanı Tasarımı:** Entity Framework Core kullanılarak "Code-First" yaklaşımı ile veritabanı modelleri oluşturulmuştur. Ürün (`Products`), Kategori (`Categories`), Kullanıcı (`Users`) ve Sipariş/Sepet (`Carts`, `CartItems`) tabloları ve bu tablolar arasındaki "One-to-Many" (Bire-Çok) ilişkiler tanımlanmıştır.

2.2. Backend Geliştirme ve Servisler (3. ve 5. Hafta)

- Servislerin Geliştirilmesi:** Sunucu tarafında veri işlemlerini yönetecek `ProductController` ve `CartController` gibi temel denetleyiciler kodlanmıştır.
- Sepet Mantığı:** Kullanıcıların ürün ekleme, miktar güncelleme ve sepetten ürün çıkarma işlemleri için backend lojiği kurulmuştur. Sepet verileri veritabanı ile senkronize çalışacak şekilde ayarlanmıştır .

2.3. Frontend ve Arayüz Tasarımı (4. Hafta)

- **Kullanıcı Arayüzü:** Razor Views motoru kullanılarak dinamik HTML sayfaları oluşturulmuştur. Tasarımın modern ve responsive (mobil uyumlu) olması için Tailwind CSS kütüphanesinden yararlanılmıştır.
- **Dinamik Bileşenler:** Sepet üzerindeki ürün sayısını anlık gösteren "ViewComponent" yapıları (CartCount) arayüze entegre edilmiştir .

2.4. Güvenlik, Yönetim Paneli ve Test (6-8. Hafta)

- **Kimlik Doğrulama (Authentication):** Kullanıcıların sisteme güvenli giriş yapabilmesi için Session-based (oturum tabanlı) kimlik doğrulama mekanizması kurulmuştur.
- **Admin Paneli:** Sadece yetkili kullanıcıların erişebileceği, ürün ekleme/silme/güncelleme (CRUD) işlemlerinin yapıldığı yönetim paneli geliştirilmiştir.
- **API ve Dokümantasyon:** Dış sistemlerin veri çekebilmesi için RESTful API uçları yazılmış ve bu uçların test edilmesi için Swagger UI entegrasyonu tamamlanmıştır .

3. Kullanılan Teknolojiler ve Altyapı

Proje, ölçüklenebilirlik ve performans göz önünde bulundurularak modern teknoloji yığını üzerine inşa edilmiştir.

3.1. Backend Teknolojileri

- **Framework:** ASP.NET Core 8.0 MVC.
- **Dil:** C# programlama dili.
- **API:** RESTful mimarı ile tasarlanmış servisler ve OpenAPI standartlarında dokümantasyon (Swagger).

3.2. Veritabanı ve ORM (Object-Relational Mapping)

- **Veritabanı Yönetim Sistemi:** SQL Server kullanılmıştır. Veriler ilişkisel tablolar halinde (örneğin; Carts ve CartItems arasında ID tabanlı ilişki) tutulmaktadır .
- **Entity Framework Core:** Veritabanı işlemleri için ham SQL sorguları yazmak yerine, nesne tabanlı programlama imkanı sunan EF Core kullanılmıştır. Bu sayede SQL Injection gibi güvenlik açıklarına karşı koruma sağlanmış ve veri manipülasyonu LINQ sorguları ile (örneğin; .Where, .Include) gerçekleştirilmiştir .

3.3. Frontend Teknolojileri

- **Görüntüleme Motoru:** Razor Views (.cshtml) ile sunucu tarafı render işlemleri yapılmıştır.
- **Stil Kütüphanesi:** Arayüz tasarımında Tailwind CSS kullanılarak hızlı ve modern bir görselleştirme sağlanmıştır.

4. Projenin Kullanım Amacı ve Elde Edilen Kazanımlar

4.1. Kullanım Senaryoları

Uygulama iki temel rol üzerinden hizmet vermektedir:

- Son Kullanıcılar:** Sisteme kayıt olup giriş yaparak ürünleri kategorilere göre filtreleyebilir, detaylarını inceleyebilir, sepete ekleyebilir ve sipariş oluşturabilirler.
- Yöneticiler (Admin):** Özel yönetim paneli aracılığıyla stok takibi yapabilir, yeni ürün ekleyebilir, mevcut ürünlerin fiyatlarını güncelleyebilir veya satıştan kaldırabilirler.

4.2. Elde Edilen Kazanımlar

Bu proje sürecinin tamamlanmasıyla birlikte aşağıdaki yetkinlikler kazanılmıştır:

- Ağ Tabanlı Sistem Mimarisi:** İstemci ve sunucu arasındaki HTTP istek-yanıt döngüsünün ve veri taşıma işlemlerinin (JSON/HTML) pratik uygulaması yapılmıştır.
- Full-Stack Geliştirme:** Hem veritabanı tasarıımı, hem backend kodlaması hem de arayüz giydirmesi yapılarak uçtan uca proje geliştirme deneyimi elde edilmiştir.
- Web Güvenliği:** Oturum yönetimi ve yetkilendirme (Authorization) mekanizmaları öğrenilmiştir.

5. Sonuç

"Mini E-Ticaret Uygulaması" projesi kapsamında, modern web standartlarına uygun, genişletilebilir ve modüler bir e-ticaret platformu başarıyla hayatı geçirilmiştir. Proje sonunda, kullanıcıların sorunsuz bir şekilde alışveriş yapabildiği, yöneticilerin ise sistemi kolayca yönetebildiği, veritabanı ile tam entegre çalışan bir sistem ortaya çıkmıştır.

Bu çalışma, teorik bilgilerin pratiğe dökülmesi açısından önemli bir adım olmuş; web tabanlı yazılım geliştirme süreçlerine, katmanlı mimariye ve ağ tabanlı sistemlerin çalışma prensiplerine hakimiyet sağlanmıştır. Proje, gelecekte ödeme sistemleri entegrasyonu ve daha gelişmiş raporlama modülleri ile geliştirilmeye açık bir altyapıya sahiptir.