

Bu ödev, **Web Programlama** dersinin teorik ve pratik olarak öğretilen bilgilerini gerçek bir probleme uygulamak amacıyla bir **kuaför işletme yönetim uygulaması** geliştirilmesini içeriyor. Ödevin detaylarına ve iş ortamına dair bilgi aşağıdaki gibi özetlenebilir:

İş Ortamı

Bu proje, kuaför veya berber salonlarının günlük operasyonlarını dijital olarak yönetmek için bir uygulama geliştirilmesini hedefliyor. Uygulama, aşağıdaki özelliklere sahip bir platform oluşturmayı amaçlıyor:

Salon Yönetimi: .i

Kuaför/berber salonlarının çalışma saatlerini, hizmetlerini, sürelerini ve ücretlerini tanımlama. ○

Salonların sunduğu işlemleri detaylandırma. ○

Çalışan Yönetimi: .ii

Çalışanların uzmanlık alanları ve uygunluk saatleri ile birlikte sisteme eklenmesi. ○

Çalışanlara göre müşteri randevusu ayarlama. ○

Randevu Sistemi: .iii

Kullanıcıların uygun çalışanlarla işlem ve zaman seçerek randevu almasını sağlama. ○

Çift rezervasyonları engelleyen uyarı sistemleri. ○

REST API Kullanımı: .iv

Veritabanı ile iletişim sağlamak için bir bölümde REST API entegrasyonu. ○

Yapay Zeka Özelliği: .v

Fotoğraf yükleyerek kullanıcıya saç modeli veya saç rengi önerilerinde bulunan yapay zeka özelliği. ○

Nasıl Yapılır?

Projeyi yaparken izlenecek adımlar şu şekildedir:

1. Teknoloji Seçimi ve Kurulum

- ASP.NET Core MVC (6 veya üstü), C#, SQL Server veya PostgreSQL.
- Bootstrap temaları ve HTML5, CSS3, Javascript gibi ön yüz teknolojileri.
- Entity Framework Core ile ORM kullanımı.

2. Proje Planlaması

- **Github Projesi:** Projenizi bir GitHub reposunda başlatın, ilerlemeleri düzenli olarak yükleyin. En az 10 farklı commit yapılması gerekmektedir.

- **Proje Dökümanı:** Başlangıçta kısa bir tanım dokümanı hazırlayın, ardından ekran görüntüleri ve veritabanı modellemeleri gibi detayları ekleyerek güncel tutun.

3. Özelliklerin Uygulanması

- **Veritabanı Modelleme:** Salon, çalışan ve randevu detaylarını içeren bir veritabanı yapısı kurun.
- **UI Tasarımı:** Kullanıcı dostu bir arayüz geliştirin, özellikle kullanıcı kayıt/giriş ve randevu ekranlarına özen gösterin.
- **API Geliştirme:** REST API ile veritabanı işlemleri yapın. Örneğin, çalışanların uygunluk durumlarını sorgulamak için API kullanabilirsiniz.
- **Yapay Zeka Entegrasyonu:** Kullanıcı fotoğrafını analiz ederek öneriler sunan bir yapay zeka modeli (örneğin, hazır bir API kullanımı) entegre edin.

4. Test ve Değerlendirme

- Projeyi tamamladıktan sonra test edin, olası hataları giderin.
- **GitHub'da Aktivite:** Düzenli commit yaparak geliştirme sürecinizi belgeleyin.

5. Teslim ve Sunum

- Hazırlanan raporda proje tanıtımı, veritabanı modellemesi ve ekran görüntüleri yer almalı.
- Proje SABİS'e yüklenecek ve sınıf sunumu sırasında hem proje hem de yazılan kodlar anlatılacaktır.