

WEB KUAFÖR PROJESİ

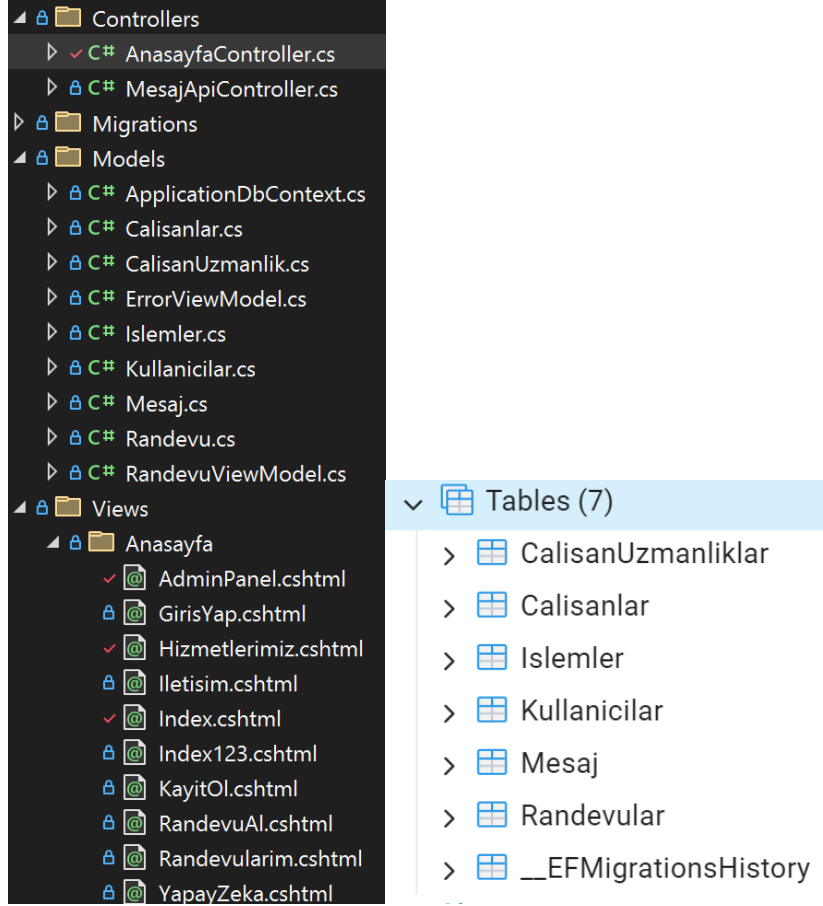
AYŞE VERDA GÜLCEMAL

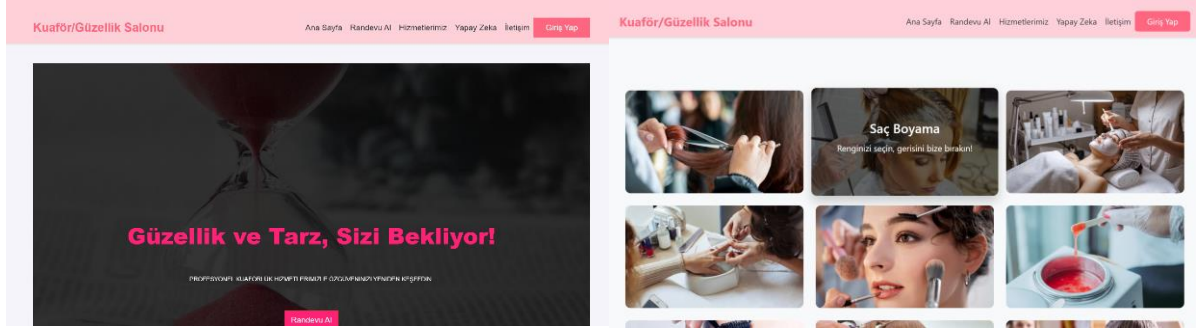
G221210038

2.öğretim B grubu

<https://github.com/ayseverda/kuafor-odevi-son>

Projenin dosya yapısı ve veri tabanı tablolarım aşağıdaki gibidir.





Projemin anasayfasında ve hizmetlerimiz sayfasında işlevsellikten çok müşteriye çekme odaklı **bootstrap** tasarımları kullandım. Tüm sayfalara uyumlu menü yapımı da **layouta** ekledim.

Giriş yapmak için kullanıcı model sınıfı kullandım.

KullanıcıID [PK] integer	AdSoyad text	Email text	Sifre text	IsAdmin boolean
2	admin	ayseverda@gmail.com	123	true
4	ibrahim	ibos@gmail.com	123	false

Kullanıcılar tablosunda id, ad, mail, sifre ve isadmin bool değerlerimiz var. Admini de kullanıcılar tablosunda tanımladım. Kullanıcı eğer admin ise true döndürüyor ve giriş yaptığında admin paneline ulaşabiliyor.

```
// Kullanıcı bilgilerini session'a kaydet
HttpContext.Session.SetInt32("KullanıcıID", kullanıcı.KullanıcıID);
HttpContext.Session.SetString("AdSoyad", kullanıcı.AdSoyad);
HttpContext.Session.SetString("Email", kullanıcı.Email);

// Kullanıcı rolünü belirle
if (kullanıcı.IsAdmin)
{
    HttpContext.Session.SetString("UserRole", "Admin");
    return RedirectToAction("AdminPanel"); // Admin için yönlendirme
}
else
{
    HttpContext.Session.SetString("UserRole", "User");
    return RedirectToAction("Index"); // Müşteri için yönlendirme
}
```

Kullanıcı bilgilerini doğru girdiği takdirde bilgiler **sessiona** kaydediliyor. Böylece oturum açılmış oluyor. Daha sonra admin mi kontrolü yapılarak gerekli sayfaya yönlendiriliyor.

Admin Panel

İşlem Yönetimi

Çalışan Yönetimi

Randevu Onayı

Kullanıcı Yönetimi

Verimlilik

Mesajlar

İşlem Yönetimi

Kuaför işlemlerini ekle, düzenle veya sil.

İşlem ID

ID (varsa)

İşlem Adı

İşlem adı

Süre (dk)

Süre

Ücret (TL)

Ücret

Açıklama

Açıklama

Ekle

Güncelle

Sil

#	İşlem Adı	Süre (dk)	Ücret (TL)	Açıklama
1	Saç Kesimi	45	300	kesim
2	Saç Boyama	90	800	boya

Admin giriş yaptığıında

adminpaneli viewine yönlendiriliyor. Burada adminin yönetim sayfalarını tek viewde yazabilmek için sekmelere ayırdım.

25	merve	Kaş-Bıyık	Sule Kara	24.12.2024 13:00:00	Onaylı	Onaylı
27	ibrahim	Kaş-Bıyık	Sule Kara	1.01.2025 12:12:00	Red	Red
39	berra	Saç Modelleri	İlke Duran	26.12.2024 12:12:00	Bekliyor	Onayla Red

Randevu onayı sayfasında admin kullanıcıların randevularını onaylama ve reddetme yetkisine sahip.

Çalışan Verimliliği

Aylık performans verileri:

Pelin Zuhul

Ezgi Mulan

Beyza Kara

Ayse Kendir

Yaren Beyaz

Sule Kara

İlke Duran

İbrahim Üzümlü

Mete Koran

Tuğçe Duygu

Mete Koran
Mete Koran: 1750 TL

Verimlilik sayfası ile çalışanların aylık kazançlarını bir daire grafiği üzerinde gösterdim.

```
localhost:7223/api/mesaj

Okunaklı hale getir

[
  {
    "mesajID": 7,
    "musteriAd": "ibrahim",
    "email": "ibos@gmail.com",
    "konu": "dsf",
    "aciklama": "xc",
    "tarih": "2024-12-22T23:51:47.348372Z"
  },
  {
    "mesajID": 8,
    "musteriAd": "ibrahim",
    "email": "ibos@gmail.com",
    "konu": "dsf",
    "aciklama": "edwa",
    "tarih": "2024-12-22T23:53:21.955489Z"
  },
  {
    "mesajID": 9,
    "musteriAd": "ibrahim",
    "email": "ibos@gmail.com",
    "konu": "dsfas",
    "aciklama": "sa",
    "tarih": "2024-12-24T23:28:20.324289Z"
  }
]
```

Mesajlar				
Gelen mesajları görüntüleyin.				
#	Gönderen	Konu	Mesaj	Tarih
7	ibrahim	dsf	xc	23.12.2024 02:51:47
8	ibrahim	dsf	edwa	23.12.2024 02:53:21

Mesajları veritabanından **restApi** ile çektim bunun için mesajlara özel bir controller oluşturdum. Aynı zamanda bu controller içinde çeşitli **linq** sorguları yaptım.

```
.Select(m => new
```

```
[Route("api/mesaj")]
[ApiController]
```

Randevu Bilgilerini Girin

Ad Soyadı:

bera

İşlem Seçin:

Saç Kesimi

Çalışan Seçin:

Çalışan Seçin

Çalışan Seçin

Ayşe Kendir (Müsait)

Yaren Beyaz (Müsait)

Mete Koran (Müsait)

Tuğçe Duygu (Müsait)

Randevularım

Sadece kendi adınıza randevu alabilirsiniz.

Randevu saatleri 09:00 - 18:00 arasında olmalıdır.

Seçtiğiniz çalışan bu saat aralığında başka bir randevuda.

Randevu sayfasında kullanıcılar sadece kendi isimleri üzerinden, giriş yaparak randevu alabiliyorlar.

Aynı zamanda her gün çalışma saatleri olan 09-18 aralığında randevu alınabiliyor. Günün tarihinden başlayarak 2 hafta içinde randevu oluşturulabiliyor.

Çakışma durumları kontrol edilerek öyle bir durum olduğunda uyarı veriyor.

```
var islem = await _context.Islemler
    .FirstOrDefaultAsync(i => i.IslemID == islemID);
```

Buradaki kısıtlama işlemlerini **Linq** kullanarak yaptım

Saç Stili Önerisi

Fotoğraf Yükle

Dosya Seç

Dosya seçilmedi

Saç Stili Önerisi Al

Önerilen Saç Stili:

Bu fotoğraftaki kişiye yakışacak bir saç stili önerisi olarak, belirgin katmanlara sahip bir uzun dalgalı saç modeli önerebilirim. Bu stilde saçlar omuz hizasından aşağı doğru uzun ve dalgalı olacak şekilde kesilmiş olmalı. Bu model kişinin yüz hatlarını yumuşatırken ve doğal görünüm sağlarken aynı zamanda da modern bir hava katacaktır. Eğer bu tarz bir saç modeline karar vererseniz, kuaförünüz ile detayları konuşarak

Son olarak **yapay zeka apisi** ile kullanıcılar kendi fotoğraflarını atıp saç stili önerileri alabilir!

Çalışanlar tablom:

CalisanID [PK] integer	CalisanAd text	UzmanlikID integer	Musaitlik boolean	GunlukKazanc numeric (18,2)
1	Tuğçe Duygu	2	true	200
2	Ezgi Mulan	2	true	0.00

İşlemler tablom:

IslemID [PK] integer	IslemAd text	IslemSuresi integer	Ucret numeric	Tanim text
1	Saç Kesimi	45	300	kesim
2	Saç Boyama	90	800	boya
3	Protez Tırnak	60	500	Protez tırnak yapımı

Kullanıcılar tablom:

KullaniciID [PK] integer	AdSoyad text	Email text	Sifre text	IsAdmin boolean
2	admin	ayseverda@gmail.com	123	true
4	ibrahim	ibos@gmail.com	123	false

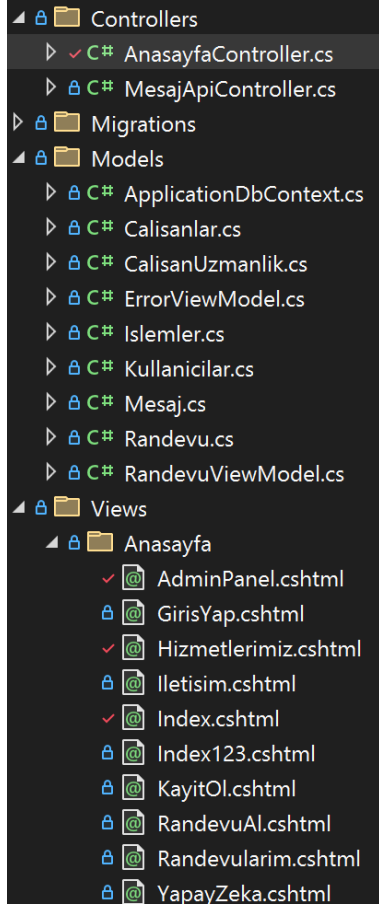
Randevular tablom:

RandevuID [PK] integer	KullaniciID integer	IslemID integer	CalisanID integer	RandevuZamani timestamp with time zone	RandevuBitisZamani timestamp with time zone	Durum text
17	4	9	9	2024-12-23 15:00:00+03	2024-12-23 15:50:00+03	Onaylı
18	5	8	8	2024-12-23 17:30:00+03	2024-12-23 17:55:00+03	Onaylı

CalisanID [PK] integer	IslemID [PK] integer
1	1
1	2

Çalışanların yapabildikleri işlemleri tanımlayan tablo. Böylece birçok işlemi bir kullanıcıya atayabildim.

Ödevi yaparken en zorlandığım kısım randevu kısmındaki çalışan-işlem bağlantılarını ayarlamaktı. Başlangıçta işlemi çalıştırarak çalışan tablosunda fk olarak çektiğim için id alıyor ve her bir çalışanı tek bir işlemle eşleştirebiliyordum. Daha sonra çalışanuzmanlık tablosuyla düzelttim. Genel olarak en zorlandığım kısım randevular ve ona bağlı kısımlardı. Veritabanımda hata yaptıkça düzeltmelerde de çok zorlandım. Migration işlemleri ve entity framework eklemek başlangıçta biraz zorladı. Çok fazla hata alıyordum ama sonrasında oturttum ve düzeldi.



AnasayfaController : METOTLAR :

- **MesajGonder(string adSoyad, string email, string konu, string mesaj):**
 - Girdi doğrulaması yapar (boş alan kontrolü).
 - Kullanıcıyı e-posta adresine göre veritabanında arar.
 - Kullanıcı varsa, mesajı veritabanına kaydeder ve kullanıcıyı ilişkilendirir.
 - Kullanıcı yoksa hata mesajı gösterir.
- **GirisYap(string email, string sifre):**
 - Kullanıcıyı e-posta adresine göre veritabanında arar.
 - Şifre doğrulaması yapar.
 - Başarılı girişte, kullanıcı bilgilerini ve rolünü oturuma kaydeder.
 - Kullanıcı rolüne göre (Admin veya User) ilgili sayfaya yönlendirir.
 - Hatalı giriş durumunda hata mesajı gösterir.
- **KayitOl(string adSoyad, string email, string sifre, string sifreOnay):**
 - Şifre ve şifre onayı eşleşmesini kontrol eder.
 - Yeni kullanıcıyı veritabanına kaydeder.
 - Varsayılan olarak, yeni kullanıcılar "Admin" rolüne sahip değildir.
 - Başarılı kayıta, kullanıcıyı giriş sayfasına yönlendirir.
 - Hata durumunda hata mesajı gösterir.
- **YapayZeka(IFormFile photo):**
 - Kullanıcı tarafından yüklenen fotoğrafı alır.

- Fotoğrafı Base64 formatına çevirir.
- OpenAI API'sine fotoğraf ve bir prompt göndererek saç stili önerisi alır.
- Öneriyi ViewBag.Suggestion ile görünümüne aktarır.
- Hata durumunda hata mesajı gösterir.
- **GetCalisanlar(int islemID):**
 - Belirli bir işlem ID'sine sahip çalışanları getirir.
 - CalisanUzmanliklar ve Calisanlar tablolarını birleştirir.
 - Sonucu JSON formatında döndürür.
- **RandevuAl(string adSoyad, int islemID, int calisanID, DateTime randevuZamani):**
 - Kullanıcıların randevu almasını sağlar.
 - Oturum açmış kullanıcının kendi adına randevu aldığını doğrular.
 - İşlem ve çalışan bilgilerini doğrular.
 - Randevu zamanının salon çalışma saatleri içinde olup olmadığını kontrol eder.
 - Seçilen çalışan için randevu zamanının müsait olup olmadığını kontrol eder.
 - Randevunun 2 hafta içinde olup olmadığını kontrol eder.
 - Randevuyu veritabanına kaydeder ve çalışanın günlük kazancını günceller.
 - Başarılı veya başarısız işlem sonucunu kullanıcıya bildirir.
- **IslemEkleGuncelle, CalisanEkleGuncelle, KullaniciEkleGuncelle, RandevuOnayRed :**
 - Yönetici panelindeki CRUD (Create, Read, Update, Delete) işlemlerini yönetir.
 - IslemEkleGuncelle metodu ile işlemler eklenir, güncellenir ve silinir.
 - CalisanEkleGuncelle metodu ile çalışanlar eklenir, güncellenir ve silinir. Ayrıca çalışanın uzmanlıkları da güncellenir.
 - KullaniciEkleGuncelle metodu ile kullanıcılar eklenir, güncellenir ve silinir.
 - RandevuOnayRed metodu ile randevular onaylanır veya reddedilir.

AdminPanel:

- Tüm işlemleri (Islemler tablosu)
- Tüm çalışanları, uzmanlıkları ve ilgili işlemleriyle birlikte (Calisanlar tablosu, ilişkili tablolarla)
- Tüm mesajları (Mesaj tablosu)
- Tüm kullanıcıları (Kullanicilar tablosu)
- Tüm randevuları, ilişkili kullanıcı, işlem ve çalışan bilgileriyle birlikte (Randevular tablosu, ilişkili tablolarla)

Veritabanından çeker.

- Her çalışanın, o ay içinde aldığı randevulardan elde ettiği toplam kazancı hesaplar.
- Çekilen ve hesaplanan tüm verileri ViewBag aracılığıyla görünüme (view) aktarır.