**Web Tabanlı Teknolojiler Proje Ödevi Raporu**

22253838

Saida Temurpularova

**Proje Senaryosu:** Projem basit Yemek Tarifleri uygulaması. Kullanıcı girişi yada kaydı gerçekleştikten sonra ana sayfa açılır. Ana sayfada kıta isimleri fotoğrafları ile birlikte butonlar şeklinde dizilir. Herhangi kıta butonuna tıklandığında o kıtaya ait ülkeler listelenir. Her ülkeye tıklandığında o ülkede bulunun yemek türleri butonlar şeklinde sıralanacak(örneğin ana yemek, çorba, tatlı...). Yemek türü seçildikten sonra seçilen yemek türüne ait yemeklerin listesi gelecek. Kullanıcı herhangi yemeğe tıkladığında da o yemeğin fotoğrafı, tarifi ve video linkinin bulunduğu yeni sayfa açılacak.

**Projenin Backend Kısmı**

Yukarıdaki senaryoyu gerçekleştirmek için 5 tablo oluşturdum bunlar User, Continent, Country, MealType ve Meal tabloları. Bu tablolar üzerinde işlemler yapabilmek için de Controller sınıflarını oluşturdum.Controller sınıflarında Post ve Put isteklerini gerçekleştirebilmek için kullanıcının veri girmesini sağlayan Dto sınıflarını oluşturdum.

**Tablolar arasında kurduğum ilişkiler**: Continent ve Country arasında 1🡪N

Country ve MealType arasında N🡪M

MealType ve Meal arasında 1🡪N

**Oluşturduğum Get İstekleri:**

ContinentsController sınıfında sadece bir tane tüm kıtaları getiren Get isteği olusturdum getAllContinents(). İçine herhangi parametre almıyor.

CountriesController sınıfında tüm Country leri ve continentId ye gore Country getiren Get istekleri oluşturdum. GetCountriesByContinent(int continentId) .İstek içine parametre olarak continentId alıyor.

MealTypesController sınıfında toplam üç istek oluşturdum ama sadece ikisini React kısmında kullandım. GetMealTypesByCountry(int countryId). Parametre olarak countryId alıyor yani ülkeye gore o ülkede olan yemek türlerini veriyor. İkinci Get isteği GetMealTypeById(int id) parametre olarak mealTypId alıyor yani hangi id de hangi yemek türü olduğunu gösteriyor. Üçüncü istekte basit tüm yemek türlerini getiren Get isteği.

MealsController sınıfında beş tane Get isteği oluşturdum GetAllMeals() bu istek veritabanında bulunan tüm yemeklerin id’sini Name’ini ve MealTypeId’sini listeler. GetMealDetails(int mealId) isteği parametre olarak mealId alıyor ve yemeğin detaylarını gosterir. GetMealsByCountryId(int countryId) isteği parametre olarak countryId alıyor ve ülkede bulunan yemeklerin Name’ini ve MealId’sini listeler. GetMealsByMealTypeId(int mealTypeId) isteği parametre olarak mealTypeId alıyor ve yemek türlerine göre yemeklerin Name’ini ve MealId’sini listeliyor. Proje senaryoma göre bu istekler yetersiz geldi o yuzden yeni bir tane istek oluşturdum gereksiz istekleri de silmedim. GetFilteredMeals([FromQuery] int? countryId, [FromQuery] int? mealTypeId) bu istek hem countryId yi hemde MealTypeId yi ayni anda control ediyor ikisininde var olmasi durumunda yemeği listeye ekliyor.

Aynı sınıfta birden fazla Get isteği oluştururken isteklerin portlarini farklı olarak belirtmeye dikkat ettim.

**Oluşturduğum Post İstekleri:**

Her Controller sınıfı için kendi Dto sınıfında belirtilen entityleri karşılayacak şekilde Post isteği gerçekleştirdim.

Diğer Controller sınıflarından farklı olarak AuthenticationController sınıfı sadece 3 tane Post isteğınden oluşuyor Bunlar Login(LoginDto dto) Register(UserDto dto) ve Logout() istekleridir.

Login isteği parametre olarak bir Username ve Password dan oluşan LoginDto objesi alıyor. Yaptığı işlem şu şekilde once dto dan gelen Name’i Users tablosunda olup olmadığını control ediyor eğer öyle bir Name varsa Password’u önceden belirlenen HashPassword’a çeviriyor ve Users tablosundaki Pasword ile karşılaştırıyor. Karşılaştırmalar doğru çıkarsa token oluşturuyor. Register aldığı UserDto yu veritabanına ekliyor aldığı password HashPassword ile şifreliyor. Logout token in zamanını bitiriyor.

Diğerlerinden farklı olarak AddMealType(int countryId, CountryMealTypeDto dto) post isteğini çoka çoklu ilişkiyi tanımlamak için ekledim. Bu istek migrationla oluşturduğum ara tabloya veri ekliyor.

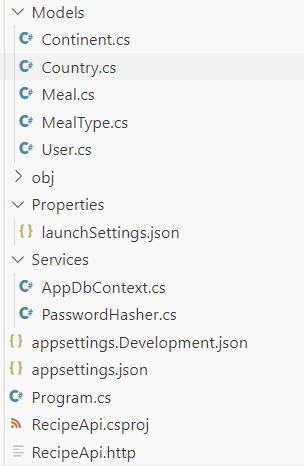
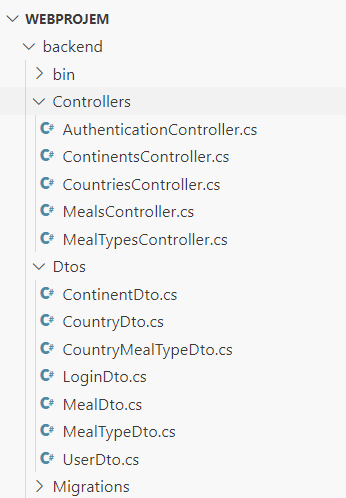
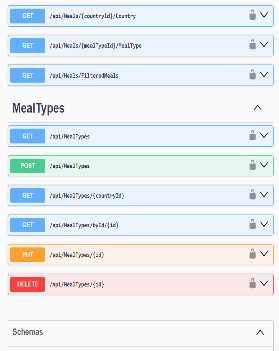
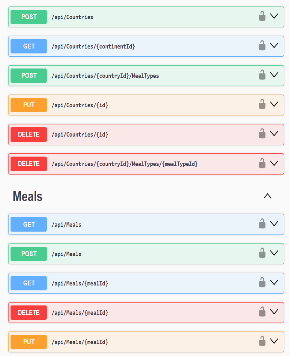
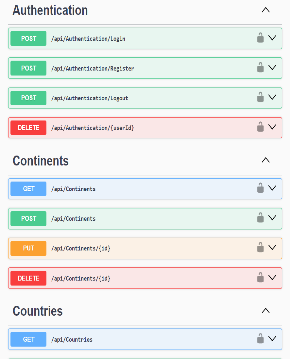
**Oluşturduğum Put İstekleri:**

Her kontroller sınıflarına tum eklenen verileri değiştirebilecek şekilde birer tane Put isteği oluşturdum. AuthenticationController sınıfı hariç bu sınıf içerisinde herhangi Put isteği bulunmuyor.

**Oluşturduğum Put İstekleri:**

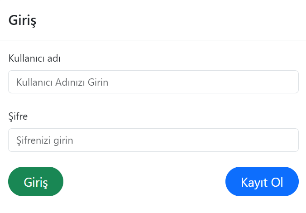
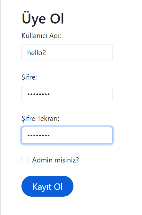
Her kontroller sınıflarına eklenen verileri silecek şekilde birer tane Delete isteği oluşturdum. Sadece CountriesController sınıfına ilişkili olduğu yemek türünü silmesi için fazladan Delete isteği oluşturdum DeleteMealType(int countryId, int mealTypeId).

Program.cs dosyasinda veritabanı ekleme, token oluşturma ve Cors için gerekli kontrolleri gerçekleştirdim. Appssettings.json sınıfında veritabanımı tanımladım.

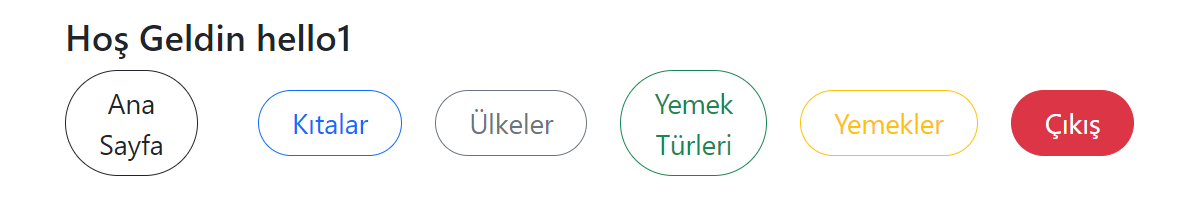
** **

**Pojenin Frontend Kısmı**

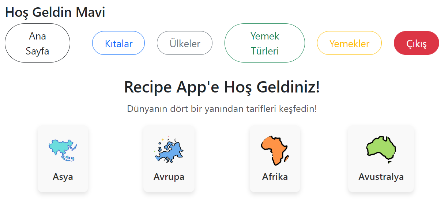
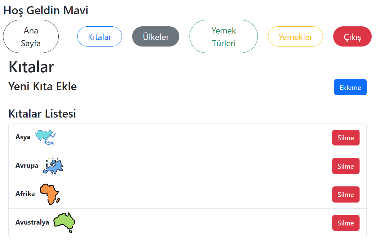
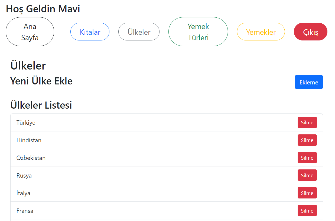
Frontend kısmında sayfa yüklendiğinde kullanıcı giriş sayfasi yükleniyor ve kullanıcının giriş yapmasını istiyor. Kullanıcının kaydı yoksa kullanıcı kayıt ol butonuna basarak kayıt yol sayfasına yönlendirelecek. Kayıt veya giriş işlemi gerçekleşince kullanıcı ana sayfaya yönlendiriliyor.

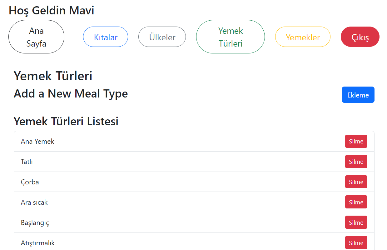
 

Ana sayfayı anlatmadan önce her sayfada bulunan butonlardan bahsetmek istiyorum. Her sayfanın üst kısmında Hoş Geldin KullanıcıIsmi şeklinde başlık bulunacak. Bu başlığın altında Ana Sayfa, Kıtalar, Ülkeler, Yemek Türleri, Yemekler ve Çıkış butonları bulunacak.

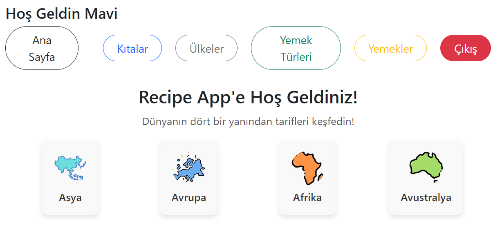
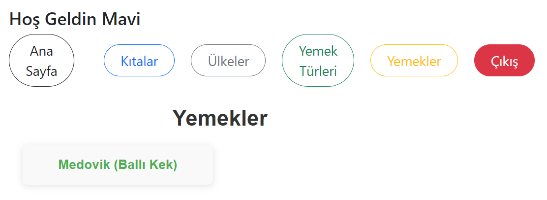
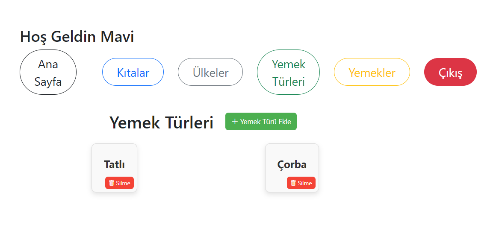
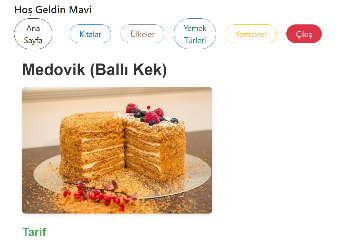
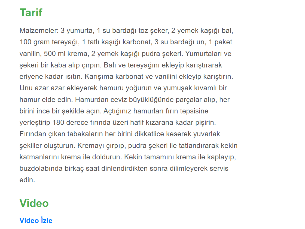


Ana sayfa butonuna tıklandığında kıtaların listelendiği sayfa açılacak.Kıtalar butonuna tıklandığında kıtaların listelendiği yenı sayfa açılacak. Bu sayfada kıtaların isimleri buluncak ve her kıta isminin yanında silme butonu bulunacak ve kıta eklemek için kıta ekleme butonu bulunacak. Ekleme butonuna tıklnadığında kullanıcıdan kıta ismi ve fotograf url ini isticek. Ülkeler butonuna tıklandığında tüm ülkelerin listesinin sıralandığı sayfa açılacak. Bu sayfada da kıtalar sayfasında olduğu gibi ekleme ve silme işlemleri yapabilmek için butonlar bulunacak. Yemke Türleri ve Yemekler butonuna tıklandığında da kullanıcı yukarıdaki işlemleri gerçekleştiren sayfalara yönlendirılecek.

Şimdi Ana Sayfadan bahsetmek istiyorum Ana sayfada kıta isimleri fotoğrafları ile birlikte butonlar şeklinde listeleniyor. Kıta butonuna tıklandığında o kıtaya ait ülkelerin listelendiği sayfaya yönlendirme yapılıyor. Listelenen ülkeler yıne butonlar şeklinde oluyor ve herhangi ülkeye tıklandığında o ülkede bulunan yemk türlerinin listelendiği sayfaya yönlendirme yapıyor. Yemek türleri sayfasında bir ülekeye ait yemek türleri butonlar şeklinde listeleniyor ve bu sayfada ilgili ülkeye yemek türü ekleme silme butonları bulunuyor. Bu butonlar ülke ile yemek türü arasında çoka çoklu ilişki kuruyor yada ilişkiyi kaldırıyor. Bir yemek türü butonuna tıklandığında yemklerin listelendiği sayfaya yönlendirme yapılıyor. Yemekler sayfasında ait olduğu yemek türü ve ülkeye göre yemekler yine butonlar şeklinde listeleniyor. Herhangi yemeğe tıklandığında o yemeğin detaylarının bulunduğu sayfaya yönlendirme yapılıyor. Yemek detaylarının bulunduğu sayfada yemeğin adı fotoğrafı tarifi ve video linki aşağıdan yukarıya doğru sıralanmış şekilde geliyor.

   ****  ****

**Video**

**https://drive.google.com/file/d/1zeaHFWiutE0PQtAFOM1tL-WBSLhGq9c2/view?usp=sharing**