

MYAZ203 – NESNE TABANLI PROGRAMLAMA LAB.

UYGULAMA 8

❖ Eğitmen: Doç. Dr. Zafer CÖMERT

E-posta: zcomert@samsun.edu.tr

❖ Yardımcı Öğretim Elemanı: Arş. Gör. Furkançan DEMİRCAN

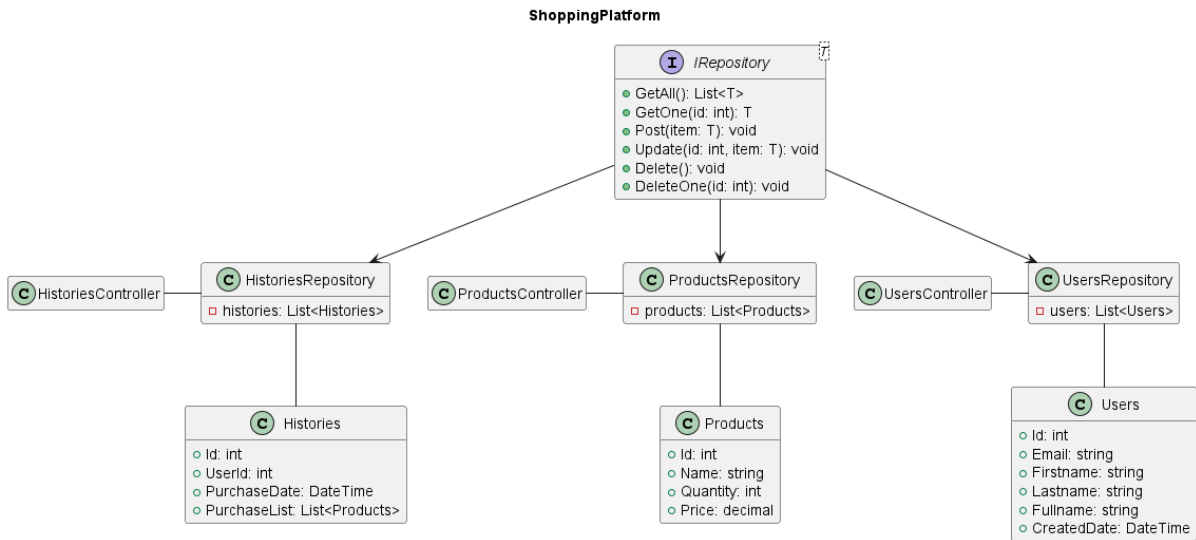
E-posta: furkancan.demircan@samsun.edu.tr

Adım 1: Uygulamanın oluşturulması

- Daha önce oluşturduğumuz **myaz203205** isimli klasörün içerisinde yer alan “**week10**” .bat dosyasını terminalde çalıştırınız.
- Çalıştırma sonucunda uygulama kapsamında kullanacağınız projeler sizin oluşturulacaktır.

Adım 2: Genel sınıf diyagramı

- Lab. kapsamında yapılacak uygulama diyagram olarak Şekil 1’de sunulmuştur.



Şekil 1. Genel sınıf diyagramı

- **ShopEntities** içerisinde;
 - “**Users**”, “**Products**” ve “**Histories**” sınıfları “**ShopEntities**” içerisinde oluşturacağınız “**Models**” isimli klasörde yazılmalıdır. Her sınıfın kendine ait .cs dosyası olmalıdır.
- **ShopRepositories** içerisinde;
 - “**IRepository**” sınıfı “**generic**” bir yapıdır. Bu sınıfı “**ShopRepositories**” içerisinde oluşturacağınız “**Repositories**” klasöründe tanımlayınız. Bu sınıfta diyagramda verilen metotlar bulunmalıdır.
 - “**HistoriesRepository**”, “**ProductsRepository**” ve “**UsersRepository**” sınıfları “**IRepository**” interface yapısını kalıtım ile kullanılmalıdır. Bu sınıflarda, veriler oluşturacağınız listelerde tutulmalıdır.

➤ **ShopApi** içerisinde;

- Diyagramda belirtilen “*Controller*” yapılarını “*ShopApi*” içerisinde yer alan “***Controller***” klasörü içerisinde tanımlayınız. Bu sınıflar “*ControllerBase*” sınıfını kalıtım ile kullanmalıdır.
- Controller yapıları HTTP metodlarını yerine getirmelidir.
- “*Program.cs*” içerisinde Şekil 2’de belirtilen kodları ekleyerek verilerin kaybolması sağlanmaktadır.

```
builder.Services.AddSingleton<ProductsRepository>();  
builder.Services.AddSingleton<UsersRepository>();  
builder.Services.AddSingleton<HistoriesRepository>();  
  
builder.Services.AddSingleton<List<Products>>();  
builder.Services.AddSingleton<List<Histories>>();  
builder.Services.AddSingleton<List<Users>>();
```

Şekil 2. Singleton kullanımı

Adım 2: Oluşturulan metodları **Swagger** sayfasında test ediniz.

Kazanımlar

Bu bölüm öğrencimiz tarafından doldurulacaktır.

- Generic sınıf ile normal sınıf arasındaki farkları açıklayabilirim.
- Generic sınıfı projelerimde kullanabilirim.
- Generic yapısını açıklayabilirim.

Github: <https://github.com/FurkancanDemircan>