

JWT Projeme Dahil Etmek İçin Neler Yaptım ?

1 – Users, OperationClaims, UserOperationClaims Tablolarımı Veritabanında oluşturdum.

```
[PasswordHash] VARBINARY (500) NOT NULL,  
[PasswordSalt] VARBINARY (500) NOT NULL
```

 olmasına mutlaka dikkat edin.

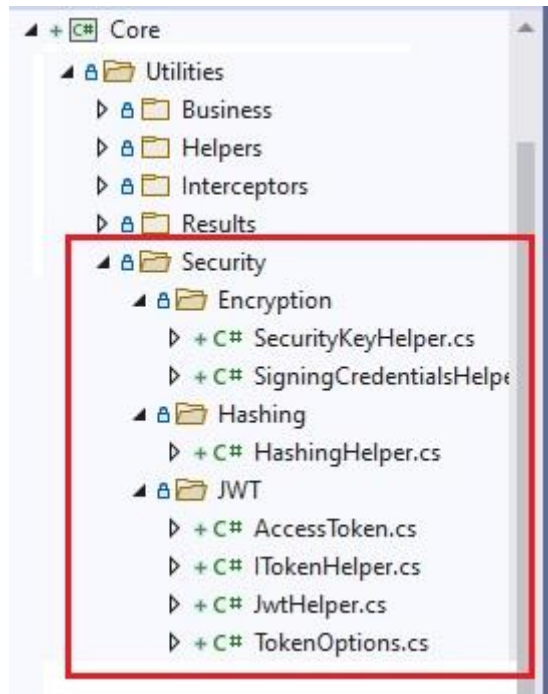
2- Users, OperationClaims, UserOperationClaims Tablolarının karşılığı olan Core\Entities\Concrete klasörünün altında Entitylerini oluşturdum.

3- Users için DataAccess\Abstract altında IUserDal interfaceini ve DataAccess\Concrete\EntityFramework altında EfUserDal classını oluşturdum.

4-Users tablosu için Business\Abstract klasöründe IUserService interfaceini Business\Concrete klasöründe UserManager 'ı oluşturdum.

5-Entities\DTOs klasörünün içine UserForLoginDto , UserForRegisterDto classlarını oluşturdum.

6-Core\Security klasörünü ve içindekileri oluşturdum :



7- Core Katmanına Manage Nuget Package diyerek şu eklentileri yükledim.

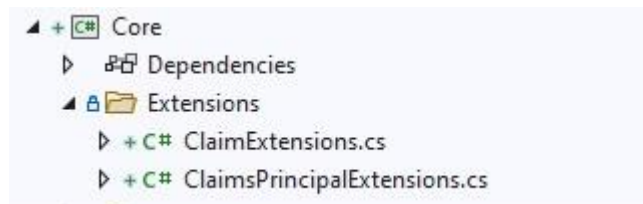
Microsoft.Extensions.Configuration

Microsoft.IdentityModel.Tokens

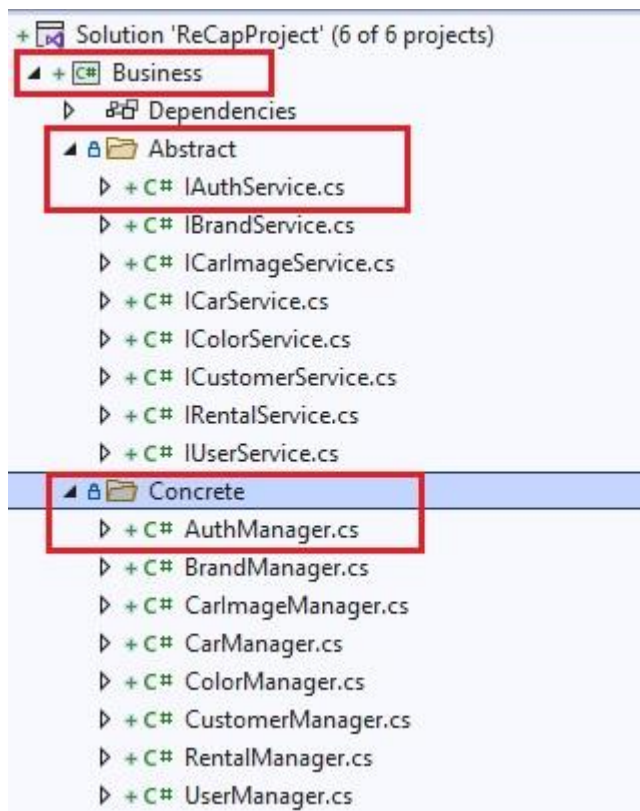
System.IdentityModel.Tokens.Jwt

	<b>Microsoft.Extensions.Configuration</b> by Microsoft Implementation of key-value pair based configuration for Microsoft.Extensions.Configuration. Includes the memory configura...	7.0.0
	<b>Microsoft.IdentityModel.Tokens</b> by Microsoft Includes types that provide support for SecurityTokens, Cryptographic operations: Signing, Verifying Signatures, Encryption.	6.25.0
	<b>System.IdentityModel.Tokens.Jwt</b> by Microsoft Includes types that provide support for creating, serializing and validating JSON Web Tokens.	6.25.0

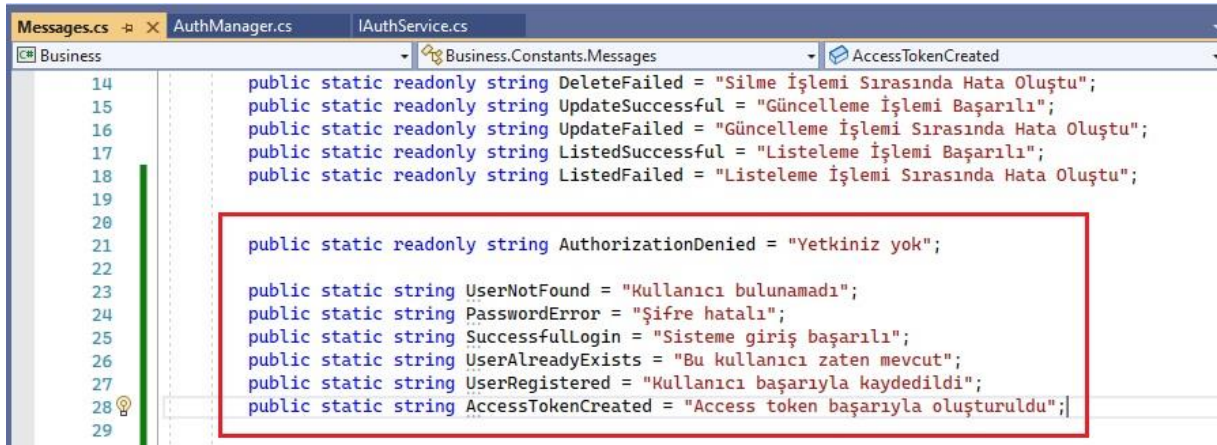
8- Core\Extensions klasörünü ve içindekileri oluşturdum :



9- Business\Abstract\IAuthService ve Business\Concrete\AuthManager'ı oluşturdum.

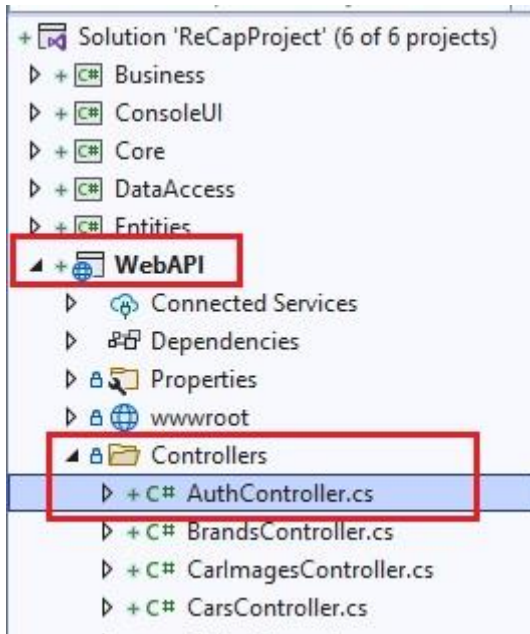


10- Business\ Constants\Messages.cs dosyasına modül için gerekli mesajları ekliyoruz.



```
14 public static readonly string DeleteFailed = "Silme İşlemi Sırasında Hata Oluşturdu";
15 public static readonly string UpdateSuccessful = "Güncelleme İşlemi Başarılı";
16 public static readonly string UpdateFailed = "Güncelleme İşlemi Sırasında Hata Oluşturdu";
17 public static readonly string ListedSuccessful = "Listeleme İşlemi Başarılı";
18 public static readonly string ListedFailed = "Listeleme İşlemi Sırasında Hata Oluşturdu";
19
20
21 public static readonly string AuthorizationDenied = "Yetkiniz yok";
22
23 public static string UserNotFound = "Kullanıcı bulunamadı";
24 public static string PasswordError = "Şifre hatalı";
25 public static string SuccessfulLogin = "Sisteme giriş başarılı";
26 public static string UserAlreadyExists = "Bu kullanıcı zaten mevcut";
27 public static string UserRegistered = "Kullanıcı başarıyla kaydedildi";
28 public static string AccessTokenCreated = "Access token başarıyla oluşturuldu";
29
```

11- WebAPI\Controllers klasörünün içine AuthController oluştur.

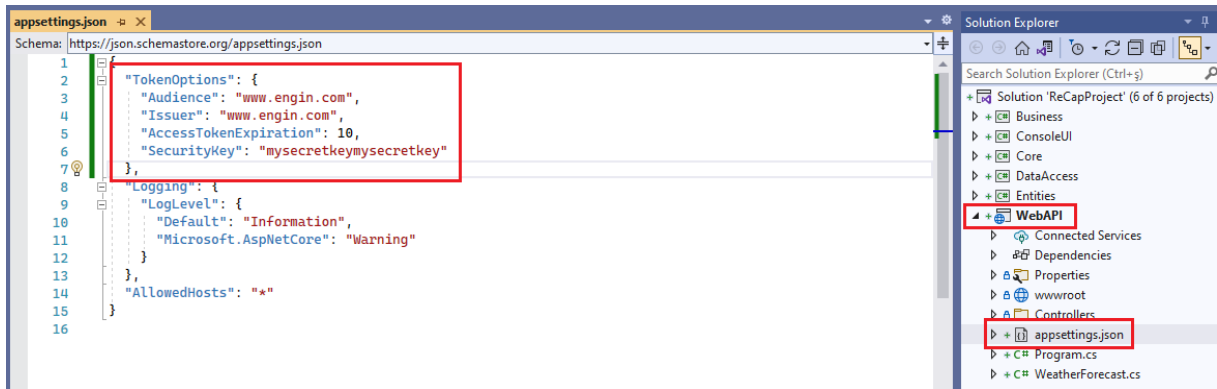


12- Business\DependencyResolvers\Autofac\AutofacBusinessModule.cs dosyasındaki konfigürasyonları yapıyoruz.



```
44
45
46 builder.RegisterType<FileHelperManager>().As<IFileHelper>().SingleInstance();
47
48 // Bu kısım Authorization için lazım eklenmeli
49 // 14.gün dersinde ekledim
50 builder.RegisterType<UserManager>().As<IUserService>().SingleInstance();
51 builder.RegisterType<EfUserDal>().As<IUserDal>().SingleInstance();
52
53 builder.RegisterType<AuthManager>().As<IAuthService>().SingleInstance();
54 builder.RegisterType<JwtHelper>().As<ITokenHelper>().SingleInstance();
55
56 builder.RegisterType<HttpContextAccessor>().As<IHttpContextAccessor>().SingleInstance();
57 // buraya kadar 14.gün dersinde ekledim
58
59
60
```

13- WebAPI içerisindeki TokenOptions ayarlarını yapıyoruz.

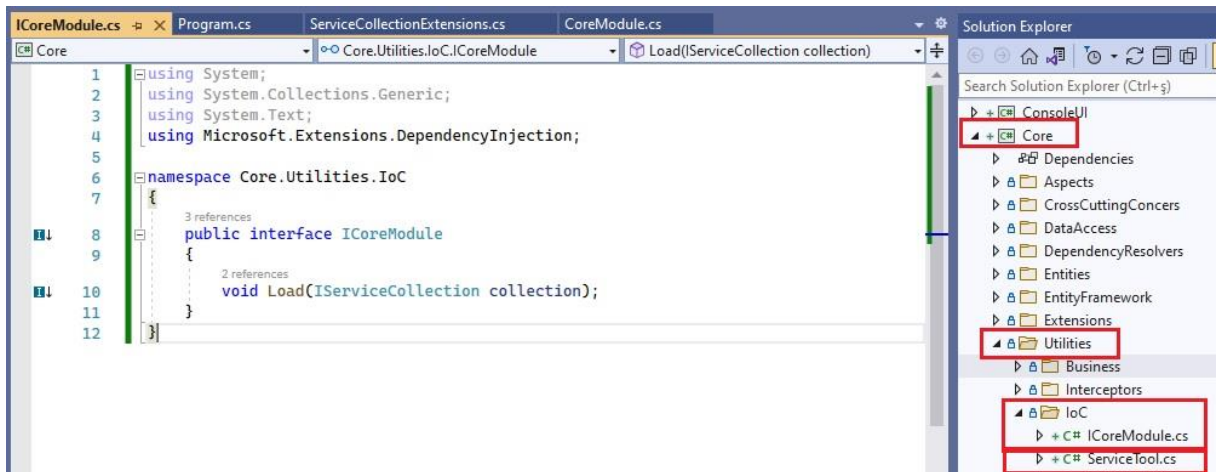


14- WebAPI deki Program.cs dosyasının için bu satırı ekliyoruz.

`app.UseAuthentication`



15- Core\Utilities\IoC klasörünün içine ICoreModule.cs ve ServiceTool.cs oluştur.





16-Core\DependencyResolvers klasörü ve onun içine de CoreModule.cs classını oluştur.

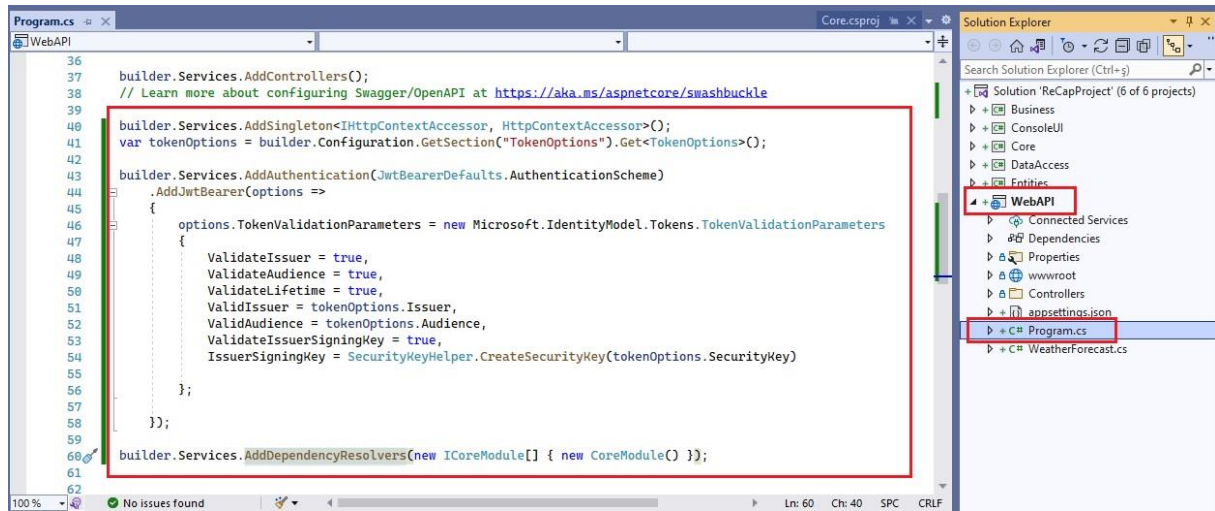


17- Core\Extensions klasörünün içinde ServiceCollectionExtensions.cs classını oluştur.



18- WebAPI Katmanına Manage Nuget Package den  
Microsoft.AspNetCore.Authentication.JwtBearer 'ı kur.

19-WebAPI\Program.cs dosyasına belirtilen kısmı ekle .



Kodlarımız gayet güzel şekilde çalışması lazım. Artık testimizi yapabiliriz.