

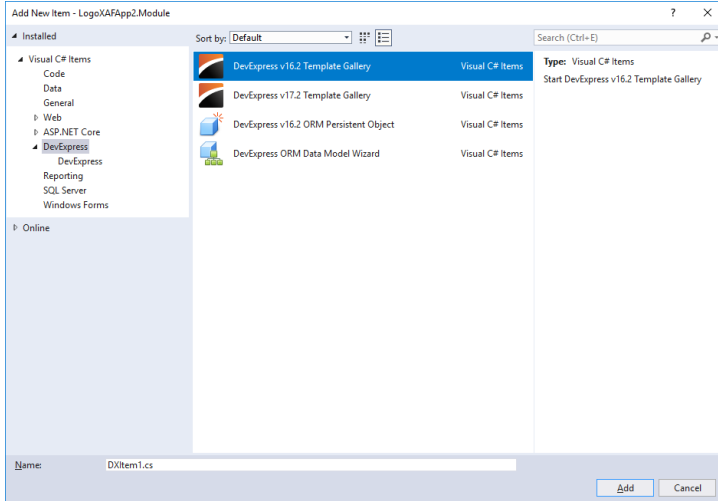
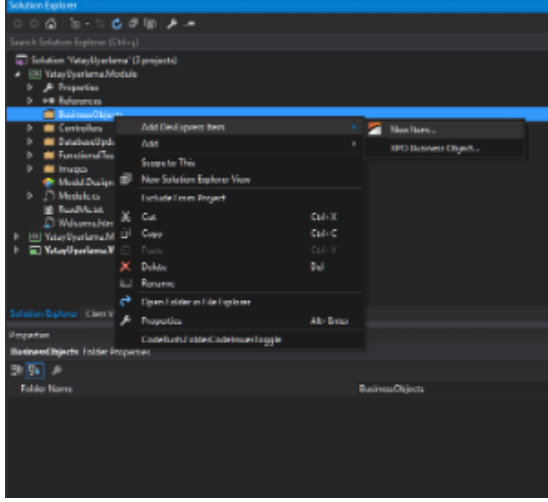
Yatay Uyarlama Geliştirme

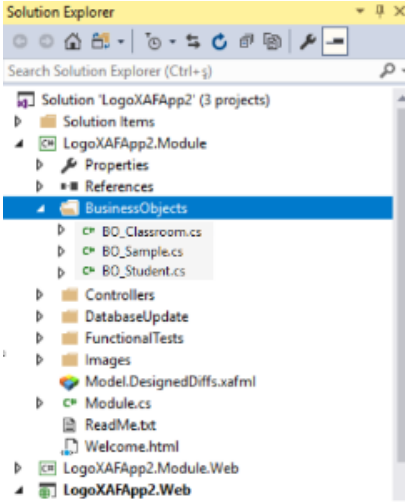
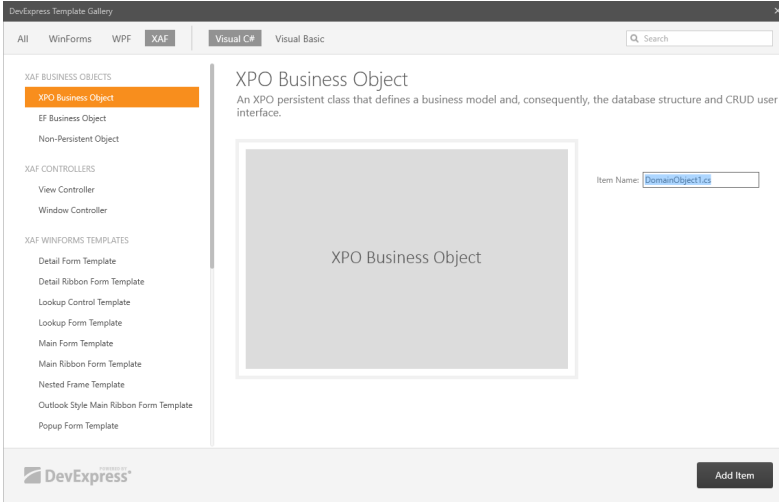
Proje oluşturulur : [Proje oluşturma](#)

Örnek Business Object Oluşturma

Oluşturulan iş nesnelerinin DevExpress.Persistent.BaseImpl.BaseObject ten türetilmesi gerekir.

İş nesnesi aşağıdaki şekilde **BusinessObject** dizini altında oluşturulmalıdır.





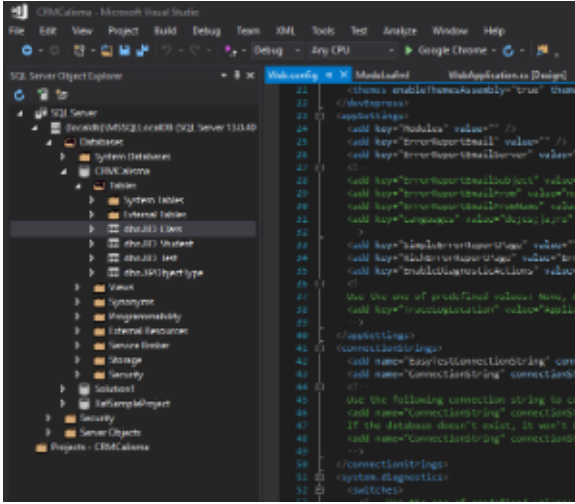
Üretilen uyarlama iş nesnelerinin **BaseObject**'ten türemiş olması gerekir.

Örn :

```
public class BO_Classroom : BaseObject  
  
public class BO_Student : BaseObject
```

İş Nesnelerinin Tablo Görünümleri

VT görünümünü aşağıdaki görüntüden inceleyebilirsiniz.

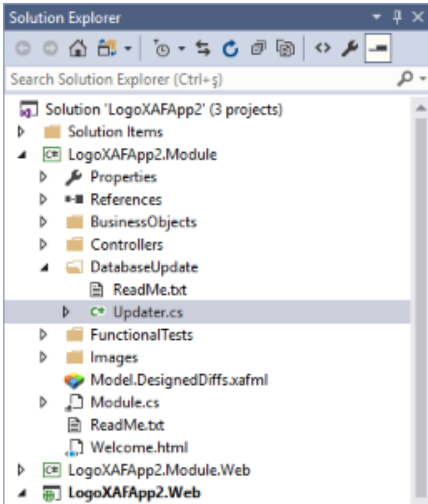


Web Uygulaması Olarak Çalıştırma

Uygulamayı Logo CRM e eklemeyen önce çalıştırıp CRUD işlemlerini test edebiliriz. Artık Logo CRM'e eklenmeye hazır bir uyarlamamız var.

sadece **Proje.Module.dll** dosyasını logoCRM/bin dizini altına taşıyacağız.

DB Updater Güncellemesi



Proje oluşturma sırasında seçilen kimlik doğrulama (authentication) mekanizmasına göre burada otomatik kod üretilir.

Kimlik denetimi Proje Logo CRM tarafından yapılacağı için kodun aşağıdaki şekilde olması gerekir.

Updater.cs

```
namespace LogoXAFApp2.Module.DatabaseUpdate {  
    // For more typical usage scenarios, be sure to check out https://documentation.devexpress.com  
    /eXpressAppFramework/clsDevExpressExpressAppUpdatingModuleUpdaterTopic.aspx  
    public class Updater : ModuleUpdater {  
        public Updater(IObjectSpace objectSpace, Version currentDBVersion) :  
            base(objectSpace, currentDBVersion) {  
        }  
        public override void UpdateDatabaseAfterUpdateSchema() {  
            base.UpdateDatabaseAfterUpdateSchema();  
        }  
        public override void UpdateDatabaseBeforeUpdateSchema() {  
            base.UpdateDatabaseBeforeUpdateSchema();  
        }  
    }  
}
```

Otomatik Deploy Tanımı

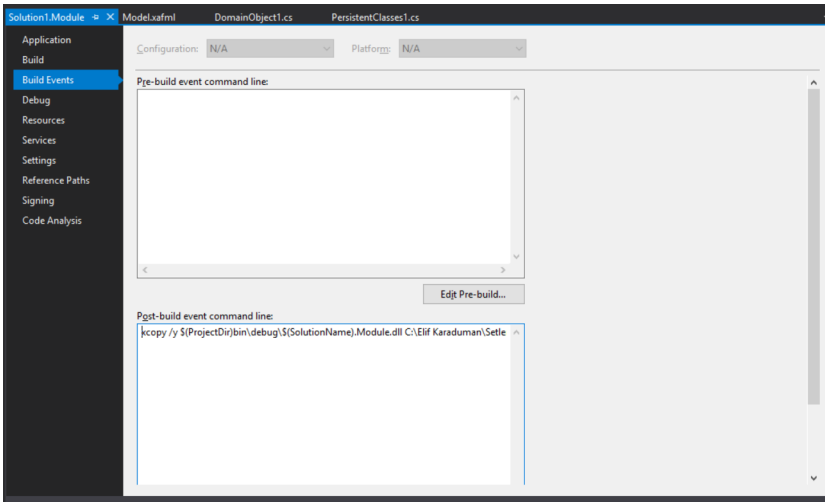
Sadece **proje.Module** çıktısı kullanılacaktır. Üretilen **dll** elle ya da aşağıdaki gibi otomatik olarak hedef dizine kopyalabilir.

Post Built Kod Örneği

```
xcopy /y $(ProjectDir)bin\release\$(SolutionName).Module.dll [Logo CRM Dizini]\bin
```

Örnein

```
xcopy /y $(ProjectDir)bin\release\$(SolutionName).Module.dll C:\LogoCRM\bin  
xcopy /y $(ProjectDir)bin\debug\$(SolutionName).Module.dll C:\LogoCRM\bin  
xcopy /y $(ProjectDir)my_file.ini $(ProjectDir)\$(OutDir)
```



[Diğer makrolar için ...](#)

Logo CRM de Uyarılama Modül Tanımı

dll adını **logo CRM** kök dizini altında **web.config** içinde **appsetting** altında **Module values** olarak kaydedilir.

web.config dosyasına uyarlama projesinin adı **dll olmadan values alanına yazılmalı.**

Örn: **Proje.Module.dll** için **Proje.Module** yazılmalı

```
<add key="Modules" value="Proje.Module" />
```

Örn: **Proje.Module.dll** için **Proje.Module** yazılmalı

```
<add key="Modules" value="Proje.Module" />
```

crm web.config

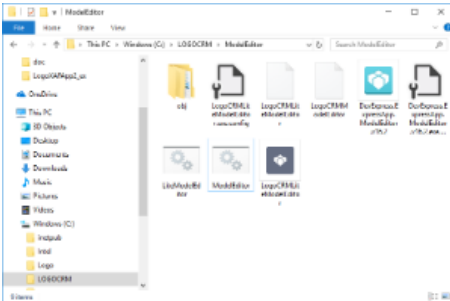
```
<appSettings>
  <add key="Modules" value="...." />
  <add key="ErrorReportEmail" value="" />
  <add key="ErrorReportEmailServer" value="" />
  <add key="SimpleErrorReportPage" value="" />
  <add key="RichErrorReportPage" value="Error.aspx" />
  <add key="EnableDiagnosticActions" value="False" />
  <add key="Languages" value="en-US;tr-TR" />
</appSettings>
```

Proje.Module build alınır **LogoCrm\bin** dizini altına **dll** otomatik ya da elle kopyalanır.

Logo CRM Model Editör

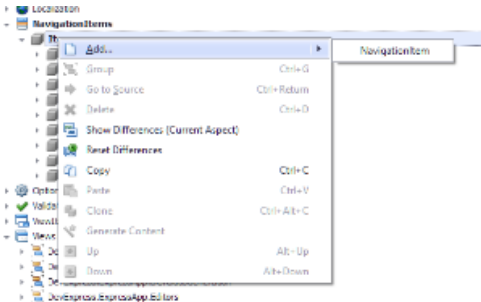
Ürün Model Editöründen Uyarlama Ekranları Navigation bölümünde tanımlanmalı.

ModelEditör klasöründeki ModelEditor.bat dosyası çalıştırılmalı.

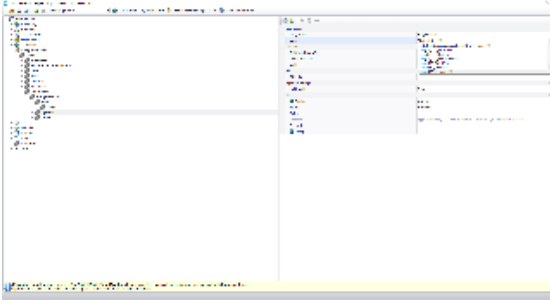


ListView ler Logo CRM navigation'a eklenir.

NavigationItems altında Items üzerinde Add/NavigationItem diyerek yeni bir NavigationItem eklenir.

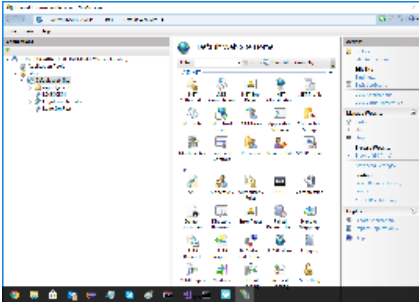


View alanında oluşturduğumuz BusinessObject'ın ListView'i seçilir.



IIS Tekrar Başlatma

Uyarlamanın Logo CRM de görebilmeniz için IIS ten restart yapmanız gerekir.



Logo CRM Login

Aşağıdaki gibi ilk uyarlama projemizi Logo CRM uygulaması içerisinde görebiliriz.

