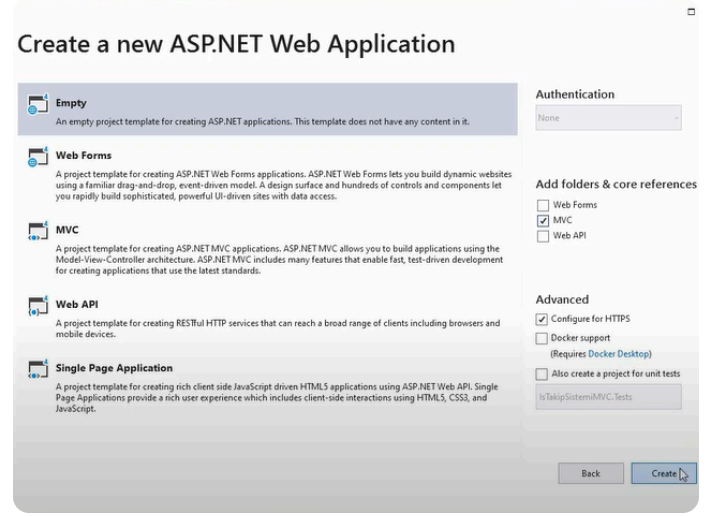
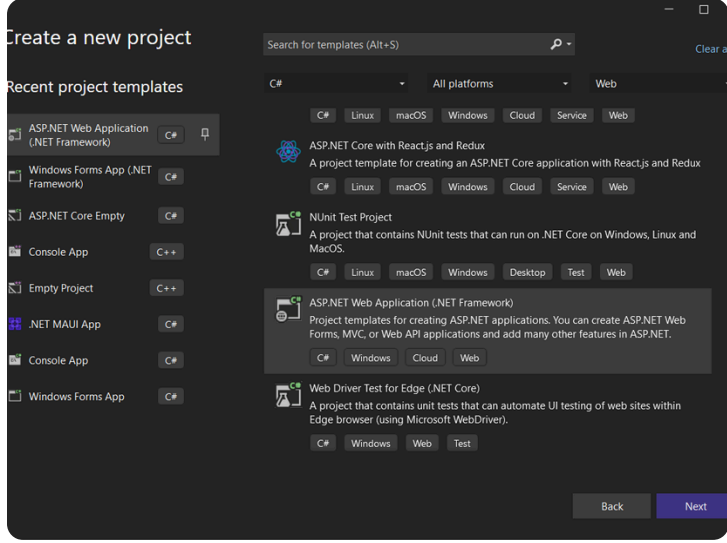


Proje fikrim, yakın bir akrabamızın lokanta firması açmasıyla ortaya çıktı. Bana bir iş takip uygulamasına ihtiyaçları olduğunu söylediler. Bunun üzerine, projeyi tasarlamaya başladım ve sunumda prototipi sunmayı planlıyorum.

## 1.Aşama:



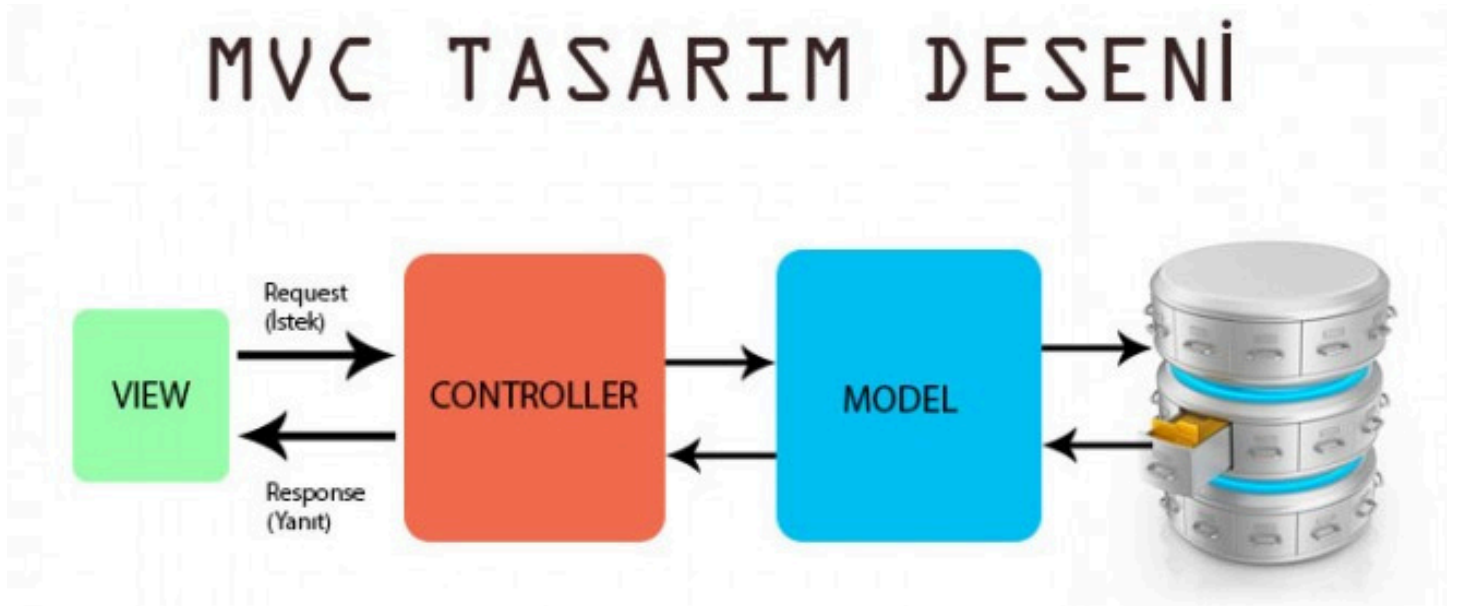
Proje oluştururken **Empty** seçeneğini seçiyorum, ancak MVC paketlerinin eklenmesi için **MVC** seçeneğini işaretliyorum. Projemde:

- **Controllers** kısmında denetleyicilerim yer alacak.
- **Models** kısmında, veritabanından doğrudan veri çeken modeller bulunacak.
- **Views** kısmı, görüntülenecek sayfalarımızı barındıracak.

Ayrıca, Manage NuGet Packages kısmından Bootstrap kütüphanesini indiriyoruz. Projenin:

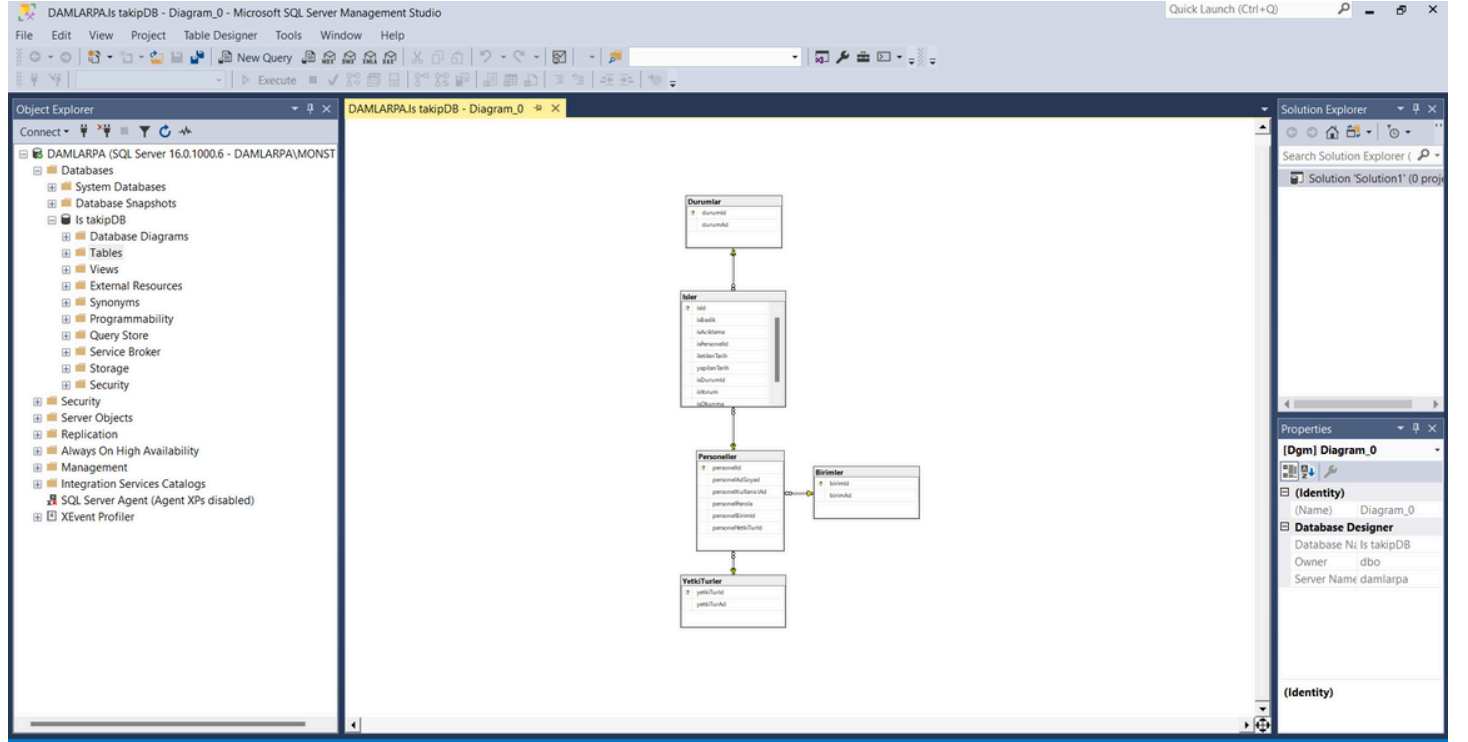
- **Content** kısmında CSS dosyalarımız,
- **Scripts** kısmında ise JavaScript dosyalarımız yer alıyor.

Aşağıdaki görselde MVC'nin çalışma prensibini görebilirsiniz.



## 2.Aşama:

MS SQL projemiz için SQL Server'ı açıyoruz. Veritabanında, bir personelin bir birimde yer alacağı ve bir birimin birden çok personeli olabileceği şekilde tablolar oluşturuyorum. Ayrıca, her birimin yalnızca bir yöneticisi olacak şekilde bir yapı tasarlıyorum.

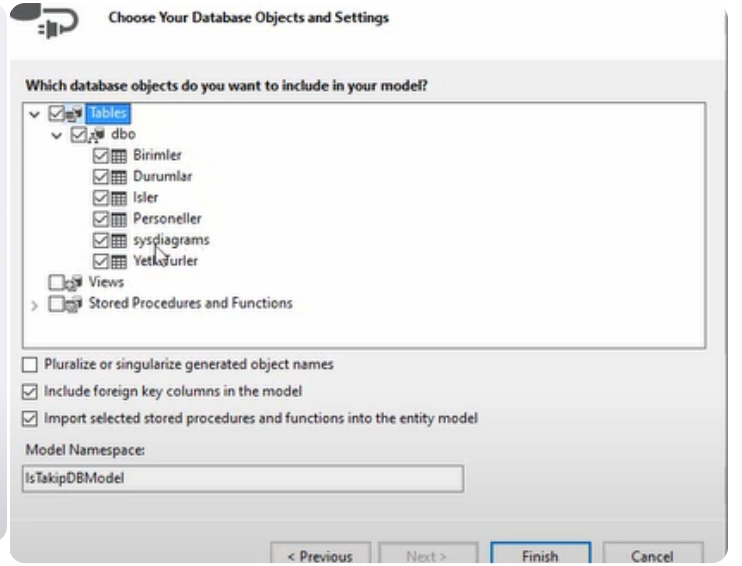
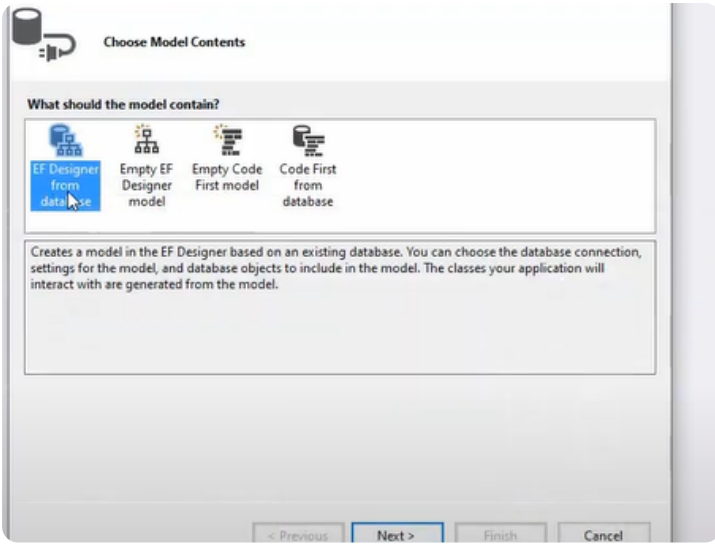


**select \* from Personeller join Birimler on Personeller.personelBirimId=Birimler.birimId**

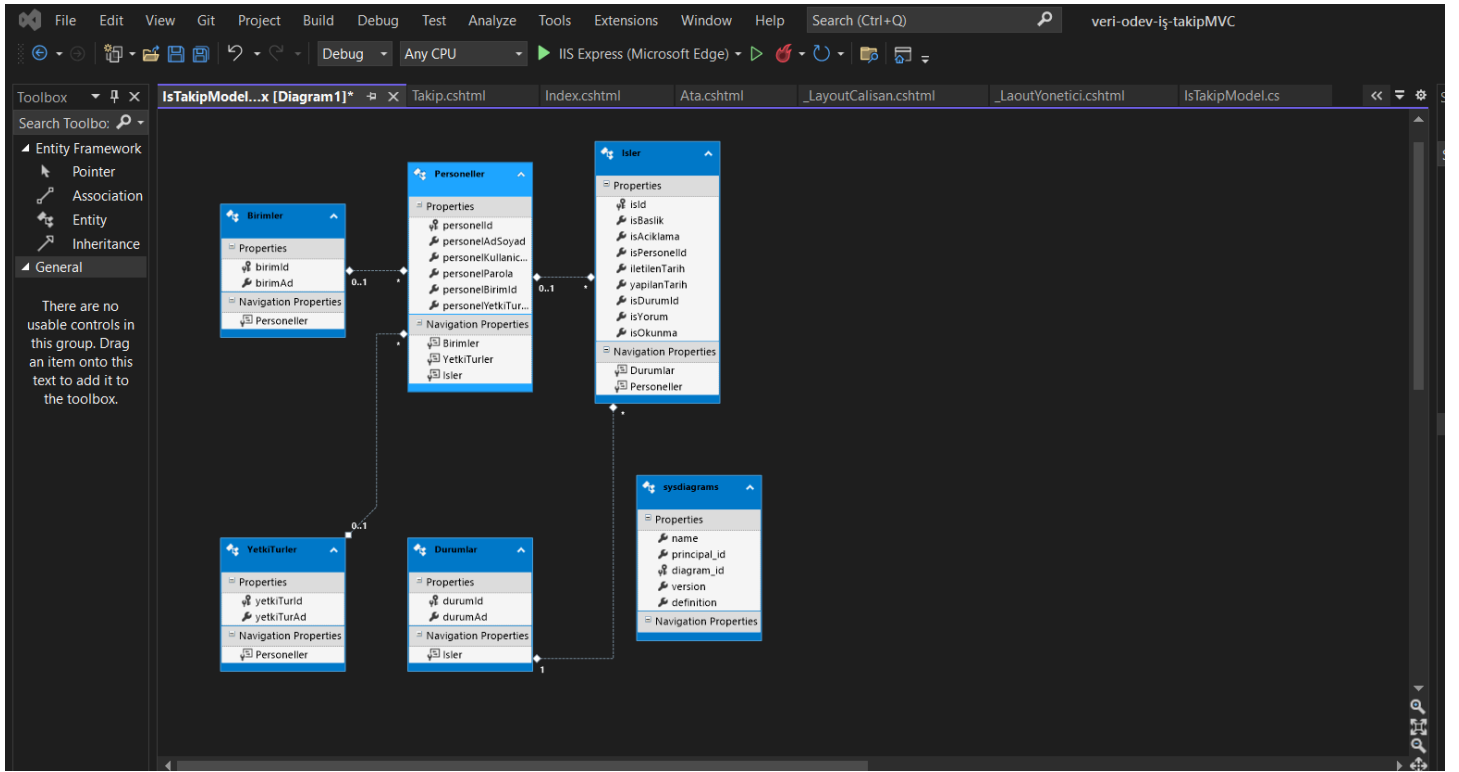
```
select * from Personeller join Birimler on Personeller.personelBirimId=Birimler.birimId
```

	personelId	personelAdSoyad	personelKullaniciAd	personelParola	personelBirimId	personelYetkiTurId	birimId	birimAd
1	1	Luffy	Luffy	Test123	1	1	1	Muhasebe
2	2	Nami	nami	Test123	1	2	1	Muhasebe
3	3	Zoro	zoro	Test123	1	2	1	Muhasebe
4	4	Franky	franky	Test123	2	1	2	Bilgi İşlem
5	5	Usop	usop	Test123	2	2	2	Bilgi İşlem
6	6	Sanji	sanji	Test123	2	2	2	Bilgi İşlem
7	7	damla	arpa	Test123	2	2	2	Bilgi İşlem

Personel listemiz bu şekilde. Şimdi veritabanımızı MVC projemize bağlayacağız ve bu işlem için ADO.NET Entity Data Model kullanacağız.



Veritabanında tasarladığımız yapıyı MVC projemize bağlıyoruz.



### 3.Aşama:

"Controllers klasöründe bir **LoginController** sınıfı oluşturuyoruz. Bu sınıfta bir Entity nesnesi tanımlayarak veritabanı bağlantısını sağlıyoruz. Ardından, **Login.Index** sayfasını oluşturmaya başlıyoruz.

Projemde örnek olması açısından iki birim bulunuyor:

- Muhasebe
- Bilgi İşlem

Bu birimlerde iki farklı yetki türü var:

- Yönetici

- Çalışan

isteme giriş yapıldığında, örneğin muhasebe biriminden bir yönetici sadece muhasebe alanında çalışanların giriş yaptığında şu bilgilendirme mesajı görüntülenir:

### “İş Takip Yönetici”

<Muhasebe birimi için yetkilisiniz.>

Ayrıca üç buton ile karşılaşır:

- AnaSayfa
- İş Ata
- İş Takip

Yönetici, bu seçeneklerden yapmak istediği işlemi seçebilir.

Eğer kullanıcı bir çalışan ise şu bilgilendirme mesajı gösterilir:

### “İş Takip Çalışan”

<Muhasebe birimi için çalışansınız.>

Bu durumda, çalışan aşağıdaki üç butonu görür:

- AnaSayfa
- İş Tamamla
- İş Takip

Çalışan, ana sayfada yöneticisinin kendisine gönderdiği işleri görüntüleyebilir.

İş Takip Çalışan

AnaSayfa

İş Tamamla

İş Takip

Çıkış

Muhasebe birimi için çalışansınız

Yeni İş !!!

üçüncü iş

Okundu İşaretle

Yeni İş !!!

birinci iş

Okundu İşaretle

İş Takip Yönetici

AnaSayfa

İş Ata

İş Takip

Çıkış

Muhasebe birimi için yetkilisiniz.

## ARAYÜZ TASARIMI:

Eğer kullanıcı adımız veya parolamız hatalıysa, sistemimiz bu durumu bildiren bir bilgi mesajı gönderiyor.

İş Takip Sistemi Giriş Sayfası

Kullanıcı Adınız:  
senal

Parola  
\*\*\*\*\*

Giriş

Kullanıcı adı ya da parola yanlış !