# STAJ Projesi

*Mustafa Tütüncü, 30 Mayıs 2025*

**Proje Adı :** ALTAY DVD RENTAL (DvdRental)

**Proje Dizini :** $HOME\Projects\Angular-Dotnet\DvdRental

**Proje Doküman Dizini :** \\sw-arsiv\Kullanıcılar\Mustafa Tütüncü\Staj\DvdRental

1. ***Kurulacak/Kullanılacak Ürünler/Araçlar:***

* NodeJS (LTS): v22.x
* NPM: 11.x
* VS Code > = 1.99 .3 veya Visual Studio 22 >= v17.9.7
* Powershell: 7.5.x
* Terminal: 1.22.x
* Angular: 19.x
* Typescript: 5.x
* MDB 5 Free (Angular için> Usage data:N, Stylesheet: SCSS, SSR/SSG: N, All modules, Roboto Font:Y, Browser animations:Y, Font Awesome:Y, Charts:Y) >= v8.x
* EF Core Power Tools for Visual Studio 2022: v2.6.x
* Microsoft SQL Server Management Studio >= v20.x
* .NET Core SDK 9.0 (Standard Term Support)
* Entity Framework Core 9.x (SQL Server için)
* SQL Server LocalDB (v15.x – MS SQL Server 2019)

1. ***Proje Yaratma Adımları:***

**Kullanılacak Araç :** Visual Studio 2022

**Yaratılacak Proje Tipi :** Angular and ASP.NET Core

**Solution Name :** DvdRental

**Location :** $HOME\Projects\Angular-Dotnet

Create in new folder (Checked)

Framework: .NET 9.0 (Standart Term Support)

Configure for HTTPS (Checked)

Enable container support (Not checked)

Enable OpenAPI support (Checked)

Do not use top-level statements (Checked).

**Not**: MinimalAPI için aşağıdakiler seçilmeli. Controller based API için ‘Checked’ olmalı.

Use controllers (Not Checked)

**Not**: Controller Based API için aşağıdakiler seçilmeli.

Use controllers (Checked)

Enlist in .NET Aspire orchestrat5ion (Not Checked)

**Created Project Names :** dvdrental.client (Angular 19), DvdRental.Server (.NET 9 Web API)

**Not**: Proje ayrıca VS Code yada Powershell 7+Notepad ile de yaratılabilir. Bu durumda $HOME\Projects\Angular-Dotnet\DvdRental\src dizini altında api ve client dizinleri oluşturulup, bunların altında ilgili proje araçları kullanılabilir (.NET için dotnet, Angular için ng).

1. ***Yapılacaklar:***

## Veri Tabanı

* SQL Server LocalDB'ye **dvdrental** veri tabanı yüklenecek (Gerekli sql script ‘**dvdrental-sqlserver.sql**’ sağlanacak).
* Veri Tabanı’nda sadece 'actor, film, category, film\_actor, film\_category’ tabloları kullanılacak.
* Yüklenen **dvdrental** veri tabanı'nın ER diagramı oluşturulacak. Bunun için araç olarak ‘**Microsoft SQL Server Management Studio**’ yada ‘**EF Core Power Tools**’ kullanılabilir.

## Uygulama Genel

* Fullstack **DvdRental** çözümü oluşturulacak (Ön uç: Angular 19, Arka uç: .NET 9)

## Arka uç (Backend)

* API geliştirme yaklaşımı olarak ‘MinimalAPI’ yada ‘Controller based API’ yaklaşımı seçilebilir. Proje yaratılırken buna dikkat edilmelidir. **Bkz**: Proje Yaratma Adımları.
* Mevcut veri tabanı (**dvdrental)** kullanılarak modeller (entities/classes) ve **DBContext** (**DvdRentalDBContext**) oluşturulacak (Database first yaklaşımıyla). Bunun için ‘**EF Core Power Tools’** aracı yada VS2022’deki ‘**Nuget Package Manager Console**’ üzerinden ‘**scaffold-dbcontext**’ komutu kullanılabilir.
* Temiz kod yapısı (Clean Architecture) gereği yaratılacak tüm objeler gruplandırılıp uygun dizinler altında oluşturulmalıdır. Örnel: Modeller/Nesneler ‘**Models’**, DBContext ‘**Data’,** servisler ‘**Services’**, minimalAPI fonksiyonları ‘**Endpoints’** yada Controller aksiyonları (Actions) ‘**Controllers’** dizininde oluşturulabilir.
* EF Core ile ‘**dvdrental** veritabanı erişimi (**DvdRentalDBContext**) projeye DI olarak tanımlanacak.
* Arka uç tarafına 'publishers, titles, authors, titleauthor’ için CRUD API’ler “**Controller**” yada “**Minimal API**” tabanlı ve async/await olarak yazılacak.
* Veri Tabanı erişimi doğrudan Controller action yada Minimal API endpoint’lerden yapılmayacak. Servis ara katmanı oluşturulacak.
* Örnek olarak ‘**publishers’** için ‘**Controller’** tabanlı API’ler için:
  + Yeni Controller (PublishersController) yaratılacak ve bu Program.cs’de ‘builder.Build()’a ‘Map’ edilecek.
  + Veriler ‘Controller’ katmanında doğrudan ‘action’lar ya da ayrı bir servis katmanı kullanılarak çekilecek.
  + PublishersController içine CRUD'lar eklenecek ve servisler ile eşlenecek (Doğrudan ‘action’lar kullanılmazsa).
  + Controller’da doğrudan ‘action’lar yada ‘Servis’ katmanında veri tabanı iletişimi, DI olarak gelen Context aracılığı ve EF yardımıyla async/await olarak yazılacak.
  + Hata durumları anlamlı olarak istemcilere (Burada ön uç) iletilecek.
* Örnek olarak ‘**authors’** için ‘**Minimal API**” tabanlı API’ler için:
  + Yeni Class (AuthorEndpoints) yaratılacak.
  + AuthorEndpoints class için ‘static void MapAuthorEndpoints’ function yazılacak ve bu function Program.cs’de ‘builder.Build()’a ‘Map’ edilecek.
  + Hata durumları anlamlı olarak istemcilere (Burada ön uç) iletilecek.
* CRUD API’lerin başlangıç temel kodları için ‘EF Core Power Tools’ aracından ve ‘VS 2022 Scaffold’ özelliğinden yararlanılabilir (Zorunlu değil).

## Ön uç (Frontend)

* Angular tarafına MDB 5 Free (>=v8.x) desteği eklenecek.
* Uygulamanın **'publishers, titles, authors, titleauthor** ' tablolarını CRUD olarak kullanan menü tasarımı MDB ile yapılacak .
* Veriler arka uç’taki api’lerden async olarak çekilecek.
* Hata durumları anlamlı olarak kullanıcıya yansıtılacak.

## Erişim Güvenliği

* Güvenlik için JWT tabanlı token ve HmacSha256 algoritması kullanılacak.
* JWT ile rol bazlı login ekranı angular ve dotnet WebAPI tarafına eklenecek. Kullanicilar (Users) tablo olarak veritabanında tutulacak. Kullanıcı rolleri de aynı tabloda tutulabilir. Burada 2 rol tanımlanacak. **Admin** ve **User**. Şimdilik her kullanıcının 1 rolü olacak. **Admin** hakkı ile giriş yapan kullanıcılar Authors, Titles, Publishers sayfalarında listeleme yanında ekleme, düzeltme ve silme de yapabilecek. **User** hakkı ile giriş yapanlar ise ilgili sayfasında sadece listeleme yapabilecek.
* Kullanıcı şifreleri PBKDF2 algoritmasıyla hash olarak veritabanında Kullanicilar (Users) tablosunda üretilen salt ile birlikte tutulacak.
* Manual olarak admin adlı kullanıcı ve kullanılan algoritma ile (Adm2025!) şifresinin hash ve salt değerleri hesaplanıp, pubs DB deki Users tablosuna eklenecek. Bunun için gerekli masaüstü (Desktop) uygulaması size sağlanacak > [\\10.0.0.4\Kullanıcılar\Mustafa Tütüncü\Staj\PasswordHash(PBKDF2 )\_Demo.zip](file:///\\10.0.0.4\Kullanıcılar\Mustafa%20Tütüncü\Staj\PasswordHash(PBKDF2%20)_Demo.zip) .
* Admin rolü için Kullanıcı Yönetimi (User Management) menüsü tasarlanacak. Bunun için bir önceki adımda tarif edilen pubs db Users tablosunda kullanıcı: ‘admin’ ve şifre: ‘Adm2025!’ yaratıldığı varsayılacak.