

ÇANAKKALE ONSEKİZ MART ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

MEOWIE

Alper Kuzucuoğlu

Soner Gezgin

Doç. Dr. Üyesi Engin Şahin

Haziran, 2024

ÇANAKKALE

MEOWIE

Çanakkale Onsekiz Mart Üniversitesi Mühendislik Fakültesi
Bitirme Ödevi
Bilgisayar Mühendisliği Bölümü

Alper Kuzucuoğlu

Soner Gezgin

Doç. Dr. Üyesi Engin Şahin

Haziran, 2024

ÇANAKKALE

ALPER KUZUCUOĞLU ve **SONER GEZGİN** tarafından **DOÇ. DR. ÜYESİ**
ENGİN ŞAHİN yönetiminde hazırlanan “**MEOWIE**” başlıklı çalışma tarafımızdan
okunmuş, kapsamı ve niteliği açısından bir Bitirme Ödevi olarak kabul edilmiştir.

Danışman

Doç. Dr. Üyesi Engin Şahin

Bölüm Başkanı

Prof. Dr. SAFİYE AYŞE GÖKER

Bilgisayar Mühendisliği Bölümü

TEŞEKKÜR

Bu projede emeği geçen herkese çok teşekkür ederiz.

Proje danışmanımız, Sayın Doç. Dr. Üyesi Engin Şahin, projede bize verdiği bilgi, beceri ve destek için teşekkür ederiz.

**ALPER KUZUCUOĞLU
SONER GEZGİN**

MEOWIE Film Değerlendirme Sitesi

ÖZET

Projemiz, film ve dizi tutkunları için özel olarak tasarlanmış ve kullanıcıların film ve dizi tercihlerini keşfetmelerine, yorumlar yapmalarına ve diğer kullanıcılarla etkileşim kurmalarına olanak tanıyan sosyal medya benzeri bir platformdur. Platform, kullanıcıların yeni filmler ve diziler keşfedebilecekleri, kendi görüş ve derecelendirmelerini paylaşabilecekleri film ve dizi meraklıları topluluğu oluşturmak için tasarlanmıştır.

Kullanıcılar hesap oluşturarak profillerini oluşturabilir ve kişiselleştirebilirler. Kullanıcılar profilleri aracılığıyla diğer kullanıcılarla etkileşim kurabilecekler ve onların film/dizi tercihlerini keşfedebilecekler. Kullanıcılar giriş yaptıktan sonra birbirlerinin profillerini arayarak sosyalleşebilir ve değerlendirme yapmak istedikleri film veya diziyi arayarak derecelendirmeler yapabilirler.

Kullanıcılar, diğer kullanıcıların derecelendirmelerini görüntüleyebilir, yaptığı yorumları okuyabilir, yorumları beğenebilir ve diğer kullanıcılarla takip isteğinde bulunabilirler. Bu sayede, film ve dizi tutkunları arasında bir topluluk oluşturulmuş olur ve kullanıcılar birbirleriyle etkileşim içinde olabilirler. Ayrıca kullanıcılar, izlemeyi planladıkları, izledikleri veya favori filmler için kendi film/dizi listelerini oluşturabilir ve takipçilerinin erişimine açık veya gizli bir şekilde bu listeleri paylaşabilirler.

Ana sayfada, kullanıcıların takip ettiği kullanıcıların listelerindeki filmler, diziler ve kendi ilgi alanlarına göre algoritma tarafından önerilen filmler ve diziler olacaktır.

Bu proje için kullandığımız teknolojiler React JS ve ASP.NET WEB API kullanılarak oluşturulmuştur. Veritabanı için docker üzerinde PostgreSQL kullanılmıştır. Backend mimarisi olarak Onion Architecture seçilmiş ve kullanılmıştır. ORM olarak ise Entity Framework kullanılmıştır.

İÇİNDEKİLER

	Sayfa
BİTİRME ÖDEVİ ONAY SAYFASI	ii
TEŞEKKÜR	iii
ÖZET	iv
BÖLÜM 1- GİRİŞ	1
1. Projenin Konusu	1
1.1. Projenin Amacı.....	1
1.2. Projenin Önemi.....	1
1.3. Projede Kullanılan Teknolojiler	1
1.4. Proje Yasal Kısıtlamaları.....	2
1.5. Projenin Ekip Üyeleri Ve Görevleri	2
1.6. Projenin Sonunda Elde Edilecek Faydalar	2
2. Projenin Oluşturulması.....	3
2.1. Backend Mimarı İnşaatı	3
2.2. Frontend İnşaatı.....	3
2.3. Veritabanı Oluşturulması.....	3
BÖLÜM 2- ANA BÖLÜM	4
1. Veritabanı Geliştirimi	4
1.1. Veritabanı İçin Veri Toplanması.....	4
1.2. Veritabanı İçin Modellerin Oluşturulması.....	4
1.3. Veritabanına Veri Girişi	4
2. Backend Geliştirimi.....	5
2.1. CQRS Pattern	5
2.2. Controllerların Oluşturulması.....	5
2.3. Ek Servislerin Oluşturulması.....	5
2.4. Kullanıcı İşlemleri	5
2.5. Token İşlemleri.....	5
2.6. Log İşlemleri	5

3. Frontend Geliştirimi	6
3.1.Gerekli Sayfaların Oluşturulması.....	6
3.2.Routerların Oluşturulması	6
3.3.Componentlerin Oluşturulması	6
3.4.Backend Endpointleri İçin Servis	6
3.5.Context API Kullanımı.....	6

BÖLÜM 3- TASARIM BÖLÜMÜ.....7

1. Varlık İlişkisi Şeması	7
2. İlişkisi Şeması.....	8
3. PostgreSQL	9
Şekil 3.1 PostgreSQL Veritabanı Tabloları.....	9
Şekil 3.2 PostgreSQL ER Diagram	10

BÖLÜM 4- Web Sitesi Ekran Görüntüleri.....11

1. Şekil 4 Ana Ekran Arayüzü.....	11
2. Şekil 5.1 Üye Ol Sayfası Arayüzü.....	12
3. Şekil 5.2 Üye ol Sayfası Arayüzü (Doldurulmuş).....	13
4. Şekil 6 Giriş yap Sayfası Arayüzü.....	14
5. Şekil 7 Ana Ekran Arayüzü (Oturum Açık).....	15
6. Şekil 8.1 Film Arama Arayüzü.....	16
7. Şekil 8.2 Film Arama Arayüzü (Arama Kutusu Doldurulmuş)	17
8. Şekil 9.1 Film Açıklama Sayfası Arayüzü	18
9. Şekil 9.2 Film Oyuncuları Sayfası Arayüzü.....	19
10. Şekil 10.1 Filmi Değerlendirme Sayfası Arayüzü.....	20
11. Şekil 10.2 Değerlendirme Güncelleme Sayfası Arayüzü	21
12. Şekil 11 Film Listeleri Arayüzü	22
13. Şekil 12 Film Listesi Görüntüle Arayüzü.....	23
14. Şekil 13 Kullanıcı Profil Sayfası Arayüzü	24
15. Şekil 14 Listeye Film Ekle Sayfası Arayüzü.....	25
16. Şekil 15 Film Listeleri Arayüzü	26

SONUÇ VE TARTIŞMA.....	27
KAYNAKLAR	28
Şekiller.....	29

BÖLÜM 1

GİRİŞ

1. Projenin Konusu

Projemiz, film ve dizi tutkunları için özel olarak tasarlanmış ve kullanıcıların film ve dizi tercihlerini keşfetmelerine, yorumlar yapmalarına ve diğer kullanıcılarla etkileşim kurmalarına olanak tanıyan sosyal medya benzeri bir web sitesidir.

1.1. Projenin Amacı

Bu sosyal medya benzeri film ve dizi derecelendirme platformu, kullanıcıların ilgi alanlarına göre yeni filmler ve diziler keşfedebilecekleri, izledikleri filmler ve diziler hakkında kendi görüşlerini ve derecelendirmelerini paylaşabilecekleri film ve dizi meraklıları topluluğu oluşturmak için tasarlanmıştır.

1.2. Projenin Önemi

Projemizin önemi, film ve dizi tutkunları için özel olarak tasarlanmış bir platform sunmasıdır. Projemiz, kullanıcıların film ve dizi keşfetmelerine, yorumlar yapmalarına ve diğer kullanıcılarla etkileşim kurmalarına olanak tanır. Bu sayede, kullanıcıların ilgi alanlarına uygun filmleri ve dizileri keşfedebilecekler ve daha fazla film ve dizi hakkında bilgi sahibi olabilecekler. Ayrıca, kullanıcıların film ve dizi değerlendirmelerinin gerçek kullanıcılar tarafından yapılması, daha gerçekçi ve doğru bir değerlendirme yapılmasını sağlar.

1.3. Projede Kullanılan Teknolojiler

Bu proje için kullandığımız teknolojiler front-end için React, back-end için ASP.NET ve veritabanı için Docker üzerinde PostgreSQL'dir. Kullanıcı deneyimini optimize etmek için React, platformun işlevsellliğini geliştirmek için ASP.NET, verileri depolamak ve yönetmek için PostgreSQL kullanılacak. Backend mimarisi olarak Onion Architecture seçilip kullanılmıştır. ORM olarak ise Entity Framework kullanılmıştır.

1.4. Projenin Yasal Kısıtlamaları

Bu projede, kullanıcıların kişisel bilgileri ve yaptıkları yorumlar korunması gerektiğinden, veri güvenliği ve gizliliği konusunda özenli çalıştık. Kullanıcılar dilediği takdirde anonim olarak web sitesi üzerinde zaman geçirebilmeliler.

Ayrıca veri koruma yasalarına ve internet yasalarına uymak gereklidir. Bu nedenle, veri koruma yasalarına ve internet yasalarına uymak için gerekli olan önlemleri aldık.

1.5. Projenin Ekip Üyeleri Ve Görevleri

Bu proje Alper Kuzucuoğlu ve Soner Gezgin tarafından yapılmaktadır. Kişilerin bireysel görevleri yoktur, herkes projenin her alanında katkı sağlamıştır.

1.6. Projenin Sonunda Elde Edilecek Faydalar

Bu proje sonunda, film ve dizi tutkunlarının kendi aralarında topluluklar oluşturması ve daha fazla sosyal iletişimleri sağlanacaktır.

2. Projenin Oluşturulması

Projeye oluşturmaya başlarken ilk yaptığımız şey boş bir React JS projesi ve boş bir ASP.NET Web API projesi oluşturmak.

2.1. Backend Mimari İnşaati

Boş bir ASP.NET projesi oluşturuktan sonra projenin içerisinde Onion Architecture'a uygun olarak katmanlar oluşturduk. Daha sonra bu katmanlar arasındaki bağımlılıkları ayarladık. İlk başta gerekli olabilecek tools ve kütüphaneleri indirip kurulumunu yaptık.

2.2. Frontend İnşaati

React JS'de kullanacağımız Material UI, Tailwind, Font Awesome gibi kütüphanelerin kurulumunu gerçekleştirdik. Router altyapısını oluşturduk ve klasörleme işlemlerini gerçekleştirdik.

2.3. Veritabanı Oluşturulması

Docker Desktop kurulumunu gerçekleştirdikten sonra Docker üzerine bir PostgreSQL veritabanı kurulumu yaptık.

BÖLÜM 2

ANA BÖLÜM

1. Veritabanı Geliştirmi

1.1. Veritabanı İçin Veri Toplanması

Internet üzerinde yapay sinir ağı için toplanılan bir veri seti bulduk. İçerisinde TMDB sitesine ait filmlerin id'leri ve IMDB sitesine ait filmlerin id'leri mevcuttu. Python üzerinde yazmış olduğumuz bir script ile TMDB'nin sunduğu API'ı ve bulduğumuz id'leri kullanarak bize uygun olan verileri çekmeye başladık. Topladığımız bu verileri bir JSON dosyasına çevirdik.

1.2. Veritabanı İçin Modellerin Oluşturulması

Veritabanı için tasarladığımız ER Diyagramlarını baz alarak ASP.NET projemizde class modelleri oluşturduk. Migration oluşturarak modellerimize uygun veritabanını code-first yaklaşımı ile oluşturduk. Daha sonra bu modeller için Repository Design Patternlarımıza oluşturduk. Bu repositoryleri kullanarak veritabanını Entity Framework ile çok kolay bir şekilde yönetebildik.

1.3. Veritabanına Veri Girişi

ASP.NET içerisinde JSON için özel modellerimizi oluşturduk. Daha sonra bu JSON dosyasını ve bu modelleri kullanarak deserialize etti. Daha önceden oluşturmuş olduğumuz repositoryleri kullanarak bu verileri kolayca veritabanımıza yerleştirdik.

2. Backend Geliştirimi

2.1. CQRS Pattern

Projemizde CQRS Pattern kullanarak kodlardaki karmaşıklığı ve yenilenebilirliğini arttırdık. Command ve Queryleri ayırarak ileride birden fazla veritabanı oluşturulmak istenilirse kolay bir şekilde geliştirilmesinin önünü açtık.

2.2. Controllerların Oluşturulması

İhtiyacımız olan controllerları oluşturarak içlerinde ihtiyacımız olan endpointleri oluşturmaya başladık. Bu endpointler CQRS pattern anlayışımızdaki her endpointte özel request, response, handler sınıfları oluşturmamız sayesinde çok sade ve geliştirilebilir oldu.

2.3. Ek Servislerin Oluşturulması

Endpointler dışında handlerlar içerisinde kullanmamız gereken servisler için soyut sınıflar ve bunların gerçekleştirmelerini yaptık. Örneğin dosya yükleme servisi, JWT Token.

2.4. Kullanıcı İşlemleri

Kullanıcıların kayıt olması, authorization ve authentication işlemleri gibi işleri yapabilmesi için kullanıcı servisini oluşturduk. Bu servisleri IOC Containerından inject ile elde ederek kodları tekrar tekrar yazmadan bu işlemlerimizi kolayca gerçekleştirebildik.

2.5. Token İşlemleri

Kullanıcıların authorization ve authentication işlemleri için token işlemleri yapmamız gerekiyordu. Token için JWT Token kullanma kararı aldık. JWT tokenlarını kullanarak Access Token ve Refresh Token oluşturabileceğimiz servisleri yazdık ve kullandık.

2.6. Log İşlemleri

Serilog kullanarak backend projesinde oluşan hataları yakaladık ve bunları veritabanı ve txt dosyası içerisinde tuttuk. Böylelikle hataları bu loglardan da inceleme şansımız oldu.

3. Frontend Geliştirimi

3.1. Gerekli Sayfaların Oluşturulması

Öngörülen sayfaların js dosyaları oluşturuldu. Gerekli kurulan kütüphaneler import edildi ve sayfalar fonksiyonel component olarak ayarlandı.

3.2. Routerların Oluşturulması

Sayfalar için ana component içerisinde router oluşturuldu. Hangi componente hangi url'den ulaşılabileceği ayarlandı.

3.3. Componentlerin Oluşturulması

Sayfalar için gerekli componentler oluşturuldu ve componentler içerisinde kullanıldı. Componentler geliştirilirken tailwind, material UI gibi kütüphaneler kullanıldı.

3.4. Backend Endpointleri İçin Servis

Backenddeki endpointlerimize kolayca ulaşmamız için bir servis yazdık. Bu servis sayesinde endpointler değişse bile tek bir yerden tüm projeye müdahale edebildik.

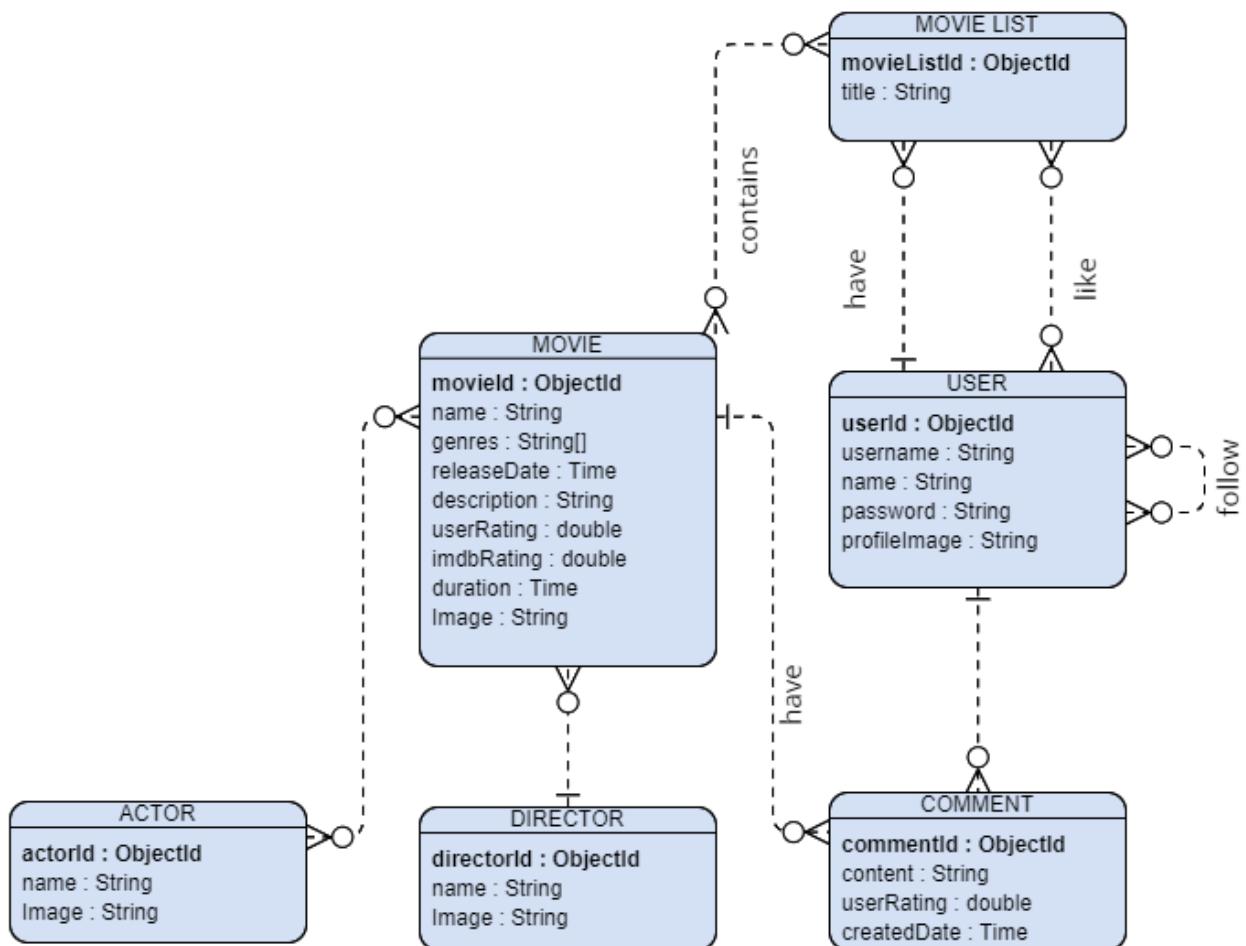
3.5. Context API Kullanımı

Controllerlar için birer Context API oluşturduk. Bu contextler sayesinde farklı componentler içerisinde istediğimiz servisleri kolayca çağrıp kullanabildik. Bu da yönetilebilirliği ve geliştirilebilirliği yükseltti.

BÖLÜM 2

TASARIM BÖLÜMÜ

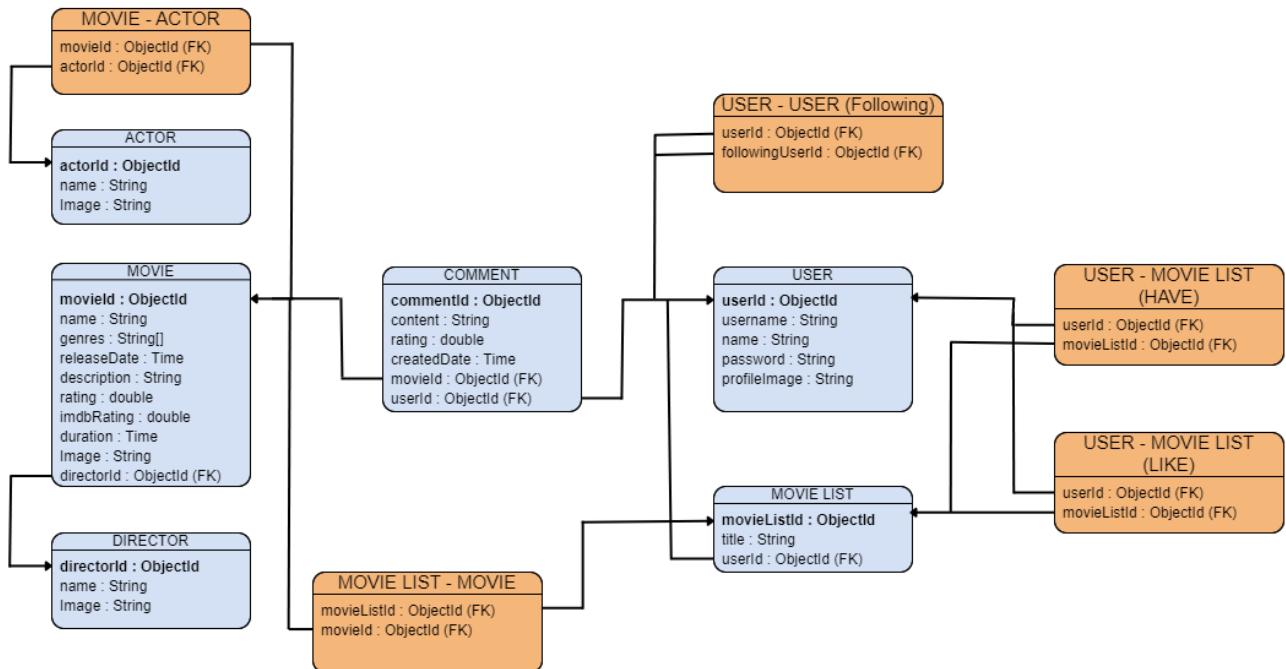
1. Varlık İlişkisi Diyagramı



Şekil 1 Entity-Relationship Diagram

Projemizde kullanacağımız veri tabanını tasarlamak için öncelikle Şekil 1'deki varlık-ilişki diyagramımızı oluşturduk.

2. İlişki Şeması



Şekil 2 Relation Schema

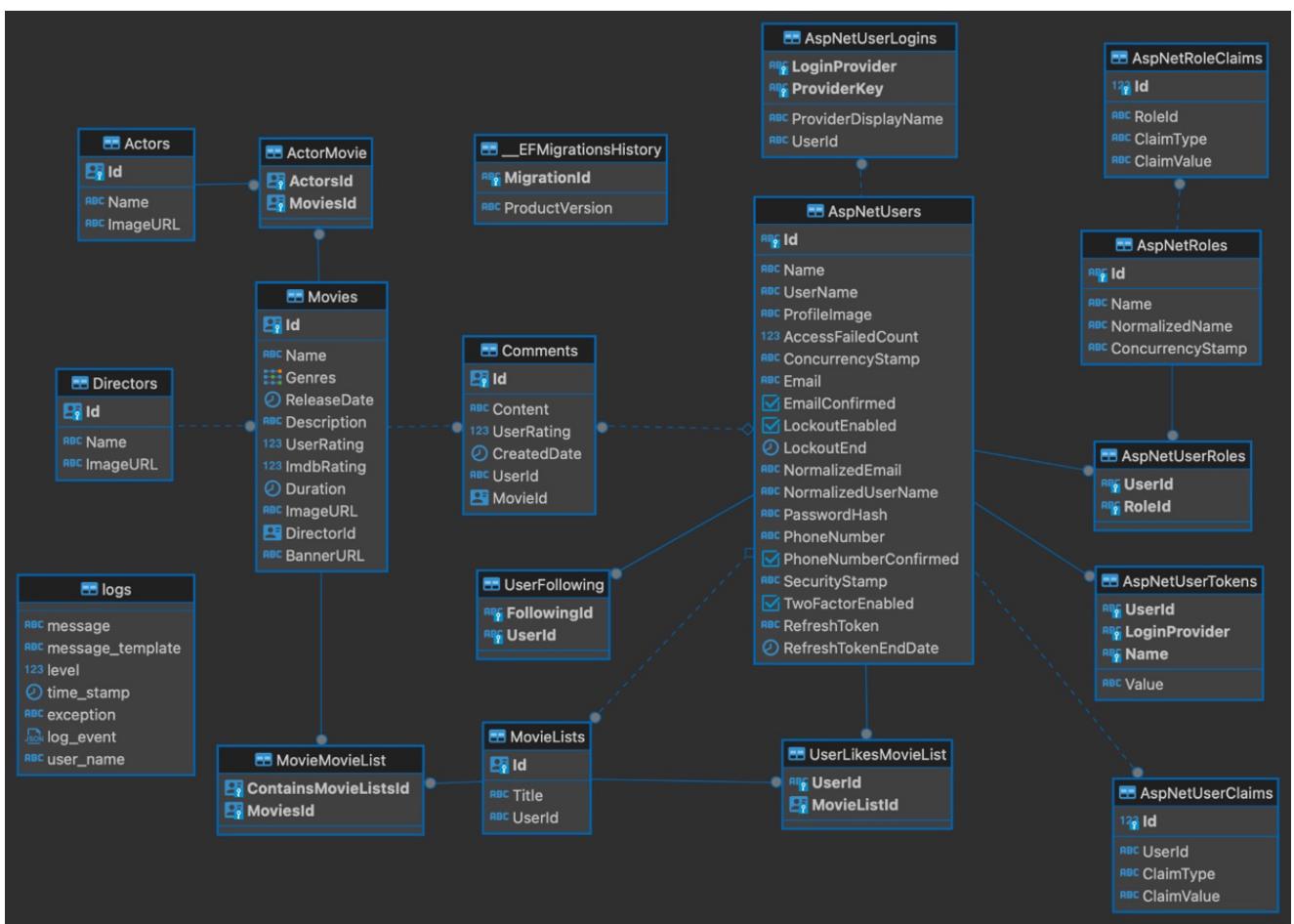
Şekil 1'deki varlık-ilişki diyagramımızı kullanarak ilişki şemamızı oluşturduk. Bu şemayı referans alarak projemizde kullanacağımız veri tabanını oluşturduk.

3. PostgreSQL

postgres - localhost:5432	
Databases	
postgres	
Schemas	
public	
Tables	
ActorMovie	160K
Actors	120K
AspNetRoleClaims	24K
AspNetRoles	24K
AspNetUserClaims	24K
AspNetUserLogins	24K
AspNetUserRoles	24K
AspNetUserTokens	16K
AspNetUsers	64K
Comments	32K
Directors	64K
MovieLists	24K
MovieMovieList	16K
Movies	208K
UserFollowing	24K
UserLikesMovieList	24K
__EFMigrationsHistory	24K
logs	605M
Foreign Tables	

Şekil 3.1 PostgreSQL Veritabanı Tabloları

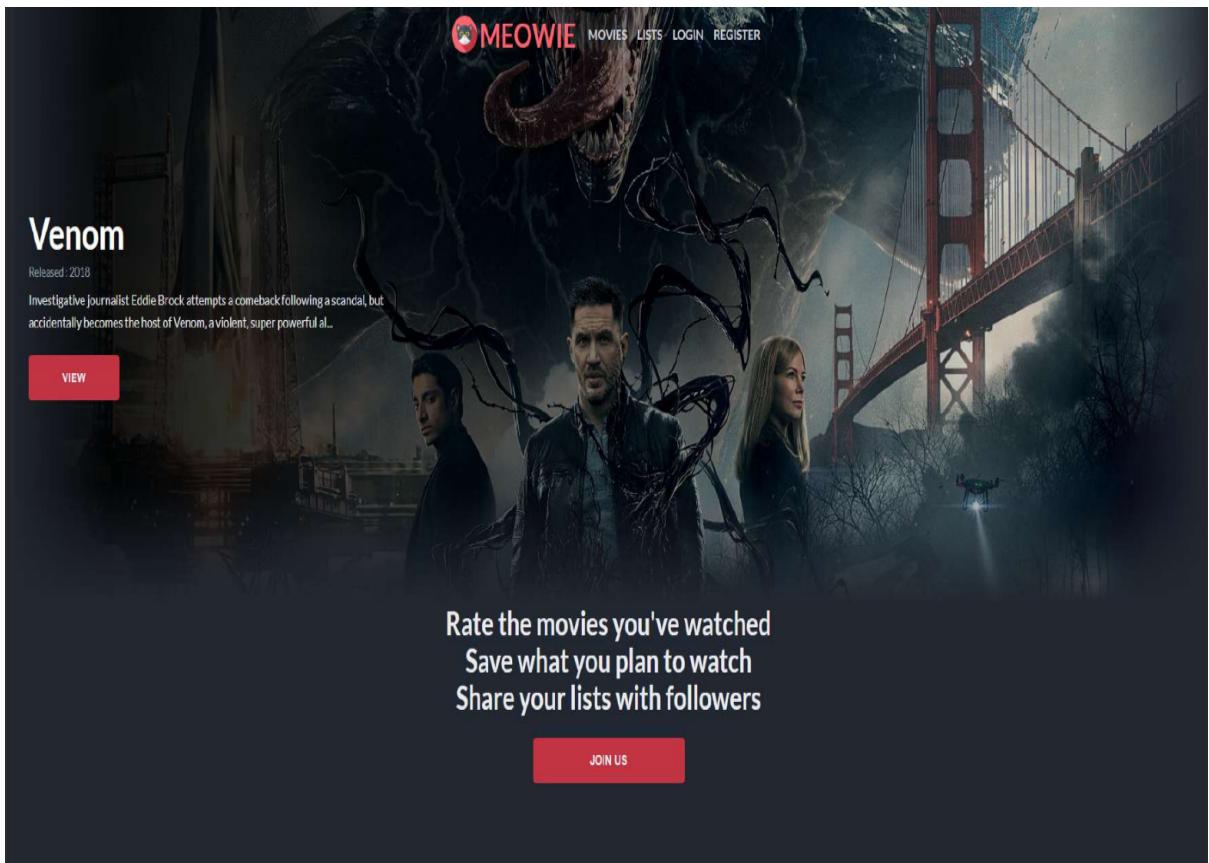
Şekil 2'deki ilişki şemamızı kullanarak PostgreSQL veritabanımızda tablolarımızı oluşturduk.



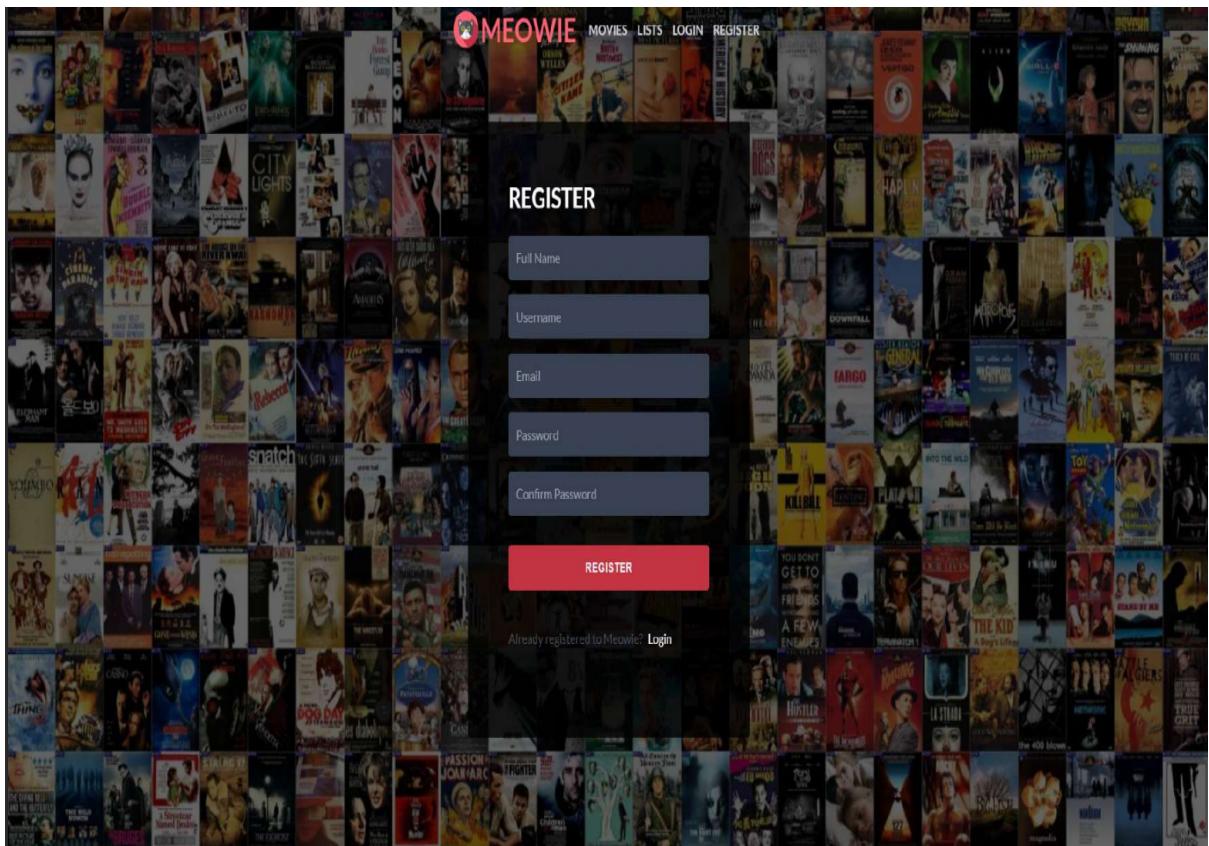
Şekil 3.2 PostgreSQL ER Diagram

BÖLÜM 4

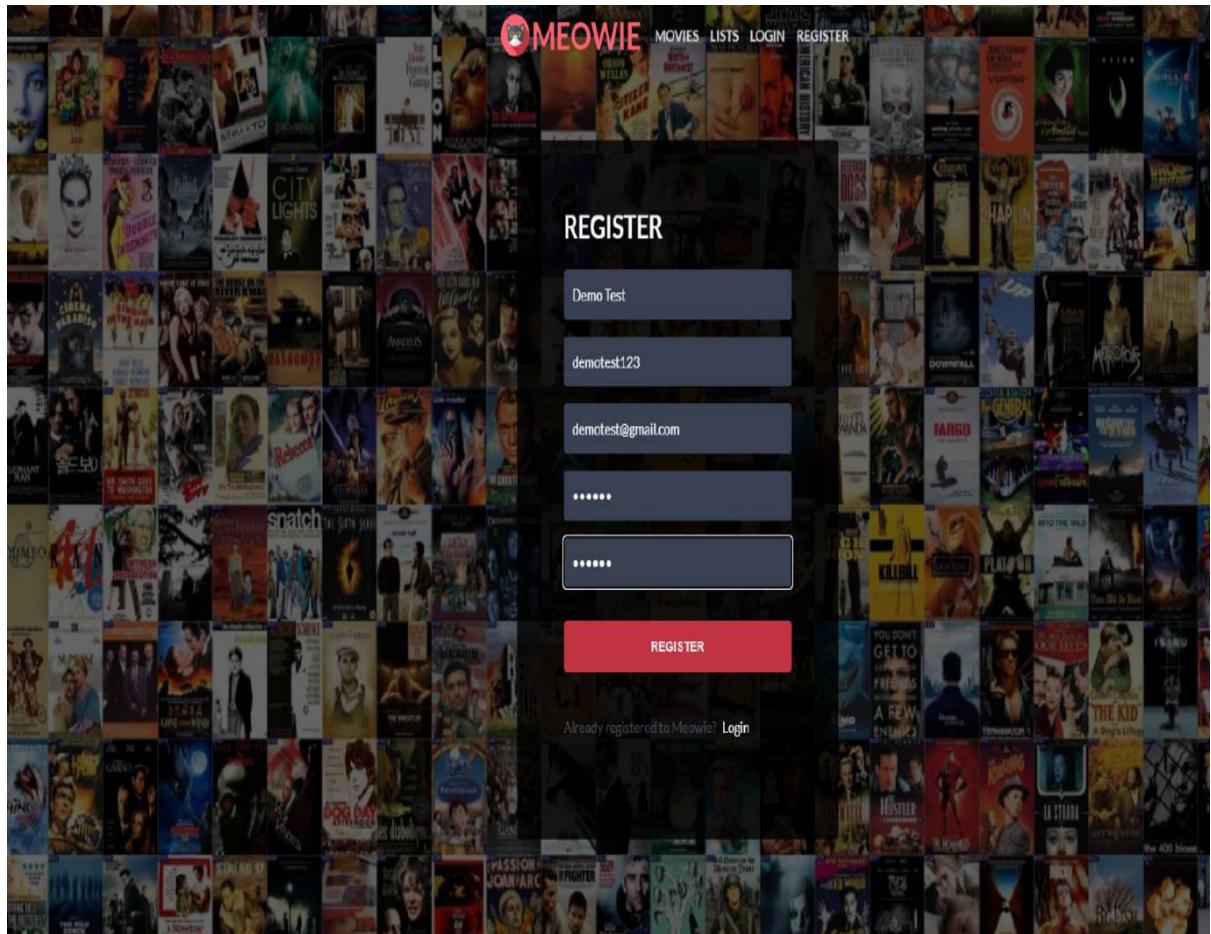
Web Sitesi Ekran Görüntüleri



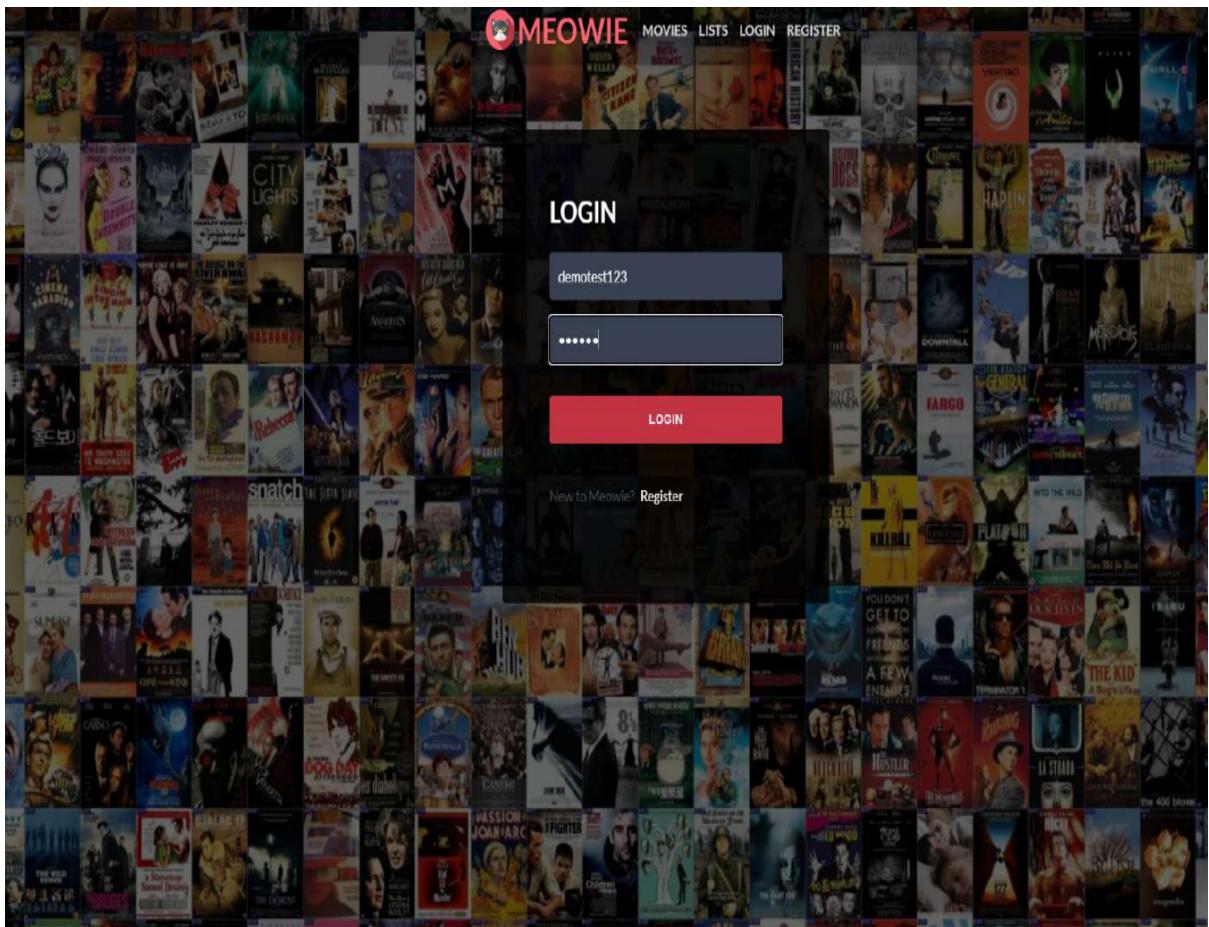
Şekil 4 Ana Ekran Arayüzü



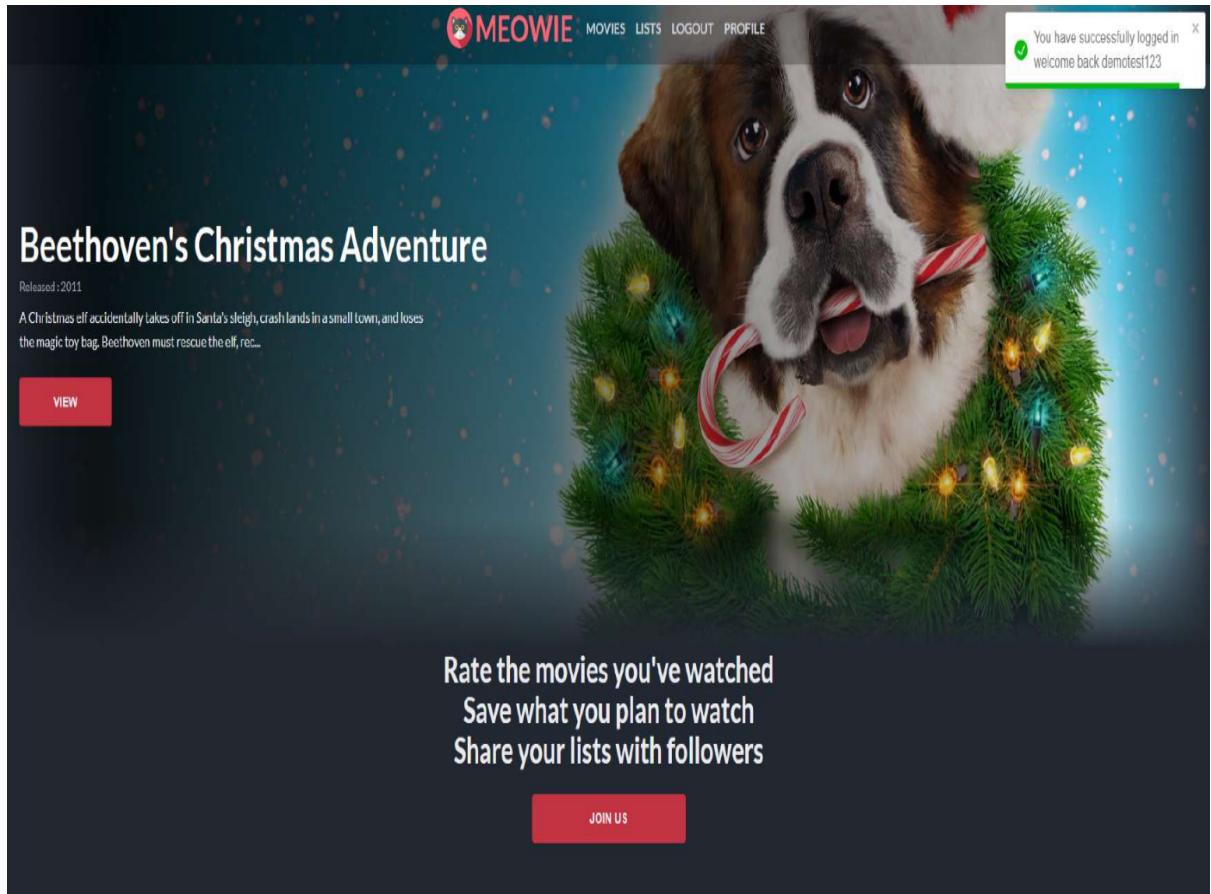
Şekil 5.1 Üye ol Sayfası Arayüzü



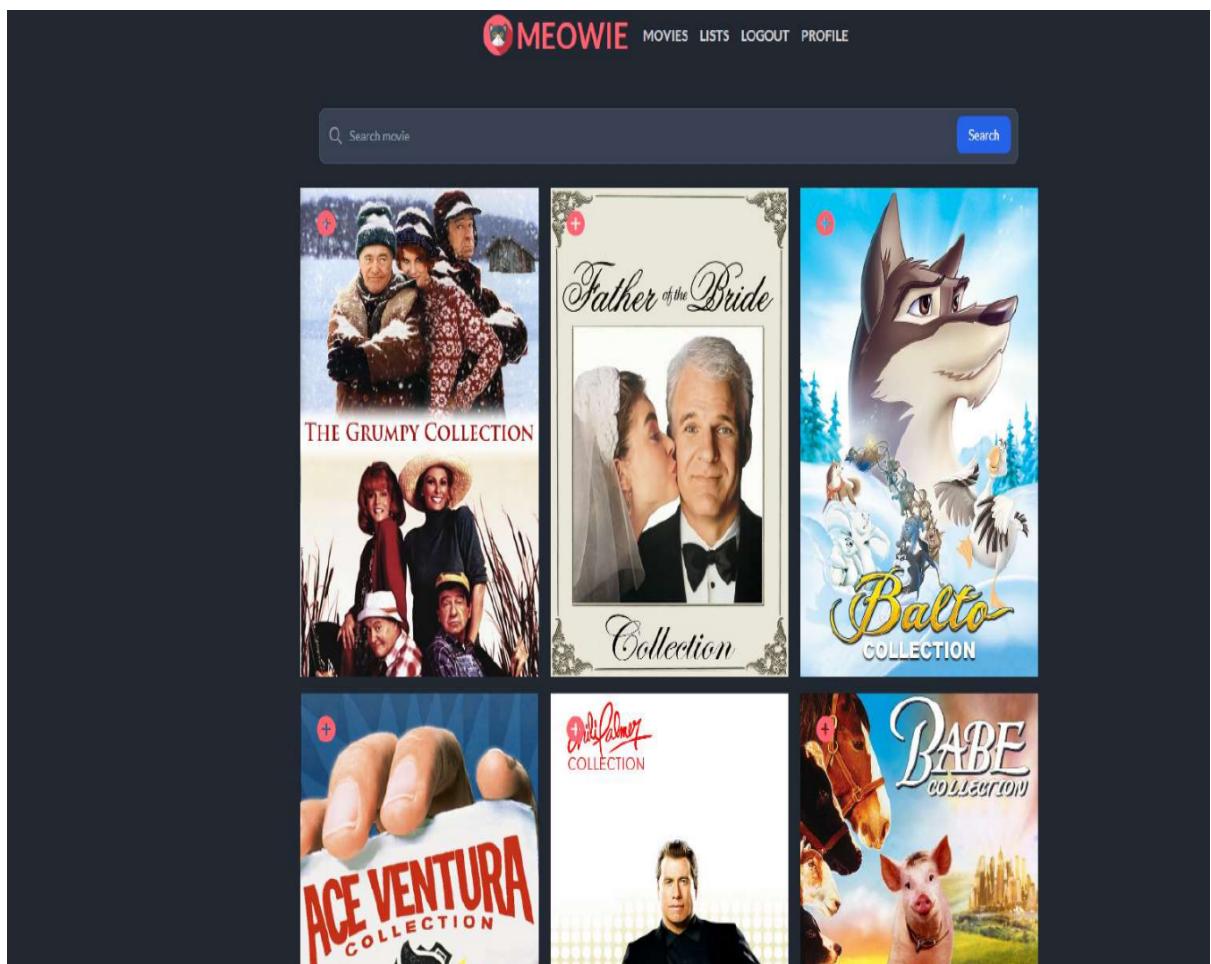
Şekil 5.2 Üye ol Sayfası Arayüzü (Doldurulmuş)



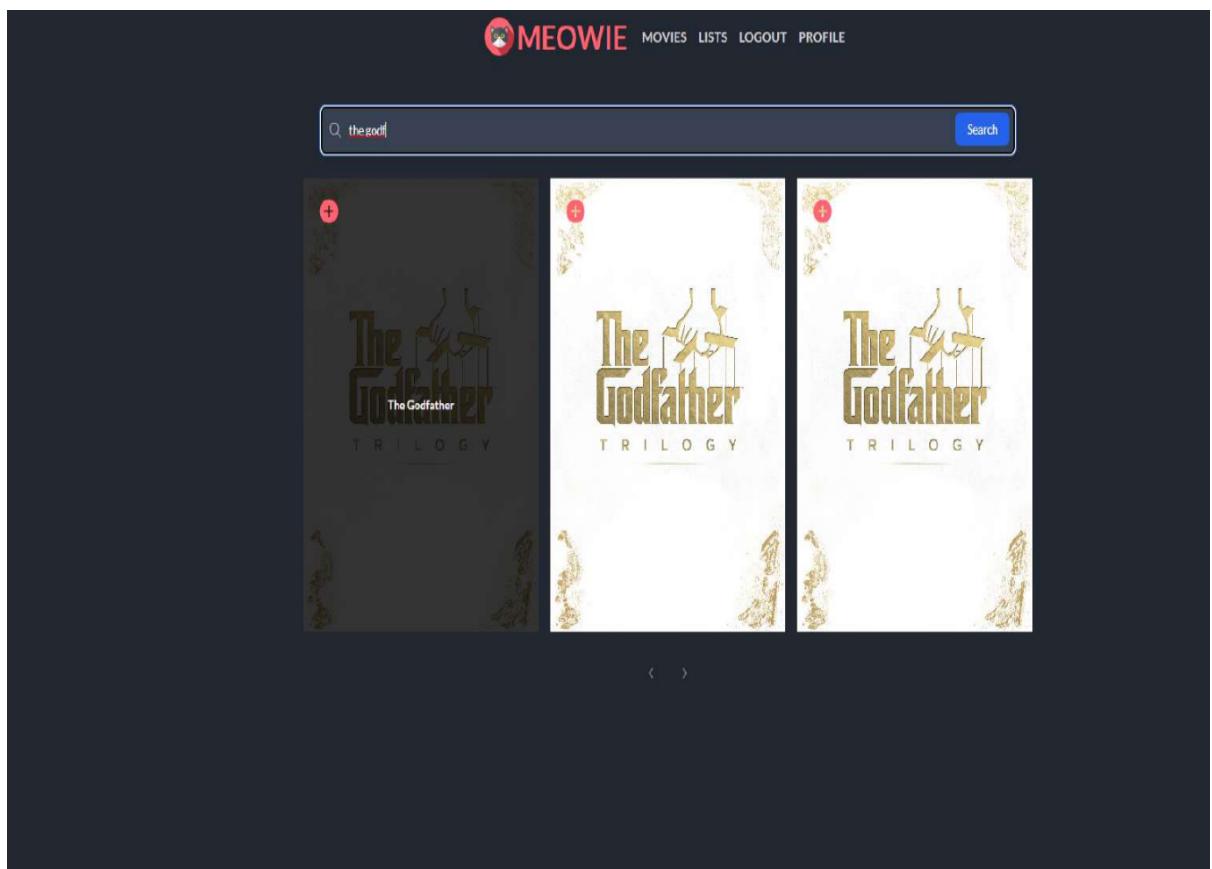
Şekil 6 Giriş Yap Sayfası Arayüzü



Şekil 7 Ana Ekran Arayüzü (Oturum Açık)



Şekil 8.1 Film Arama Arayüzü



Şekil 8.2 Film Arama Arayüzü (Arama Kutusu Doldurulmuş)



Şekil 9.1 Film Açıklama Sayfası Arayüzü

DIRECTOR



Francis Ford Coppola

CASTS



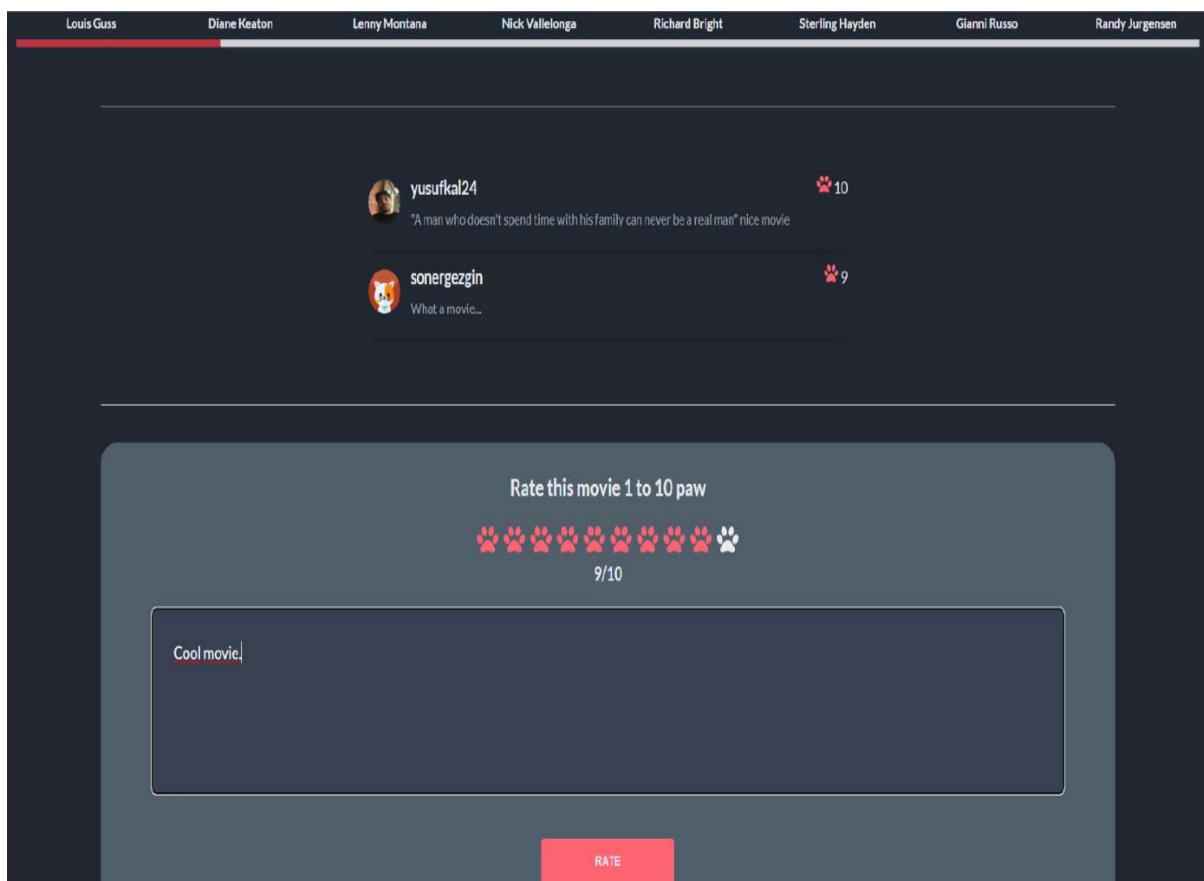
Louis Guss Diane Keaton Lenny Montana Nick Vallelonga Richard Bright Sterling Hayden Gianni Russo Randy Jurgensen

 **yusufkal24**
"A man who doesn't spend time with his family can never be a real man" nice movie

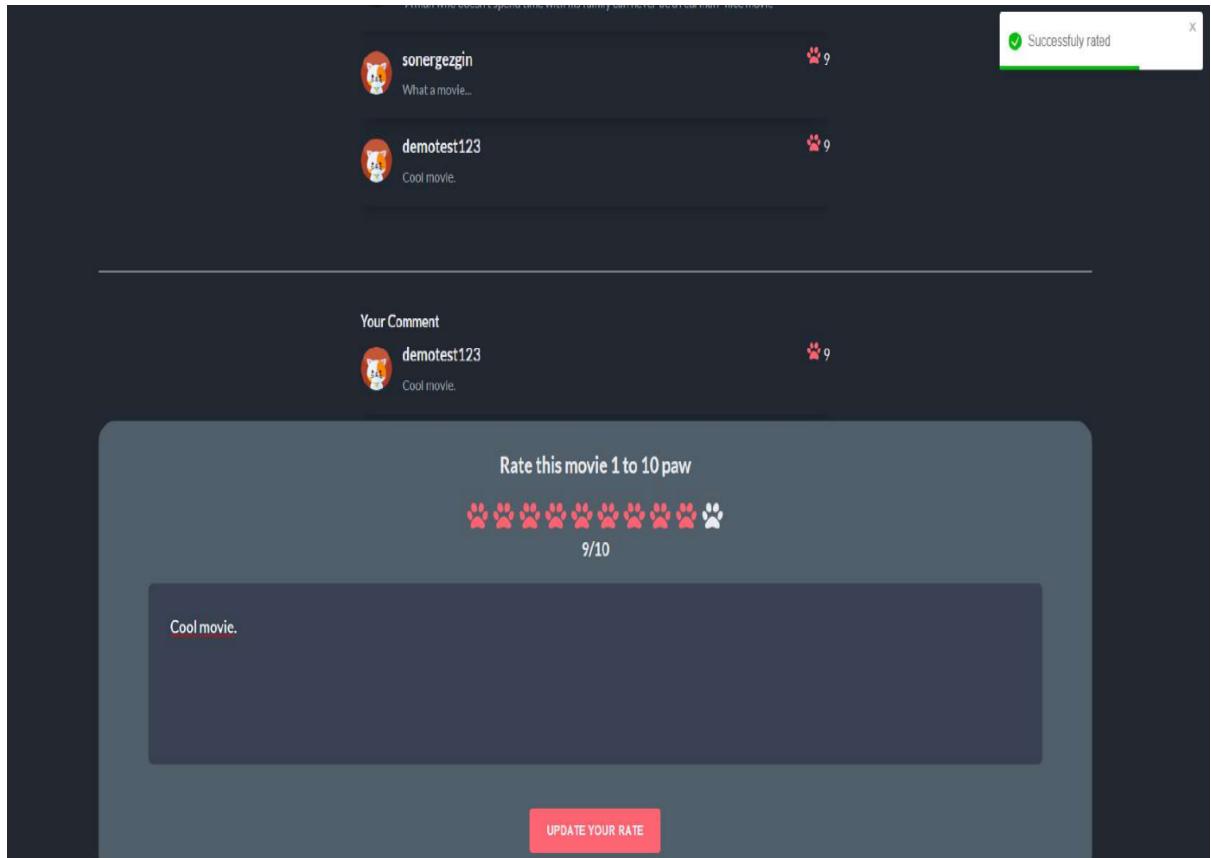
 **sonergezgin**
What a movie

 10
 9

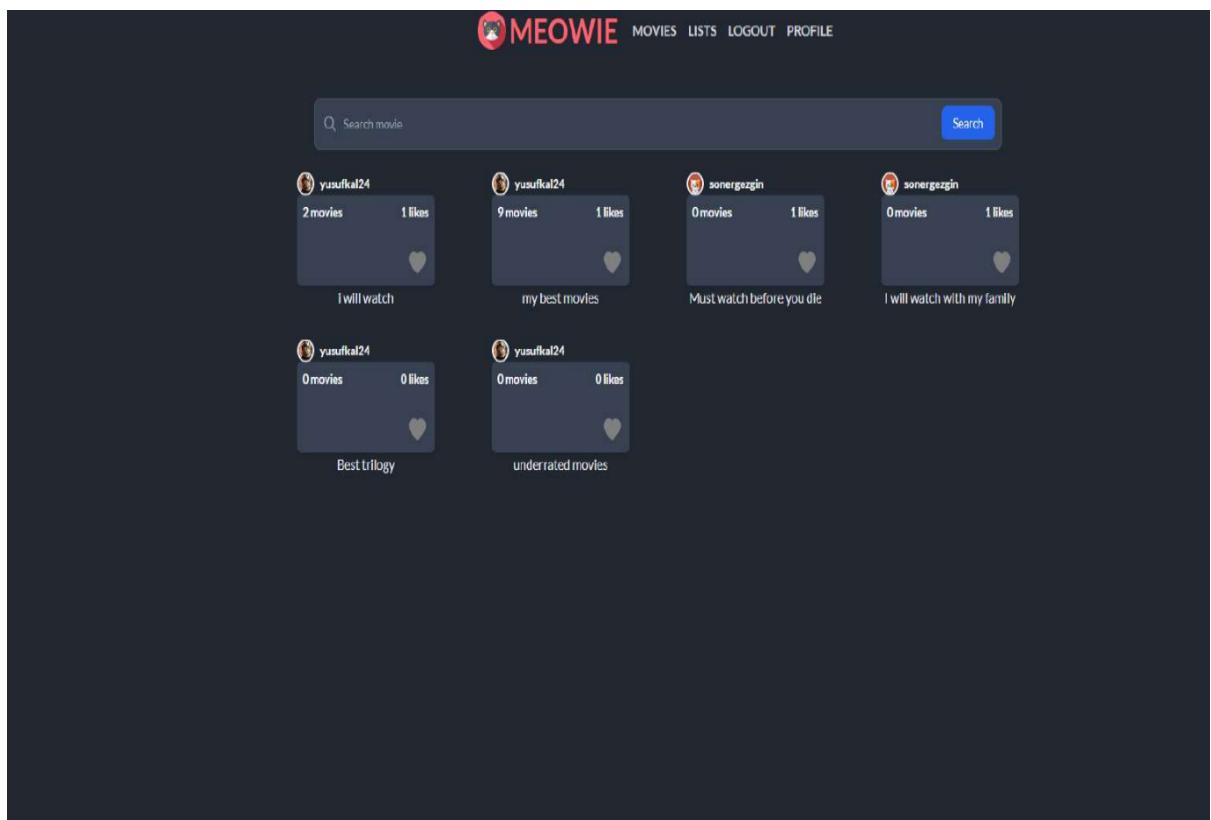
Şekil 9.2 Film Oyuncuları Sayfası Arayüzü



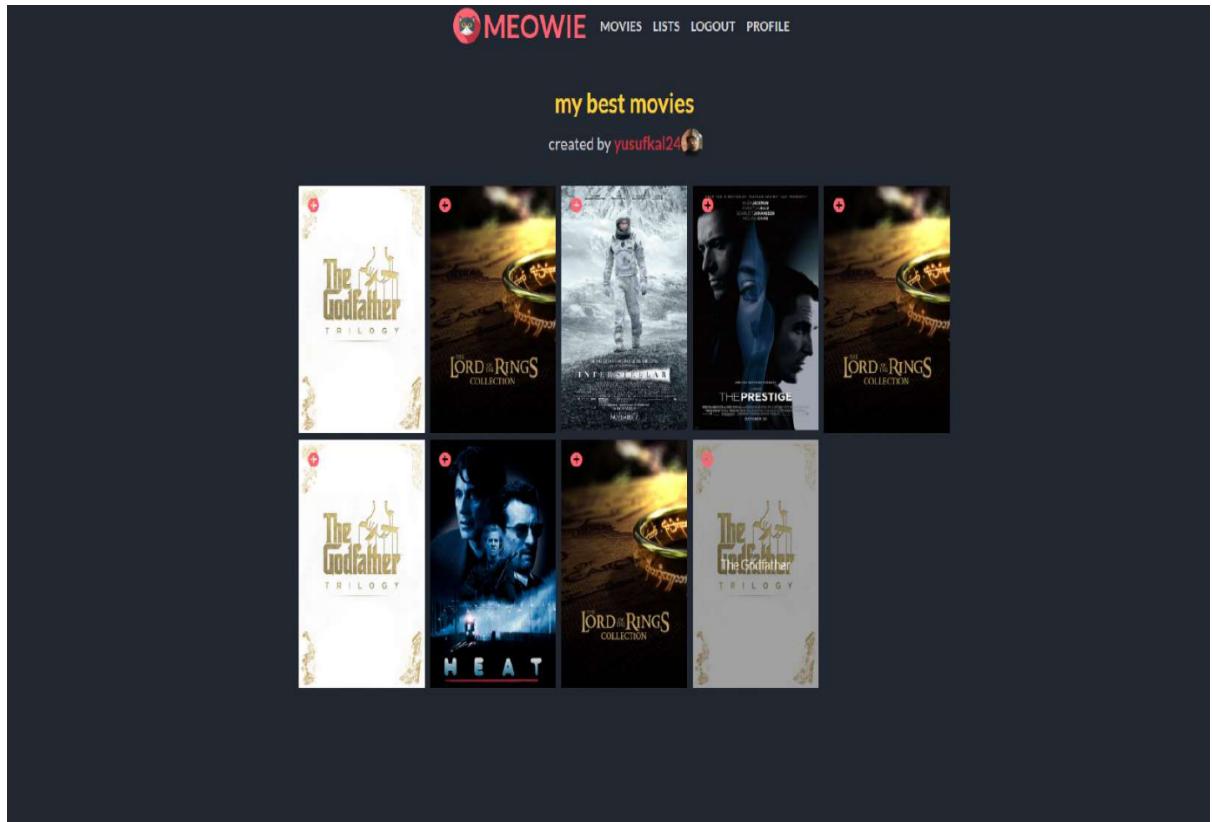
Şekil 10.1 Filmi Değerlendirme Sayfası Arayüzü



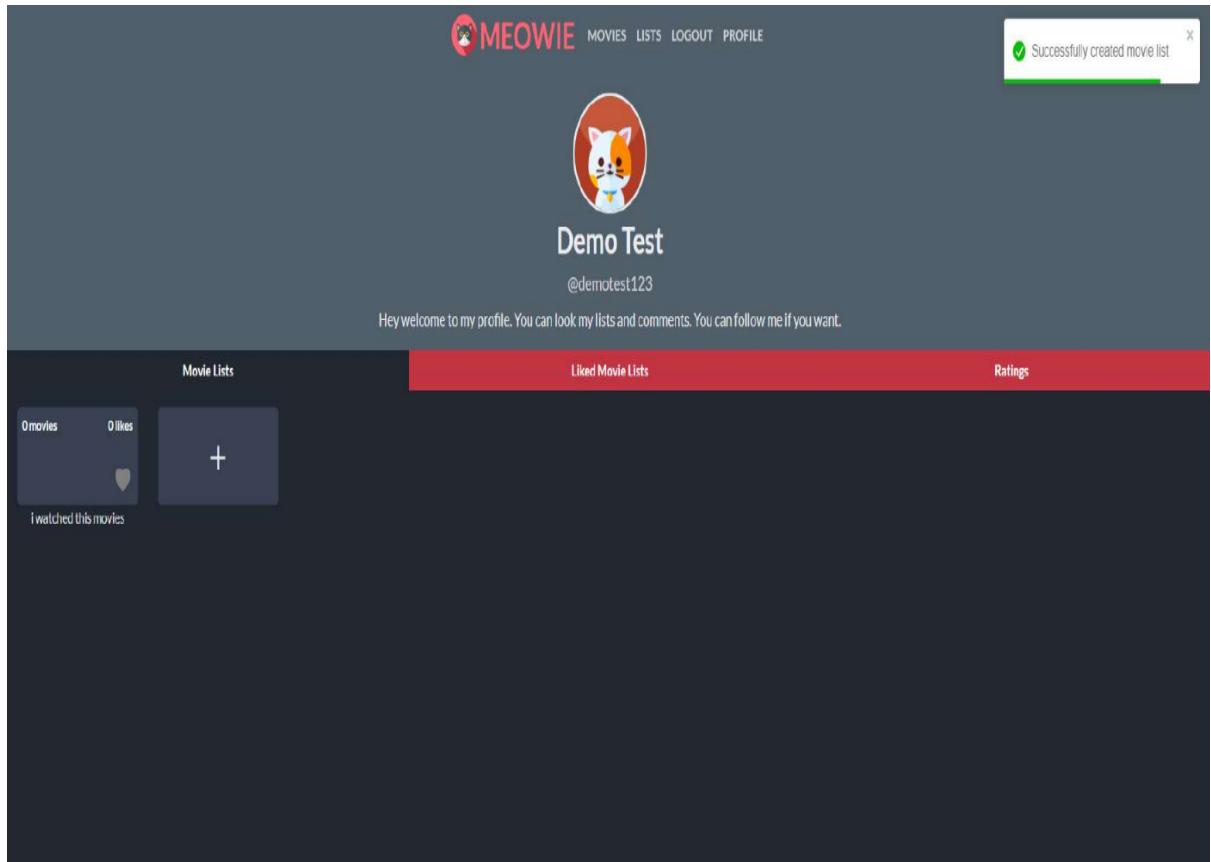
Şekil 10.2 Değerlendirme Güncelleme Sayfası Arayüzü



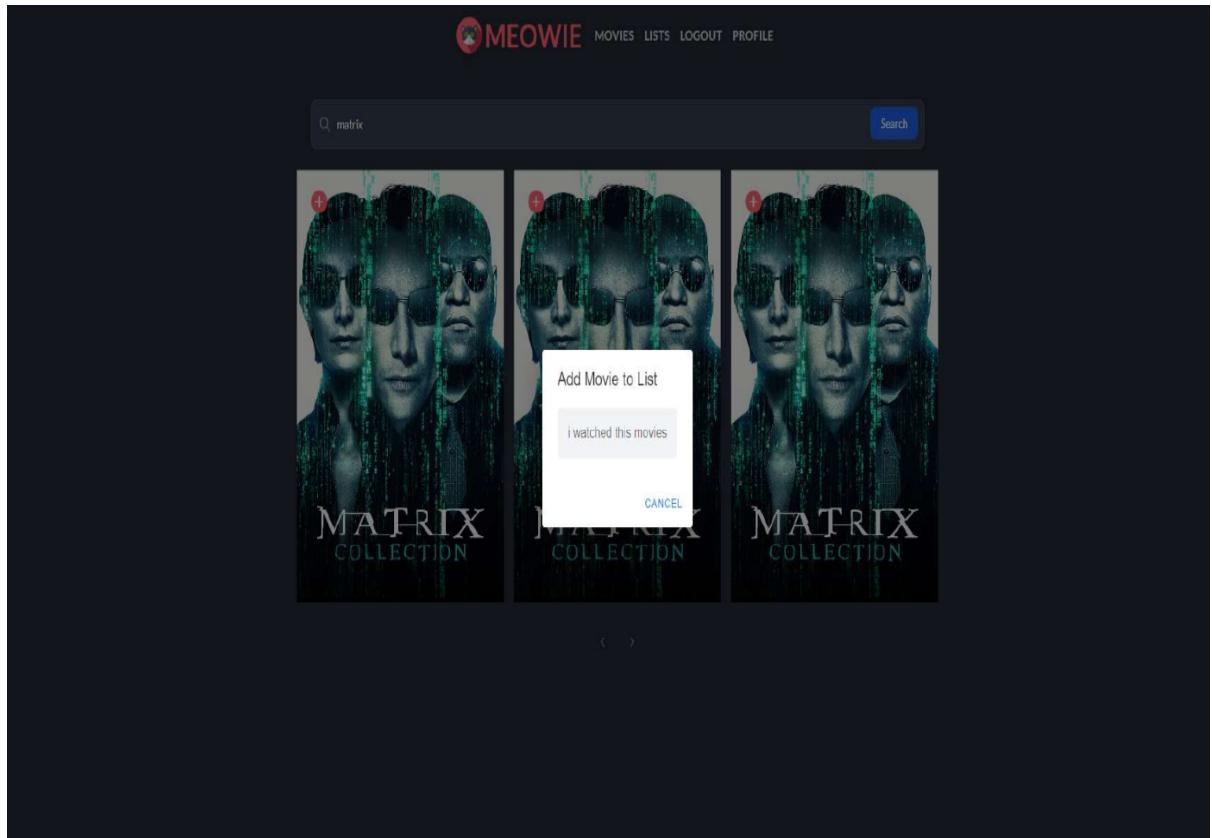
Şekil 11 Film Listeleri Arayüzü



Şekil 12 Film Listesi Görüntüle Arayüzü



Şekil 13 Kullanıcı Profil Sayfası Arayüzü



Şekil 14 Listeye Film Ekle Sayfası Arayüzü



MEOWIE

MOVIES LISTS LOGOUT PROFILE



Demo Test

@demotest123

ne to my profile. You can look my lists and comments. You can follow me if

Sekil 15 Profil Fotoğrafı Ekle Arayüzü

SONUÇ

Bu proje kapsamında, film ve dizi meraklıları için tasarladığımız MEOWIE Film ve Dizi Değerlendirme ve Derecelendirme Platformu'nu tamamladık. Bu platform, kullanıcıların filmleri yorumlayıp değerlendirebildiği, listeler yapıp diğer kullanıcılarla paylaştığı, film izlemeye karar vermeden önce diğer kullanıcıların fikirlerinden faydalandığı ve kendi favori filmlerinin listelerini oluşturabildiği interaktif bir web uygulaması olarak tasarlandı ve geliştirildi.

Proje sürecinde Onion Architecture kullanarak katmanlı mimariye kiyasla veri bağımlılığını en aza indirmeyi amaçladık. Çeşitli tasarım desenleri kullanarak kodun geliştirilebilirliğini ve değiştirilebilirliğini en üst düzeye çıkardık ve böylece mühendislik standartlarına uygun bir yazılım geliştirdik. Code first yaklaşımı ile veritabanı yönetimini kodlar aracılığıyla gerçekleştirdik, kullanıcı yetkilendirme ve kimlik doğrulama işlemlerini JWT token kullanarak sağladık ve CQRS pattern uygulayarak platformun gelecekte daha da geliştirilebilir olmasını sağladık.

Teknik altyapıyı front-end için React, back-end için ASP.NET ve veritabanı için PostgreSQL kullanarak oluşturduk. Bu süreçte veri güvenliği ve gizliliği sağlamak, kullanılan teknolojilerin performansını optimize etmek ve yasal düzenlemelere uymak gibi önemli hususları titizlikle göz önünde bulundurduk.

Projemiz, kullanıcı deneyimini ön planda tutan bir yaklaşım ile geliştirildi ve kullanıcılarla kapsamlı bir film ve dizi değerlendirme platformu sunmayı başardık. Kullanıcıların filmleri değerlendirmeleri, listelemeleri ve paylaşmaları, diğer kullanıcıların da bu değerlendirmelerden faydalananları platformumuzun sağladığı başlıca özellikler arasında yer aldı.

Sonuç olarak, MEOWIE Platformu'nu başarıyla tamamlayarak, belirlediğimiz hedeflere ulaştık. Platform, film ve dizi meraklılarının bekleyenlerini karşılayan, etkileşimli ve kapsamlı bir hizmet sunmaktadır. Bu süreçte edindiğimiz bilgi ve tecrübeler, gelecekteki projelerimizde de bize yol gösterici olacaktır.

KAYNAKLAR

<https://online.visual-paradigm.com/>
<https://www.kaggle.com/datasets/tmdb/tmdb-movie-metadata>
<https://www.gencayyildiz.com/blog/>

ŞEKİLLER LİSTESİ

	<u>Sayfa No</u>
Şekil 1 Entity-Relationship Diagram	7
Şekil 2 Relation Schema	8
Şekil 3.1 PostgreSQL Veritabanı Tabloları	9
Şekil 3.2 PostgreSQL ER Diagram	10
Şekil 4 Ana Ekran Arayüzü	11
Şekil 5.1 Üye Ol Sayfası Arayüzü	12
Şekil 5.2 Üye ol Sayfası Arayüzü (Doldurulmuş)	13
Şekil 6 Giriş yap Sayfası Arayüzü	14
Şekil 7 Ana Ekran Arayüzü (Oturum Açık)	15
Şekil 8.1 Film Arama Arayüzü	16
Şekil 8.2 Film Arama Arayüzü (Arama Kutusu Doldurulmuş)	17
Şekil 9.1 Film Açıklama Sayfası Arayüzü	18
Şekil 9.2 Film Oyuncuları Sayfası Arayüzü	19
Şekil 10.1 Filmi Değerlendirme Sayfası Arayüzü	20
Şekil 10.2 Değerlendirme Güncelleme Sayfası Arayüzü	21
Şekil 11 Film Listeleri Arayüzü	22
Şekil 12 Film Listesi Görüntüle Arayüzü	23
Şekil 13 Kullanıcı Profil Sayfası Arayüzü	24
Şekil 14 Listeye Film Ekle Sayfası Arayüzü	25
Şekil 15 Film Listeleri Arayüzü	26