

Programlama Laboratuvarı

Netflix Veritabanı Yönetim Sistemi

Özge POYRAZ
Kocaeli Üniversitesi
Mühendislik Fakültesi
Bilgisayar Mühendisliği
180202025@kocaeli.edu.tr

Burak Can TEMİZEL
Kocaeli Üniversitesi
Mühendislik Fakültesi
Bilgisayar Mühendisliği
180202024@kocaeli.edu.tr

Özet—Bu çalışmada arayüz bileşenleriyle birlikte bir veritabanı yönetim sistemi oluşturularak, Netflix klonu gerçekleştirilmiştir. Bu uygulamayı geliştirirken Java programlama dili, Swing arayüzü ve Sqlite veritabanı kullanılmıştır.

Anahtar kelimeler— Veritabanı, Veri, Varlık, İlişki, Arayüz, Netflix, Uygulama

I. GİRİŞ

Gerçekleştirdiğimiz proje temel olarak bir veritabanı yönetim sistemiydi. Uygulama kullanıcı tarafından etkileşime girilebilen birtakım arayüz öğelerine sahipti. Bu etkileşimler sonucu ilgili veritabanı üzerinde çok çeşitli işlemler gerçekleştiriliyor, bazen kullanıcıdan çeşitli veriler alınıyor, bazen ise kullanıcıya veriler sunuluyordu.

Netflix içerisinde çeşitli görsel içerikler bulunan ve bunları kullanıcılara sunan bir platform, proje kapsamında sorumlu tutulduğumuz kısım ise içeriklerin yalnızca bilgilerinin kullanıcıya aktarılmasıydı. Böyle bir programın çalışması için bazı temel unsurlar bulunmaktaydı. Bunları sırası ile gerçekleştirmeye başlamadan önce ilk olarak kullanıcının program içerisinde izleyebileceği bir yol haritası çıkardık. Bu şekilde kullanıcının etkileşime ilk başladığı andan programı sonlandırdığı ana kadar ki sürdürdüğü tüm hareketliliği takip edebilecek ve program akışımızı da buna göre yönlendirebilecektik.

Kullanıcı programı çalıştırdığında kendisini giriş sayfası karşılayacak. İsteğe bağlı olarak kayıt olma ya da sisteme giriş işlemlerini gerçekleştirebilecek ve birbirini tetikleyen art arda başka seçenekler ile işlemlerini sürdürebilecekti. Bu akış şemasını referans alarak taslak bir arayüz hazırlama fikrinin projenin gidişatını oldukça rahatlatacağını düşünerek arayüzlerin basit ön tasarımlarını gerçekleştirdik. Daha önceden de bahsettiğimiz bu akışları hayali olarak bu arayüzler üzerine yerleştirmeye başladık.

Arayüzlerin plana dahil olmasıyla birlikte kullanıcı etkileşimi de başlamış oldu. Bu etkileşim sonucunda veriler ile gerçekleştireceğimiz işlemleri de önceden planlamamız gerekmektedir. Bu yüzden sistemimizde kullanacağımız veritabanı modelini bize verilen kılavuza da bağlı kalarak oluşturmaya başladık. Öncelikle bir dizayn programında veritabanını tasarladık ve ardından eksik kısımları oluşturduk. Veritabanı modelimiz tamamen hazır olduğunda akış üzerinden etkileşimleri tek tek tespit edip veritabanına veri girdisi yapan ya da çeşitli verileri elde etmemizi sağlayan sorguları oluşturmaya başladık. Bu aşamadan sonra taslak olarak projemizin çok büyük bir kısmı zaten oluşmuştu. Daha sonrasında uygun teknik şartları oluşturup gerçekleştirmek kalmıştı.

Projemiz için gerekli çalışma ortamlarını hazırlayıp, hazırladığımız tüm bu planlar çerçevesinde geliştirme işlemine başladık.

II. YÖNTEMLER VE PROGRAM MIMARISI

Bu kısımda programın farklı özelliklerini oluşturmak için kullandığımız araçlar ve yöntemler üzerinde durularak ayrıntılı olarak bilgi verilecektir ve program mimarisi daha detaylı bir şekilde açıklanacaktır.

A. Veritabanı

Giriş bölümünde bahsettiğimiz gibi dizayn aşamasından sonra veritabanını oluşturmaya başladık. İlk etapta veritabanı teknolojisi olarak Mysql kullanmaya karar vermiştik bu yüzden ilk şemamızı burada oluşturduk. Daha sonra ilgili tabloları ekledik. Burada bizim doldurmamız istenen bazı kısımlar da vardı. Programımızda beş adet tablo bulunmaktaydı. Bunlar kullanıcı, kullanıcıprogram, program, programtur ve tur tablolarıydı.

Verilen kılavuzdaki ilgili alanları uygun veri türleri ile tablolar içerisinde oluşturduk. Aslında “kullanıcı”, “program” ve “tur” tablosu ana tablolarımızdı. “kullanıcıprogram” ve “programtur” ise ilişki tablolarıydı. Verileri düzgün ve bizden istenilen şekilde tutabilmek için aralarındaki ilişkileri anlamamız gerekiyordu. “kullanıcı” ve “program” tablosu arasında çok çok ilişki vardı. Yani bir kullanıcının birden fazla izlediği program olabileceği gibi, bir programı izleyen birçok kullanıcı da olabiliyordu. Aynı çok çok ilişkiye sahip program ve tur tablosu da mevcuttu. Bunların aralarında da bir ilişki tablosu oluşturulmalıydı. Burada çok çok ilişki olmasının sebebi ise bir programın birden fazla türü olabilmesi aynı zamanda bir türün de birçok programa ait olabilmesiydi.

Bunları belirledikten sonra tablolarımızın temel anahtar değerlerini ayarladık. “kullanıcı” tablosunda “KullanıcıEposta” alanını eşsiz ve birincil anahtar yaptık çünkü sistemdeki her kullanıcı aynı E-posta adresi ile sisteme bir kez kaydolabilecekti. Aynı şekilde program tablosundaki “programAdı” alanı da o tablo için eşsiz ve birincil anahtar olacaktı. Çünkü aynı isimli bir program tablo içerisinde mevcut olmamalıydı. “tur” tablosu altında “TurAdı” alanı da aynı şekilde eşsiz ve birincil anahtar olmalıydı. “kullanıcıprogram” tablosu, “kullanıcı” tablosu ve “program” tablosunun ilişki tablosu olmalıydı yani bu tablo ilişkili olduğu “kullanıcı” ve “program” tablosundaki birincil anahtarları bulundurmalıydı. Bu sebeple bize verilen alanlar haricinde “ProgramAdı” ve “KullanıcıEposta” alanlarını da tabloya birincil anahtar olarak ekledik. İkisinin bir birincil anahtar olmasının sebebi kullanıcının izlediği bir

programın iki kere kaydedilmemesinin gerekmesiydi. Yani bir kullanıcı birçok programı izleyebilecek fakat kullanıcının izlediği programlar defalarca kaydedilemeyecek. Bizim oluşturmamız beklenen “programtur” tablosunda ilişkili olduğu “program” ve “tur” tablosunun birincil anahtarları olan “ProgramAdi” ve “TurAdi” olmalıydı. Bunlar da “programtur” tablomuzda birincil anahtardı. Bu programa bir türün iki kere kaydedilmesini önüyor yani programa birden fazla tür ekleyebiliriz fakat bunlar aynı türler olamaz.

Daha sonra ilişki kurduğumuz tablolarda gerekli yabancı anahtar tanımlamalarını gerçekleştirdik. “kullaniciprogram” tablosundaki “KullaniciEposta”, “kullanici” tablosundaki “KullaniciEpostası” alanını referans alan, “ProgramAdi” ise “program” tablosundaki “ProgramAdi” alanını referans alan yabancı anahtarlardı. Böylece “kullaniciprogram” tablomuzdaki “KullaniciEposta” verisi “kullanici” tablosundaki “KullaniciEposta” değerlerinden farklı bir değer alamayacak. Aynı şekilde bu tablodaki “ProgramAdi” da “program” tablosundaki “ProgramAdi” verilerinden farklı olamayacaktı. Diğer yabancı anahtar eklememiz gereken “programtur” tablosunda “ProgramAdi”, “program” tablosundaki “ProgramAdi” alanını; “TurAdi” alanı da “tur” tablosundaki “TurAdi” alanını referans alıyordu. Bu şekilde yabancı anahtarları verdiğimizde referans aldıkları alanlardaki verilerden farklı değerler alması önlenmiş oldu.

Tüm bunların sonunda elimizde tabloları tamamen oluşturulmuş bir şema oldu. Daha sonrasında proje yalnızca sunumda değerlendirileceği ve tek bilgisayarda çalışması gerekeceği için ayrıca veritabanının taşınabilirliğini sağlamak amacıyla Sqlite veritabanı sistemini kullanmaya karar verdik. Bu Mysql şeması üzerinden Sqlite üzerinde tekrardan veritabanımızı oluşturduk. Burada Sqlitedeki farklı veri tiplerine dönüşümler yaparak yeniden oluşturmayı tamamladık.

B. Veri Yapıları, Özel Sınıflar ve Sistemler

Proje boyunca çeşitli veri yapılarını kullanarak işlemler gerçekleştirdik. Bazı noktalarda kendi özel sınıflarımızı yazarak arayüz elemanlarını kendimize göre düzenledik, yeniden oluşturduk. Bazı noktalar da ise programımız için faydalı sistemler oluşturduk. Bu kısımda bu yapılara değineceğiz.

Programımızda tüm pencereler statik olarak Netflix ana sınıfı altında tanımlı bir şekilde bulunmakta ve buradan yönetilmektedir. Bu şekilde pencereler arasındaki iletişim kolay bir şekilde sağlanmaktadır. Aynı şekilde tüm veritabanı yapıları kendi özel veritabanı sınıfı içerisinde tanımlıdır ve gerekli sorgu işlemlerinde statik tanımlı bu yapılar çağrılarak kullanılmaktadır. Bu da her sınıfta tekrar tekrar veritabanı elemanlarını yeniden tanımlama gereksinimini ortadan kaldırarak işimizi oldukça kolaylaştırmaktadır.

Arkaplan ve FilmKutusu sınıfları özel olarak yaratılmış sınıflardır ve kendi arayüz bileşenlerimizdir. Burada çeşitli swing bileşenleri yeniden düzenlenerek oluşturulmuşlardır.

Programın bazı noktalarında sorgulardan dönen sonuçlar Collection sınıfındaki veri yapıları kullanarak tutularak çeşitli işlemlere sokulmuşlardır. Örnek olarak herhangi bir arama türünde işlem gerçekleştirdiğinizde sorgudan dönen filmler bir ArrayList içerisinde dönmekte ve bu liste üzerinde ilgili panele ekleme, çizme ve hizalama işlemleri yapılmaktadır.

Arama işlemlerini gerçek zamanlı gerçekleşmesi için aramanın yapılacağı textbox üzerinde document listener event sistemiyle değişiklik tespiti yapılmış. Her değişiklikte arama ve film kutularının çizdirilmesi işlemleri tekrardan yapılarak dinamik arama sistemi oluşturulmuştur.

Arayüzümüzün sahip olduğu bir diğer özellik olan film görsellerinin gösterilmesi işlemi bir data klasörü içerisinde ilgili filmin veritabanındaki adı ile bu klasördeki aynı isimli resim dosyasının eşleştirilip gösterilmesiyle sağlanmaktadır.

Program boyunca bunlar gibi daha birçok irili ufaklı geliştirme aşamasını kolaylaştıracak sistemler oluşturulup kullanılmıştır.

C. Kullanıcı Arayüzü

Arayüzü Swing kullanarak oluşturduk. Swingin sunduğu kişiselleştirme özelliklerinden faydalanarak darcula kütüphanesi ile temel swing bileşenlerine karanlık tema desteği sağladık. Geliştirme ortamımız olan Netbeans’in dahili form editörü aracılığıyla free-layout düzeninde çeşitli formlar üzerinden arayüzü pencere pencere gerçekleştirdik.

Çoğu noktada arayüz bileşenlerini override ederek amacımız doğrultusunda düzenledik. Program açılışında kullanıcı girişi için karşımıza bir adet giriş formu gelmektedir. Bu formda bir label üzerinden ürettiğimiz dinamik bir arkaplan nesnesini kullanmaktayız. Aynı şekilde görsellik için bu yapıyı diğer formlarda da dahil etmekteyiz. Windowsun titlebarını devre dışı bırakarak yine kendi yapımız olan bir titlebar ile görsel bir bütünlük yakalamaya çalıştık. Bu düzeni de yine tüm pencerelerde sürdürmeye çalıştık. Giriş penceresinden sonra kayıt işlemi için de kullanıcı önüne birtakım formlar sunulmakta. Bu formlar işlevlerine göre farklı farklı arayüz bileşenleri bulundurmaktadır.

Kullanıcı giriş işleminden sonra yönetim panelimizde film seçiminin yapılacağı ekranda bir JPanel nesnesi çoğu fonksiyonuyla override edilip yeniden düzenlenerek film kutucuklarını tutacak bir yapı haline getirilmiştir. JLabel nesneleri ile kutucuklu bir arayüz tasarımı gerçekleştirilmiştir. Bu film kutucukları özel sistemler kısmında anlattığımız şekilde çalışarak dosya sistemindeki önceden tanımlanmış film görsellerini veritabanındaki program adı ile eşleştirerek ana sayfada kullanıcıya ilgili filmi görsel bir şekilde bildiriyor ve etkileşime girmesini sağlıyordu. Bu arayüz için pencere içerisinde dinamik olarak yerleşme ve hizalanma işlemlerini gerçekleştirdik.

Arayüzün detaylı görünümü raporun ilerleyen kısımlarında mevcuttur.

III. GELİŞTİRME ORTAMI VE KULLANILAN DİL

Projeyi Java Programlama dilinde JDK 8 ile Windows işletim sistemi üzerinde gerçekleştirirken, geliştirme ortamı olarak Apache Netbeans kullandık. Son proje dosya yapısı Netbeans dosya formatındadır ve javac derleyicisi ile derlenmiştir. İçerisinde harici kütüphane ve dosya bağımlılıklarına sahip çalıştırılabilir jar dosyası ve exe bulunmaktadır.

IV. PROGRAMIN GENEL YAPISI VE TASARIMI

Program temel olarak çeşitli form pencerelerinden oluşuyor ve kullanıcının bu ekranlar üzerinden seçtiği çeşitli

işlemleri gerçekleştiriyor. Bu işlemler arasında giriş yapma, üye olma, film aratma, bölüm seçme, puan verme vs. gibi özellikler bulunuyor.



Kullanıcı Giriş Ekranı



Anasayfa

V. PROGRAMIN ÇALIŞTIRILMASI VE KULLANILMASI

Programın detaylı olarak çalıştırılması Ek 4 kısmında görseller ile adım adım verilmektedir. Daha detaylı bir bilgi için lütfen o kısma bakınız.

Program 2_180202024_180202025 içerisindeki Netflix.exe ya da aynı klasör içerisindeki Netflix.jar, calistir.bat üzerinden eğer java sisteminizde düzgün bir konfigürasyon ile kuruluysa direkt olarak çalıştırılabilir. Yukarıdaki giriş ekranını aynı şekilde görüyorsanız program sorunsuz bir şekilde başlamıştır. Eğer java konfigürasyonunuz bozuk ise veya path değişkeni tanımsız ise tıkla-çalıştır işlemi gerçekleşmeyebilir. Programın bağımlılık duyduğu dosyalar olduğu için komut satırı üzerinden başka bir klasör dizinindeyken yapacağınız çalıştırma işlemleri programın alt dizinindeki dosyalara ulaşmasını engelleyecek ve hatalı çalışmasına sebep olabilecektir. Bu şekilde bir çalışmada görseller ve veritabanı yüklenmeyecektir. Bu kısım ekte ayrıntılı olarak gösterilmiştir. Çalıştırma işlemi düzgün kurulmuş bir java ile tıkla-çalıştır şeklinde gerçekleştiriniz ya da “program ile aynı dizinde” komut satırı ile çalıştırınız.

Proje üzerinden derleme yapılmak istenirse Projenin dosya yapısı Netbeans projesi formatındadır. Proje klasörünü Netbeans ortamında açabilirsiniz. Projeyi tüm ayarları yapıldı bir şekilde derleyebilirsiniz.

Program içerisinde tüm işlemler arayüz elemanları üzerinde detaylı olarak yazmaktadır. Program akışını yöneterek istediğiniz işlemleri gerçekleştiriniz. Görseller ile adım adım çalışma aşamaları ekte mevcuttur.

VI. GELİŞTİRME AŞAMASINDA KARŞILASILAN PROBLEMLER

Proje boyunca karşılaştığımız ilk ve tek büyük problem veritabanı sistemiyle ilgiliydi. Projenin sunum esnasında çalıştırılabilir olması isteniyordu ve Mysql sistemi sunucuda çalışan bir veritabanı olduğu için program çalıştırılmadan önce ilgili bilgisayarda local bir sunucuya kurulması ya da herhangi harici bir sunucu üzerinden bağlantı yapılması gerekiyordu. Problemi aşmak için ilk etapta veritabanımızı komple bir sql script haline getirdik. Böylece projenin test edileceği bilgisayarda Mysql komut satırı ya da Workbench gibi arayüzlü programlarda sql script çalıştırılarak kolayca veritabanı local sisteme kurulabilecek ve program içerisindeki bağlantı stringi değiştirilerek bu veritabanına bağlanılabilecekti. Fakat daha sonradan proje kapsamında Sqlite veritabanı sistemi kullanılabilecek hale getirilince ve Sqlliteda veritabanı sunucu üzerinde çalışmayıp bir dosya gibi çalıştığı için projeyi bu sisteme geçirmeye karar verdik. Böylece veritabanımızı bir dosya gibi karşı tarafa kolay ve hızlı bir şekilde ulaştırabilecektik ve herhangi bir bağlantı ayarının değiştirilmesine gerek kalmayacaktı. Biz de veritabanını tekrardan oluşturduk. Burada Sqlite ve Mysql arasında bazı değişken tipi farklılıkları bulunmaktaydı. Fakat sorgular ve java içerisindeki bağkanti kütüphanesi fonksiyonları neredeyse tamamen aynı olduğu için hızlı bir şekilde dönüşüm işlemini tamamladık.

VII. DENEYSSEL SONUÇLAR

Projeyi gerçekleştirmeden önce teorik olarak veritabanı tasarımını ve dizaynını gerçekleştirdik. Bu noktada Veritabanı yönetim sistemleri dersinde gördüğümüz ilişkisel cebir üzerinde pratik yapma imkanı bulduk. İlk defa ilişkisel cebri uygulamalı olarak kullandık. İlk defa veritabanı dizayn programlarını kullanarak elde ettiğimiz sonuçları referans aldık. Bir programı baştan aşağıya adım adım tasarlayarak tasarım deneyimimizi geliştirdik. Arayüzü gerçekleştirirken ilk defa bir projede form editörlerini kullandık bu noktada farklı layout sistemleri üzerinde deneyimler gerçekleştirme fırsatı bulduk. Form editörlerinin farklı layout sistemlerini deneyerek birbirleri arasında karşılaştırıp ihtiyacımıza göre kullandık.

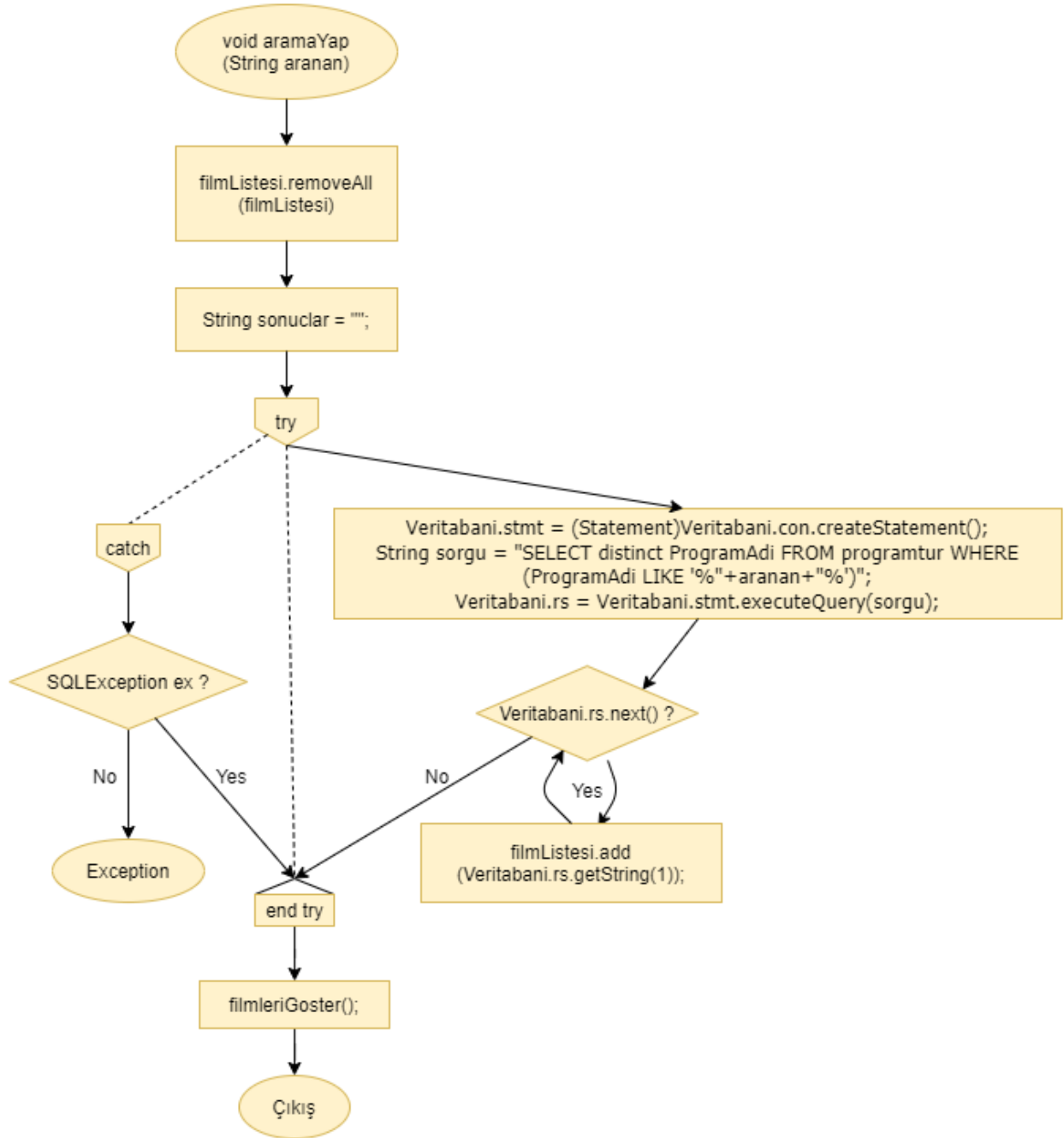
VIII.SONUÇ

Bu projeyi gerçekleştirerek veritabanı yönetim sistemlerinin uygulanması konusunda oldukça önemli bilgiler edindik. Ayrıca arayüz üzerinde çeşitli işlemler gerçekleştirerek son kullanıcıya yönelik bir programın oluşma aşamalarını net olarak gözlemledik.

IX. KAYNAKLAR

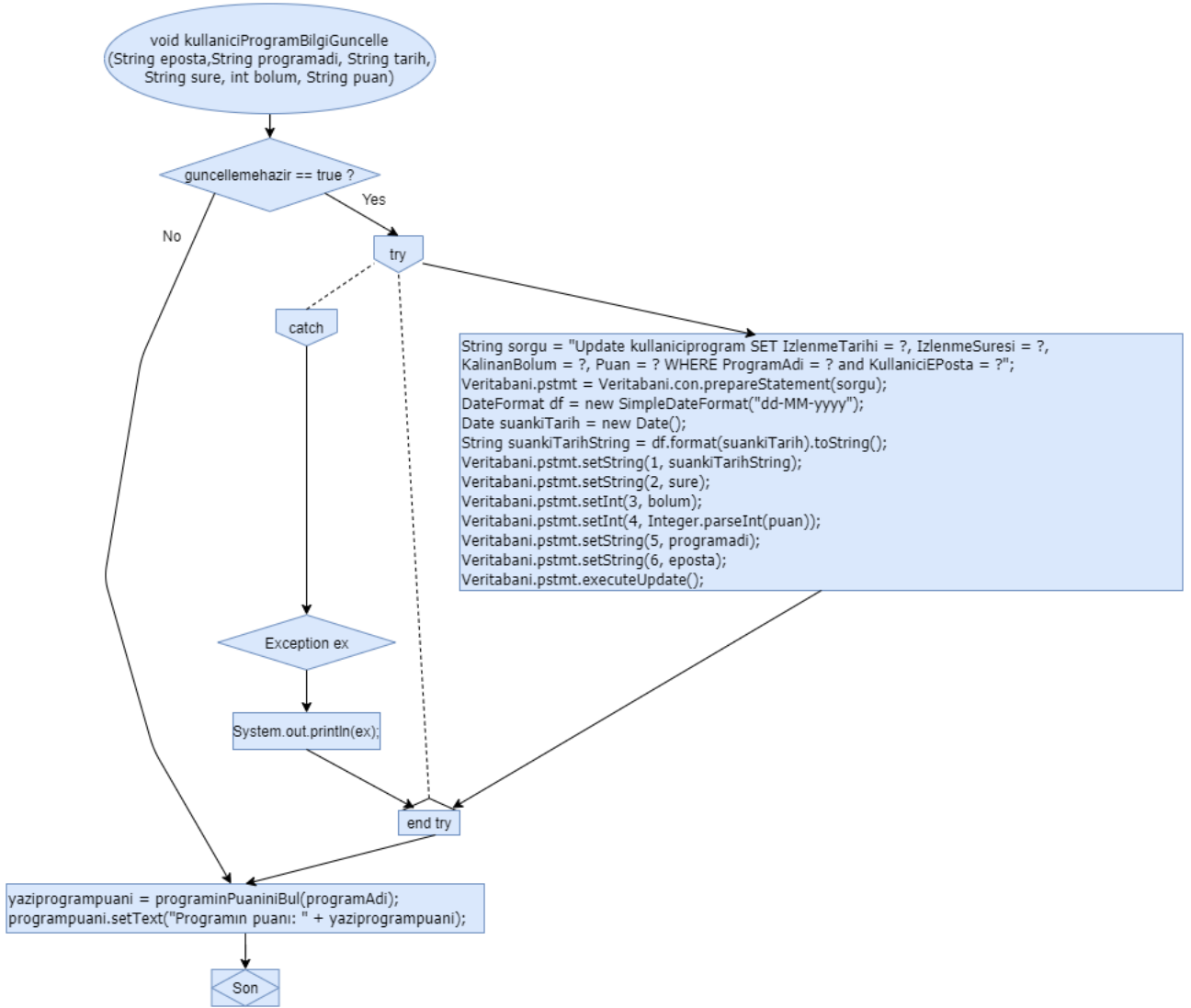
- (1) “Java Documentation:Creating a gui with swing” , <https://docs.oracle.com/javase/tutorial/uiswing/index.html>
- (2) “Sibel Somyürek Ders Notları”, http://sibelsomyurek.com/veritabanı/ders_notları.html

Ek 1 -Akış Diyagramı 1



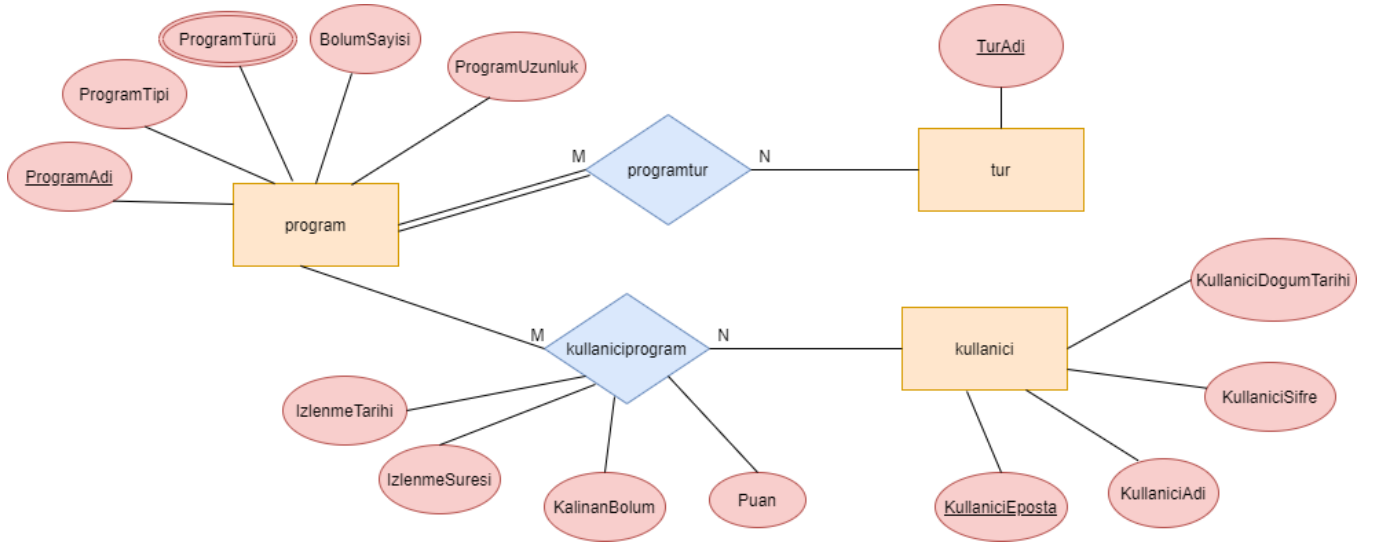
Arama fonksiyonunun akış şeması

Ek 2 – Akış Diyagramı 2



“kullaniciprogram” tablosunda bilgi güncellemesi yapan fonksiyonun akış şeması

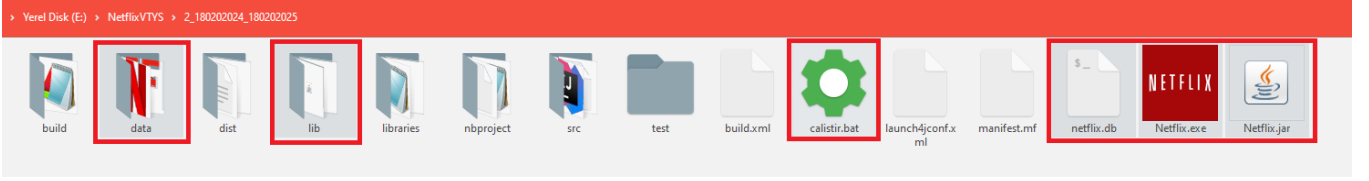
Ek 3 – Veritabanı Varlık, İlişki Diyagramı



Veritabanı varlık, ilişki diyagramı

Ek 4 – Programın Adım Adım Görseller İle Detaylı Çalıştırılması

Programı çalıştırmadan önce sisteminizde minimum JRE 8 sürümünün düzgün bir konfigürasyonla hatasız bir şekilde kurulu olduğundan ve java path değişkeninin ayarlı olduğundan emin olunuz. Aksi halde tıkla-çalıştır işlemi gerçekleşmeyebilir.



Programı çalıştırmak için yukarıda bulunan dosyalardan Windows için Netflix.exe, calistir.bat diğer işletim sistemleri için Netflix.jar'ı tıklayınız.



Kullanıcı giriş paneli bu şekilde karşınıza geliyorsa program çalışmıştır. Eğer bu ekran karşınıza gelmediyse ve program düzgün bir şekilde başlamıyorsa java path değişkeninizi ve jrenizi tekrar kontrol etmeniz gerekmektedir. Eğer program komut satırından başlatmak istiyorsanız şu şekilde bir çalıştırmadan kaçınmalısınız!

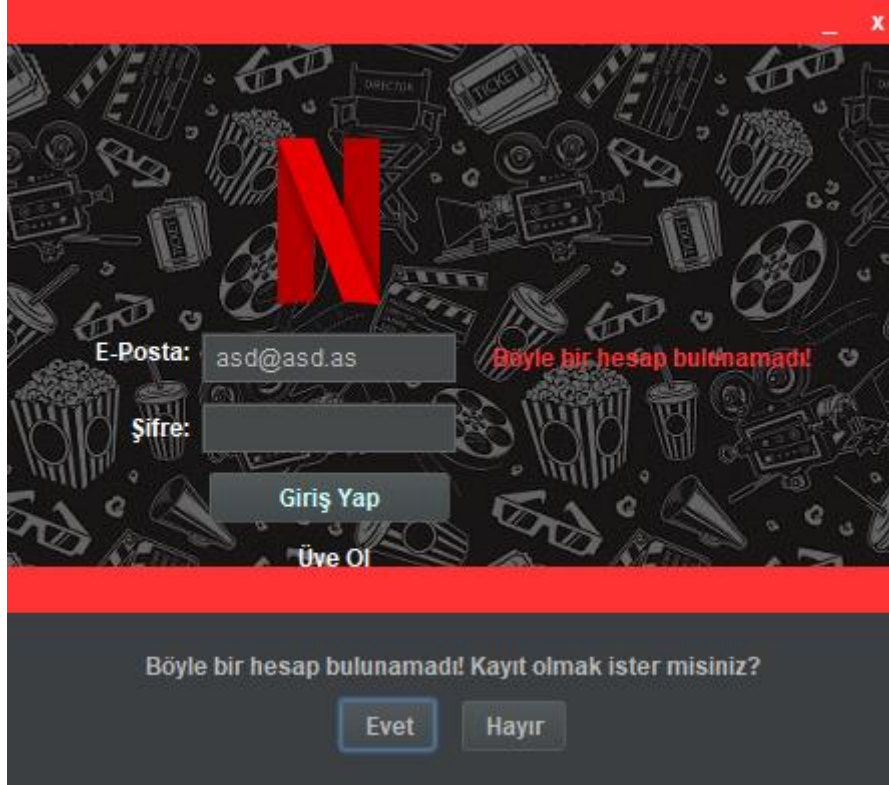
```
C:\Program Files\Java\jre1.8.0_251\bin>java -jar "E:\NetflixVTYS\2_180202024_180202025\Netflix.jar"  
HATA - Veritabanı bulunamadı!
```

Bu şekilde farklı dizinde yapacağınız bir komut ilgili jar'ı o dizinde çalıştıracağı için program alt dizinindeki bağımlılıklarına ulaşamayacaktır ve hata verip kapanacaktır. Komut ile çalıştırma yaparsanız tüm bağımlılıklarıyla program komutu çalıştırdığınız dizine taşınmanız gerekmektedir. Bağımlılıklar ilk görselde gösterilmiştir. Ya da komutu jar dosyasının alt dizininde yürütmeniz gerekmektedir. Ve son olarak program çalışırken veritabanını harici bir program ile görüntüleyip müdahale etmeyiniz. Bu programda Sqllitedan kaynaklı aksaklıklara yol açacaktır. Aynı anda sadece tek bir program ile veritabanına bağlanınız.

Programı sorunsuz bir şekilde çalıştırdıysanız;



Üye ol bağlantısı ile üye olma penceresine gidebilirsiniz.
Ya da programda mevcut olmayan bir hesap girerseniz program tarafından bu ekrana yönlendirileceksiniz.



Karşınıza kayıt ekranı gelecek bilgileri doldurup kayıt işlemine devam edin.

A registration form with a dark background featuring a pattern of movie-related icons like film reels, clapperboards, and popcorn. A large red 'N' logo is centered at the top. The form fields are: 'Ad:' with 'Selçuk' entered, 'E-Posta:' with 'selcuk@gmail.com' entered, 'Şifre:' with '.....' entered, and 'Doğum Tarihi:' with dropdowns for '09', '04', and '1992'. A 'Devam Et' button is at the bottom.

Ad: Selçuk


E-Posta: selcuk@gmail.com

Şifre:

Doğum Tarihi: 09 04 1992

Devam Et

En sevdiğiniz 3 türü seçerek kayıt işlemi tamamlayın. Eğer 3 tür seçmezseniz program size bu konuda uyaracaktır.

A genre selection screen with a dark background and movie-related icons. The title 'En Sevdiğiniz 3 Tür' is at the top. Below it are ten genre buttons: 'Aksiyon ve Macera', 'Bilim ve Doğa', 'Gerilim', 'Anime', 'Bilim Kurgu ve Fantastik', 'Drama', 'Belgesel', 'Komedi', 'Romantik', 'Çocuk ve Aile', 'Korku', and 'Reality Program'. A 'Kayıt işlemi tamamla' button is at the bottom.

En Sevdiğiniz 3 Tür

Aksiyon ve Macera Bilim ve Doğa Gerilim Anime

Bilim Kurgu ve Fantastik Drama Belgesel Komedi

Romantik Çocuk ve Aile Korku Reality Program

Kayıt işlemi tamamla

Kayıt işlemi tamamlayınız.



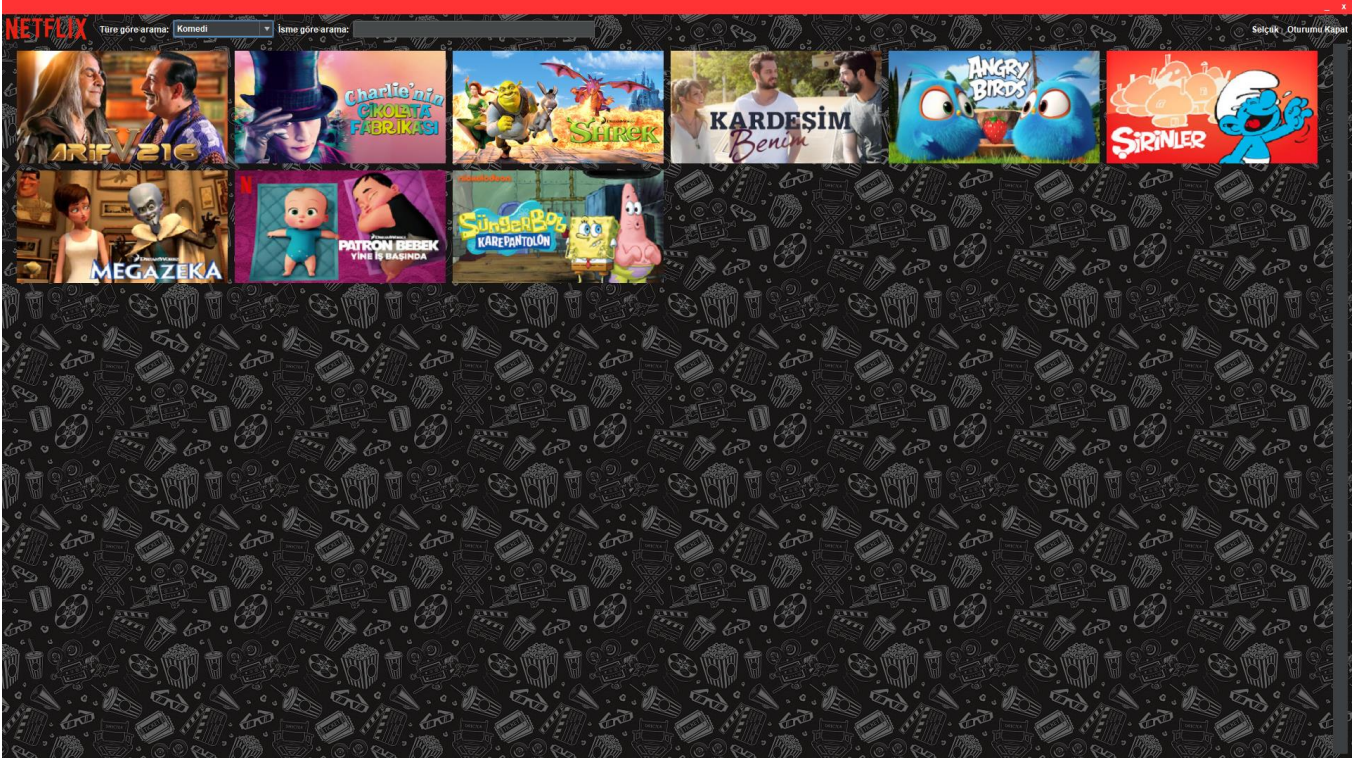
Bu şekilde bir pencere ile seçtiğiniz kategorilerdeki en yüksek puanlı ikişer film size gösterilecek ve kayıt işleminiz tamamlanacak. Beni anasayfaya götür diyerek yönetim paneline gidebilirsiniz.



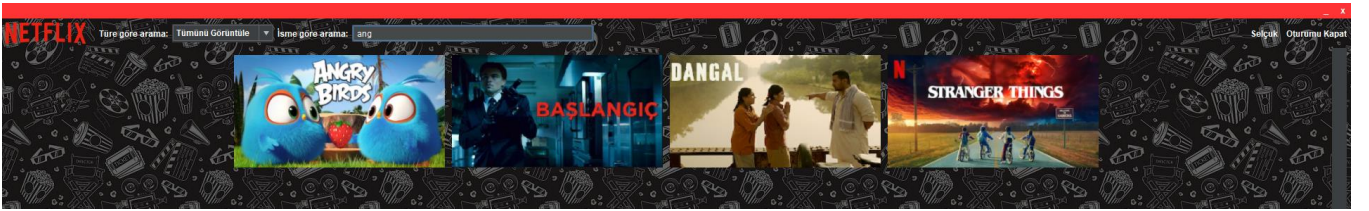
Karşınıza bu şekilde bir anasayfa gelecek. Sol üstten arama yapabilir. Sağ üstten programdan ya da oturumdan çıkış yapabilirsiniz. Aramayı türe göre ya da isme göre gerçekleştirebilirsiniz.

Aşağıda 2 farklı arama türü bulunmaktadır.

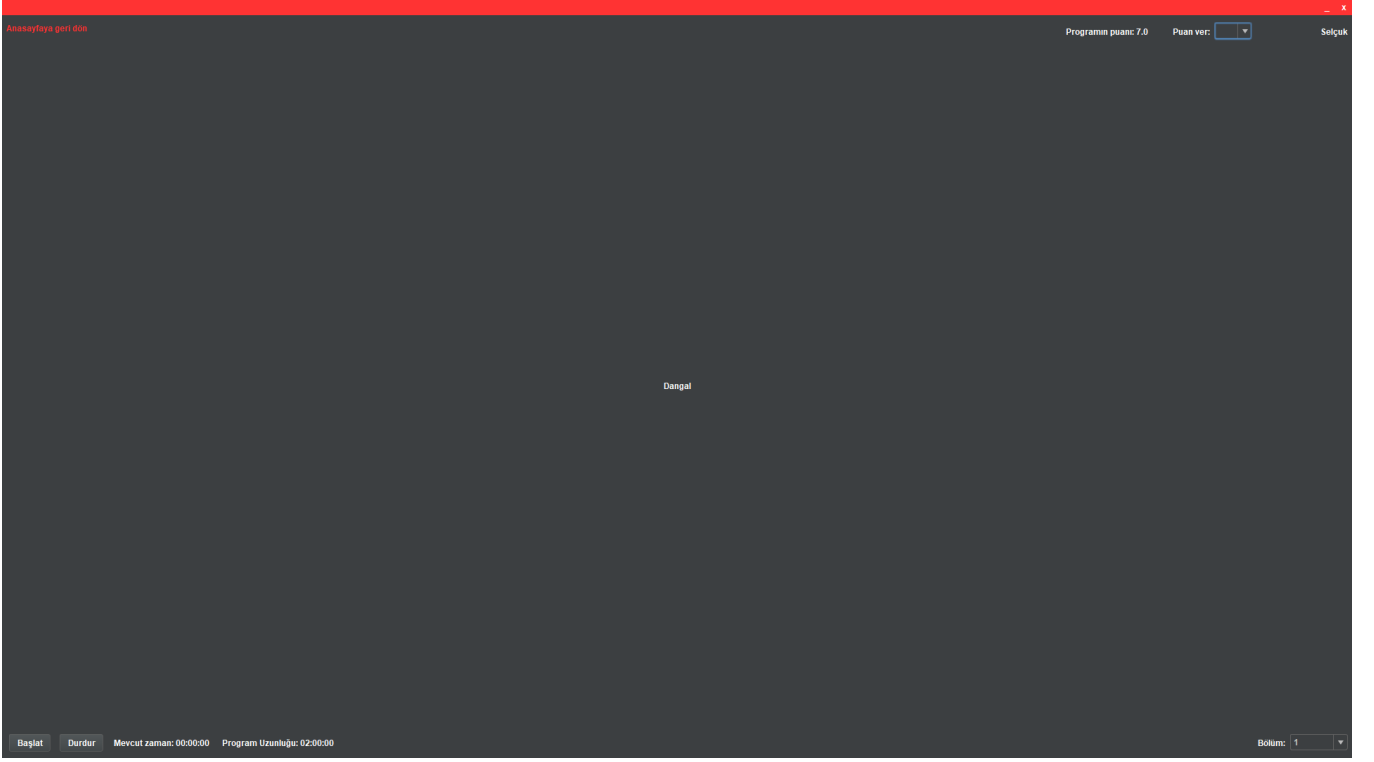
Komedi Kategorisine göre arama işlemi



İsme göre arama işlemi



Arama işlemini gerçekleştirdikten sonra ilgili programa tıklayarak izleme ekranına gidebilirsiniz.



İzleme ekranında sol üstteki “Anasayfaya dön” ile tekrar anasayfaya dönebilirsiniz. Sağ üstteki çıkış butonu ile programdan çıkabilirsiniz. Sağ üstte programın ortalama puanını görebilirsiniz. Puan ver butonu ile puan verebilirsiniz. Burada bir rakam olmaması henüz puan vemediğinizi göstermektedir. Puan vererseniz, verdiğiniz puan programın genel puanını etkileyecektir. Sağda mevcut kullanıcı adını görebilirsiniz. Solda programın zamanının kontrolü mevcuttur. “Başlat” ve “Durdur” ile izleme işlemini gerçekleştirebilirsiniz ve mevcut zamanı ilerletebilirsiniz. Programdan çıkış yaptığınızda ya da anasayfaya döndüğünüzde bu zaman ve ilgili tüm diğer veriler kaydedilecektir. Sağda bölüm seçimi yapabilirsiniz. Filmler tek bölümden oluşmaktadır. Fakat dizi ve reality programlarda bölümler arasında geçiş yapabilirsiniz. Bir bölümün izleme süresinin sonuna geldiğinizde otomatik olarak diğer bölüme geçiş yapılacaktır.

Program genel olarak bu aşamalardan oluşmaktadır. Bu şekilde farklı akışlarla programı kontrol edebilirsiniz. Veritabanını aynı anda sadece tek bir program ile görüntüleyiniz. Aksi halde veritabanına erişimde hatalar oluşabilir.