# Logo, company name Description automatically generated

# Opis rješenja

Rješenje „MovieScape“ napravljeno je za korisnika koji želi ostati u toku s novitetima u svijetu filma i serija. On konstantnim pretraživanjima vanjskih izvora predstavlja nove i popularne filmove i serije u cijelom svijetu. Osim te mogućnosti, rješenje prilikom prvog korištenja pita korisnika za odobrenje korištenja njegove lokacije čime osim korisne mini ikone za opis vremenske prognoze, pretražuje bazu za filmovima i serijama koje su trenutno najpopularnije na njegovom govornom teritoriju.

„MovieScape“ uz početne popularne filmove i serije, pomoću korisnikovih odabira žanrova ponuđuje korisniku top 10 filmova i serija za njega. Korisnik je za te funkcionalnosti obavezan prijaviti se preko Facebook profila.

Za svaki film i seriju, moguće je dobiti detaljne informacije o njima, dok prilikom pregleda detalja, korisnik ima opciju ocijeniti taj sadržaj kako bi bolje podesio algoritam za predlaganje sadržaja.

# Popis svih korištenih izvora s opisom u koju svrhu su se koristili

# 

|  |  |  |  |
| --- | --- | --- | --- |
| **Izvor** | **URL** | **Metoda** | **Opis** |
| Facebook API | https://developers.facebook.com/docs/facebook-login/ | Login | Korisniku se omogućuje prijava putem Facebookovog logina |
| Mapquest API | https://developer.mapquest.com/documentation/geocoding-api/ | Reverse | Dohvaća podatke o korisnikovoj lokaciji |
| OpenWeather API | https://openweathermap.org/api | Weather | Dohvaća podatke o vremenu |
| TMBD API | https://developers.themoviedb.org/3/getting-started/introduction | Movies, Series | Dohvaća podatke o filmovima i serijama |

Facebook API se koristio kako bi se korisnicima aplikacije omogućila prijava u aplikaciju putem Facebook Logina. Također se nabavilo korisnikovo ime i prezime kako bi se moglo prikazati na kartici korisnikovog profila.

Mapquest API se koristio kako bi se na temelju geografskih koordinata korisnika(širina i dužina) moglo dohvatiti podatke o korisnikovom gradu i zemlji. Ovi podaci su bili proslijeđeni dalje kako bi se moglo odrediti vremensku prognozu i filmove koji su popularni u korisnikovoj državi.

OpenWeather API se koristio kako bi se dohvatila trenutna vremenska prognoza za korisnikov grad. Informacije o vremenu u obliku ikone i kratkog opisa su bili prikazani u kartici korisnikovog profila.

TMBD API je bio najvažniji izvor podataka iz kojega su se dohvaćali filmove i serije kako bi se mogli prikazivati na aplikaciji. Koristilo se nekoliko njegovih različitih API-ja za te dohvate.

# Profil korisnika

Profil korisnika u aplikaciji opisan je njegovim punim imenom i email adresom. Ti podaci se spremaju u bazu podataka nakon što se korisnik prijavi u aplikaciju preko društvene mreže Facebook. Podaci se dobiju preko Facebook API-ja te se uz autentifikacijski token pohranjuju u bazu podataka u tablicu user. Ta tablica sastoji se od atributa :

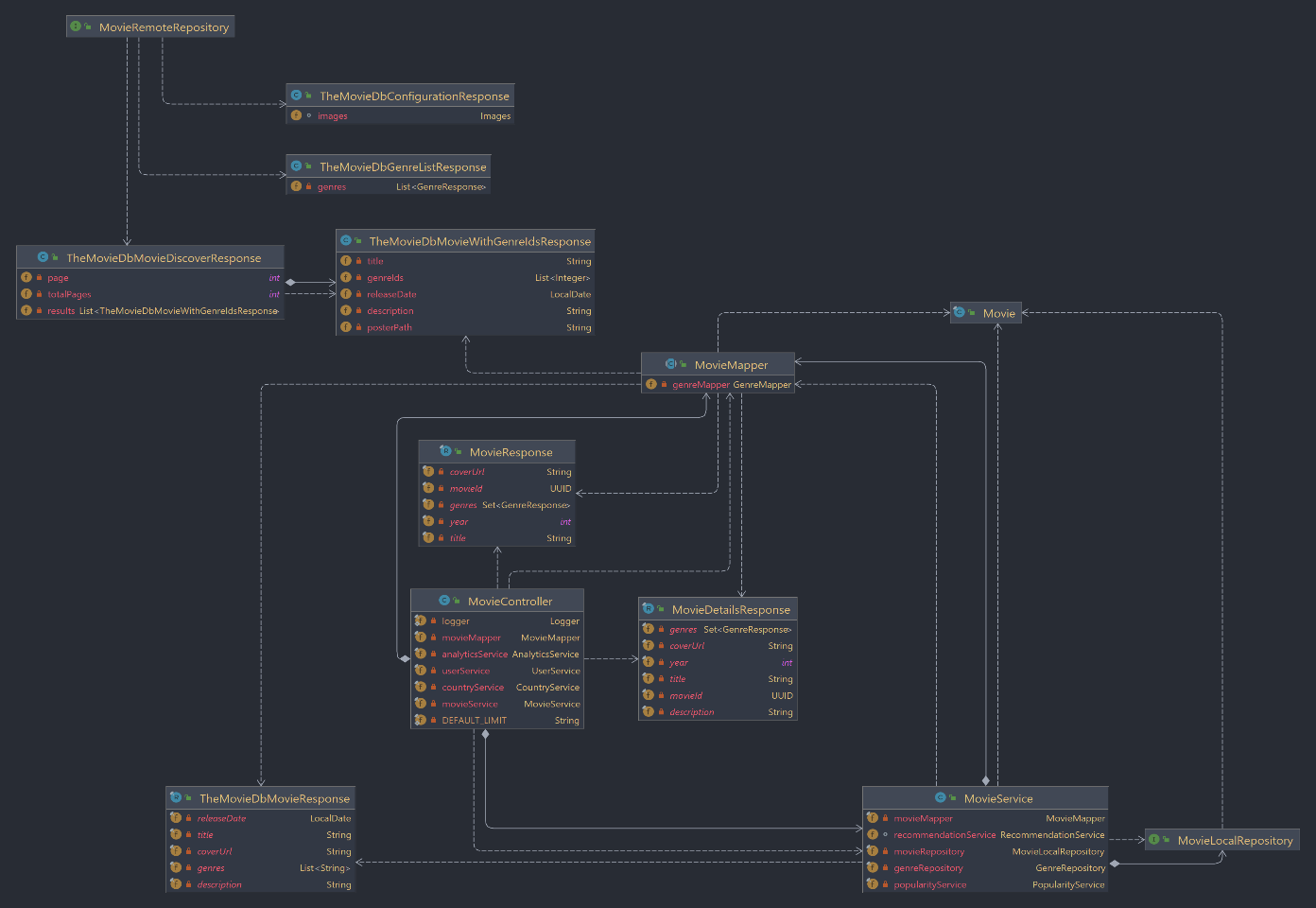
1. ID,
2. auth\_token
3. Email
4. Fullname

Detalji te tablice i spajanje tih podataka s drugim tablicama opisano je u poglavlju baza podataka.

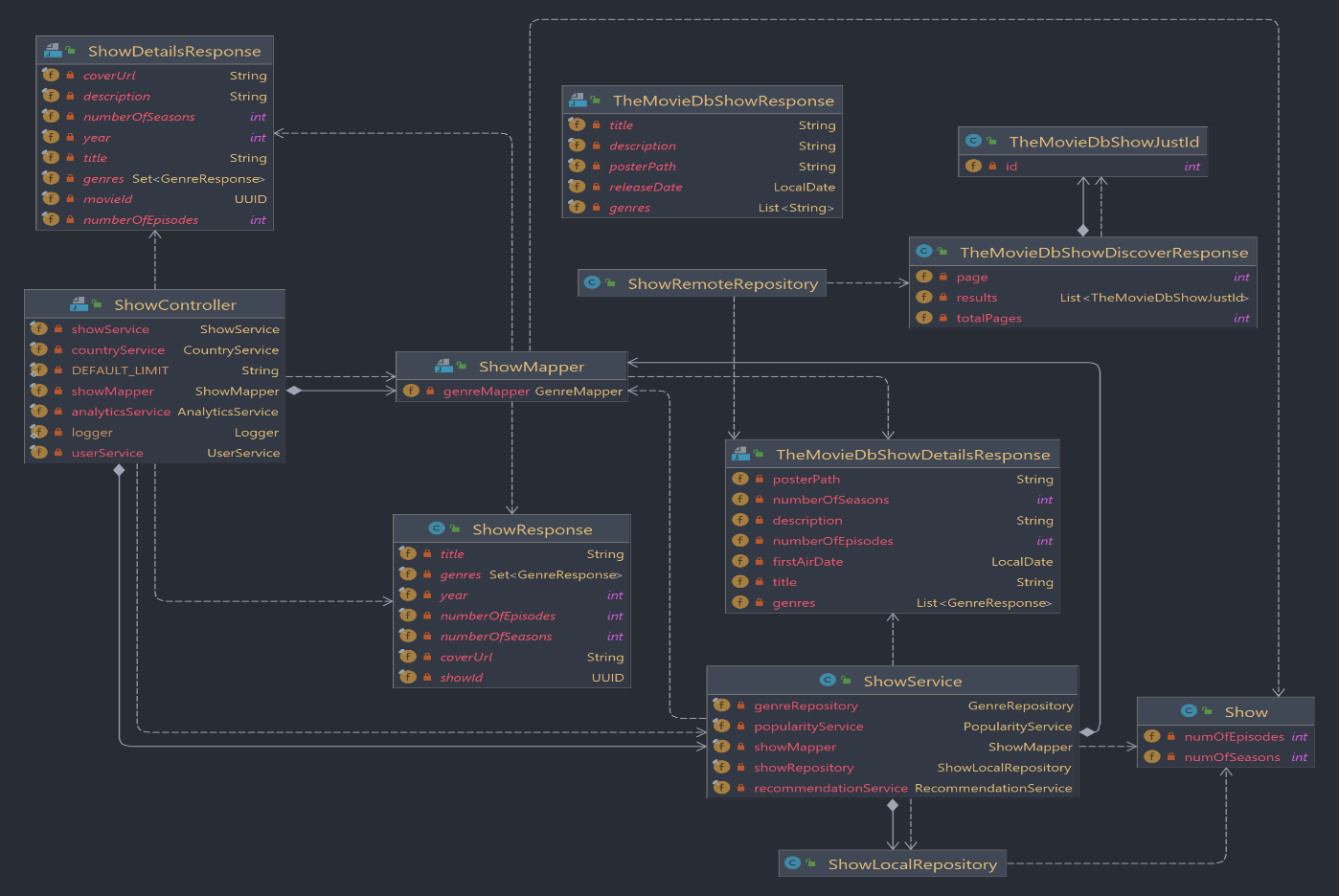
# Arhitektura sustava

Arhitektura sustava na backend dijelu aplikacije, podijeljena je na komponente za obradu korisnika, filmova, serija, popularnosti pojedinog sadržaja te analitike. Komponente za korisnika, film i serije modelirani su standardnim obrascem model, controller, service i repository. U model razredima, definiraju se atributi pridruženi pojedinoj komponenti koji se spremaju u bazu podataka. Razredima controller izrađuje se REST API koji može komunicirati s frontend dijelom aplikacije. U njima je definiran dohvat podataka iz service sloja te su implementirani end-pointovi aplikacije preko kojih se mogu u konačnici dohvatiti i izmijeniti podaci. U service sloju definirana je poslovna logika kojom se provjeravaju određeni parametri te se podaci nakon provjera prosljeđuju repository sloju. U repository sloju komponenata, definirane su metode za pristup bazi podataka te se u nju ti podaci pohranjuju. U nastavku ovog poglavlja bit će prikazane slike dijagrama razreda u kojima su definirane najvažnije komponente aplikacije.

Glavna komponenta u aplikaciji je *movie*, čiji se prikaz razreda može vidjeti u nastavku.

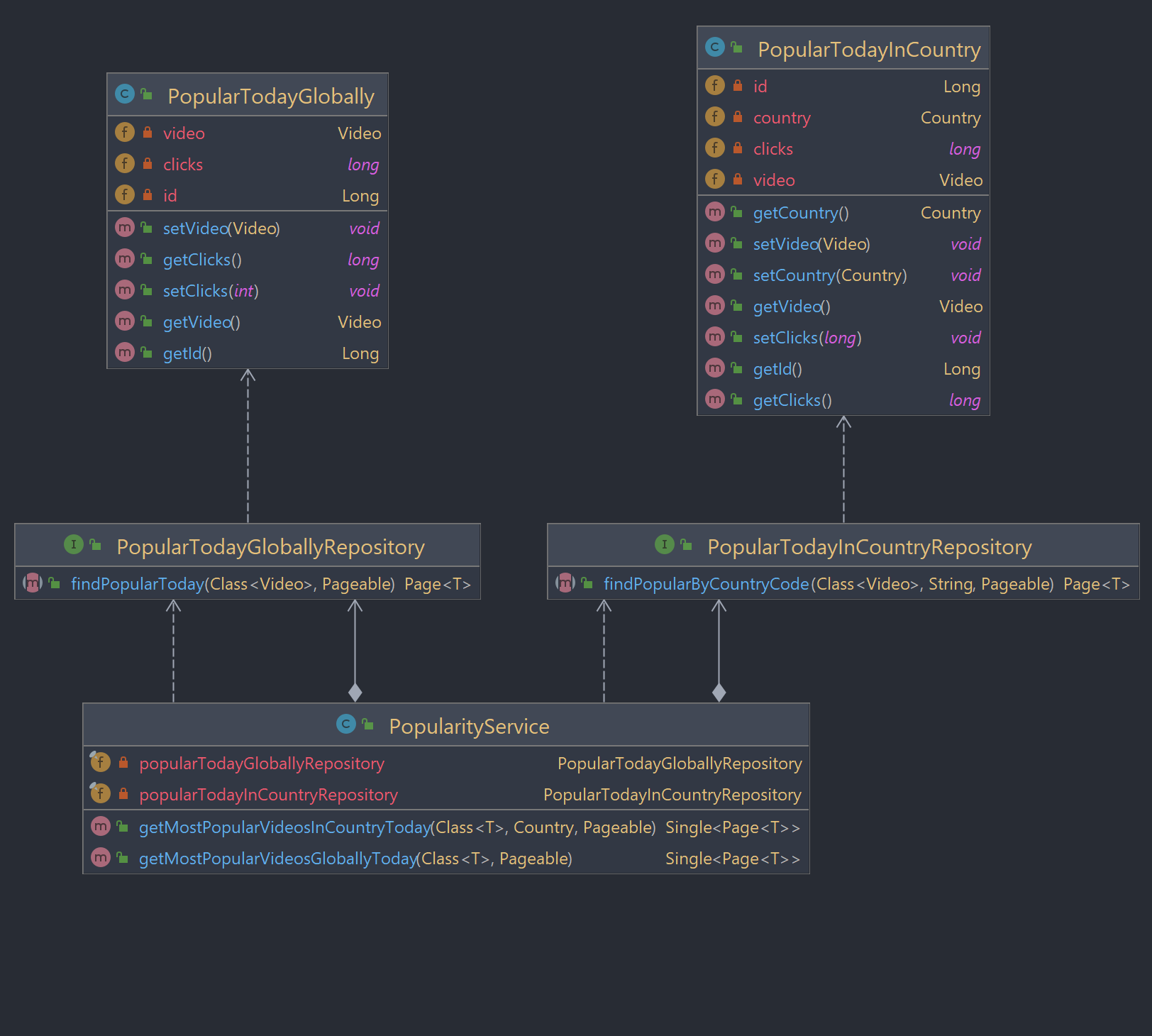
Na slici se može vidjeti tok podataka te spajanje razreda po slojevima kako je prethodno opisano. Uz tri glavna sloja, ova komponenta se također služi i pomoćnim razredima za dohvat podataka s udaljenog i lokalnog sjedišta.

Komponenta *show*, zadužena je za modeliranje i obradu podataka o serijama i spremanje istih u bazu podataka. Na slici u nastavku može se vidjeti prikaz njezinih razreda.



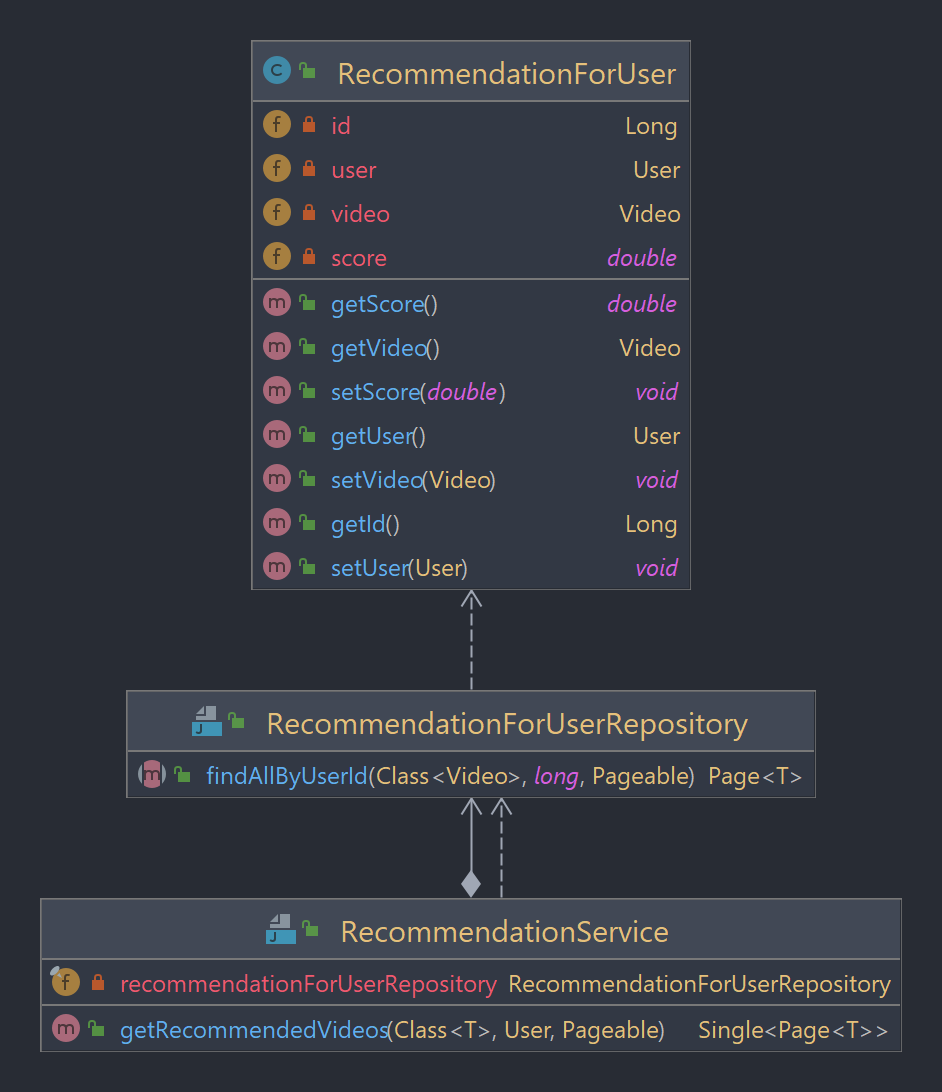
Na slici se mogu vidjeti svi razredi koji su korišteni u implementaciji show komponente. Komponenta se modelirala kao ekstenzija na model video te se u njoj obrađuju podaci slično kao i u movie modelu no uz dodatne dijelove za dohvat pojedinih podataka prilagodno za prikaz tih podataka na frontend dijelu aplikacije.

U komponenti *popular*, implementirani su modeli popularnih videa podijeljeni po državama i globalno. Na slici u nastavku može se vidjeti prikaz njezinih razreda.



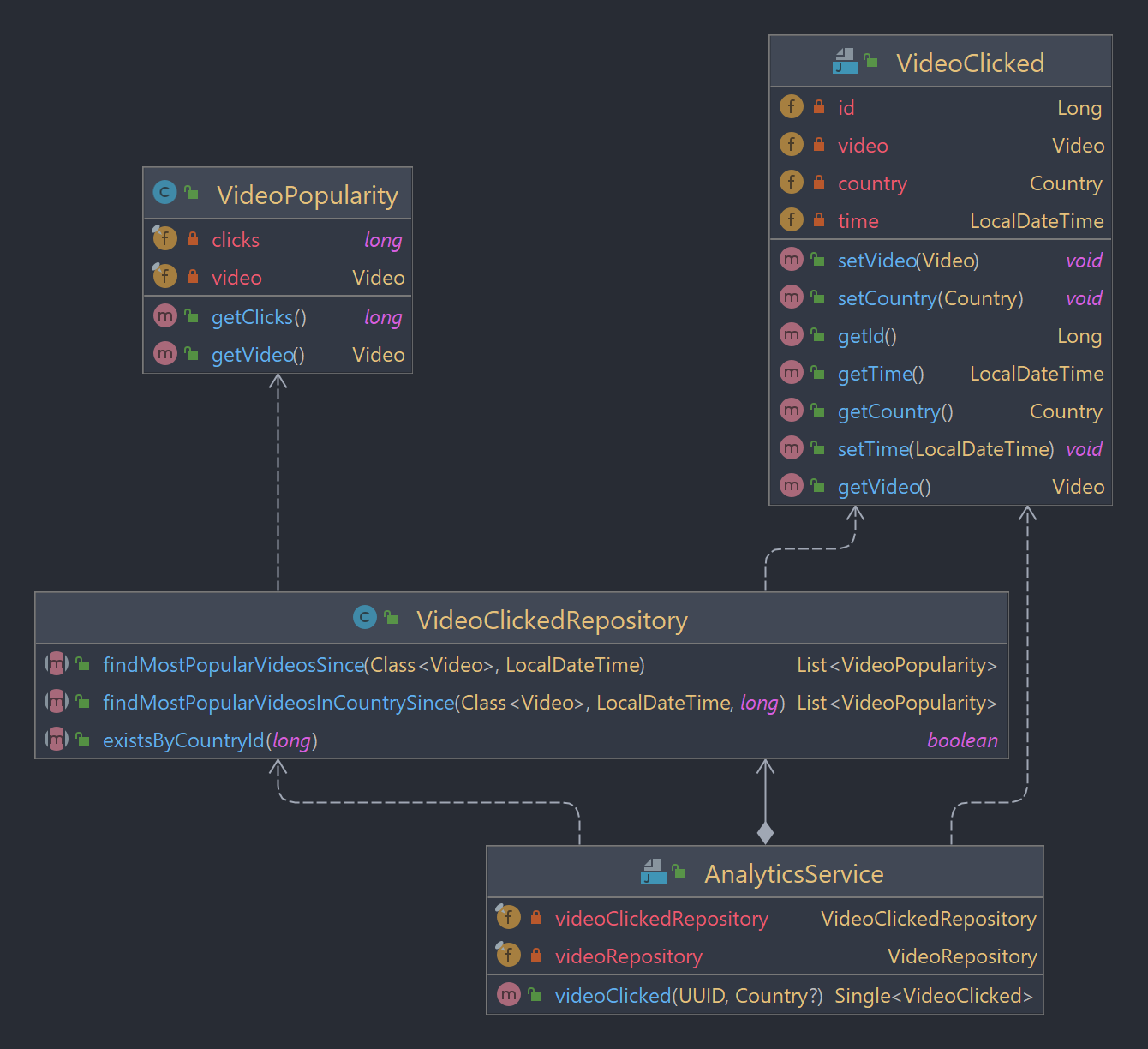
Na dijagramu se mogu vidjeti modeli videa koju su trenutno popularni i koji razredi komuniciraju s bazom podataka preko repository sloja aplikacije. Ovu komponentu, druge komponente mogu iskoristi kako bi saznali koji video modeli su trenutno popularni.

Komponentom *recommendation* modelira se preporuka korisniku o videu za koji se vjeruje da bi mogao biti prilagodna pojedinom korisniku. Slika dijagrama razreda slijedi u nastavku.



Za metodu izračuna preporuke koristi se Collaborative Filtering algoritam koji na osnovi korisnikove ocjene, daje preporuku o videu. Tu povezanost razreda vidi se na slici i kako service razred dohvaća preporuku videa.

Komponenta *analytics* bavi se analizom broja klikova za pojedini video te komunicira s popularnosti videa. Slika dijagrama razreda može se vidjeti u nastavku.



Na slici se mogu vidjeti svi razredi koji međusobno komuniciraju. U repository sloju, dohvaćaju se podaci iz popularnosti videa i prosljeđuju servisu za analitiku.

# Upute za korištenje razvijene aplikacije s opisima dijelova apliakcije

A group of people wearing 3d glasses

Description automatically generated with low confidence

Ovo je prikaz prve stranice aplikacije. Ovdje se od korisnika traži prijava preko Facebook servisa kako bi mogao nastaviti koristiti aplikaciju MovieScape.

A screenshot of a computer

Description automatically generated with medium confidence

Aplikacija potom otvara novi korisnički prozor u koje potvrđuje prijavu pokazivanjem podataka o korisniku i renderiranjem tablice koja sadržava najčešće zadržane žanrove. Korisnik odabire žanrove i pritišće gumb za dalje.

Graphical user interface

Description automatically generated

Nakon odabranih žanrova, gumbi mijenjaju oblik i boju, kako bi se korisnik uvjerio da su žanrovi odabrani.

A screenshot of a computer

Description automatically generated with low confidence

Aplikacija potom korisnika preusmjerava na popise filmova i serija, gdje se prvenstveno prikazuju top 10 filmova po popularnosti, po popularnosti na njegovom govornom području i top 10 filmova kuriranih za njega.

A screenshot of a computer

Description automatically generated with medium confidence

Ako korisnik odabere gumb serije, dobit će tematikom iste liste, samo vezano za serije.

Graphical user interface, text, application

Description automatically generated

Graphical user interface, website

Description automatically generated

Pritiskom na bilo koji film ili seriju, korisnik može vidjeti pojedinosti odabranog sadržaja, kao i ocjenu koju je dao ili će tek dati.

Ovako izgleda lista zvjezdica ako korisnik ocjeni otvoreni sadržaj, a ispod njegove ocjene je prosječna ocjena dana od naših korisnika.

# Programska logika, implementacija, opis korištenih tehnologija (front, back)

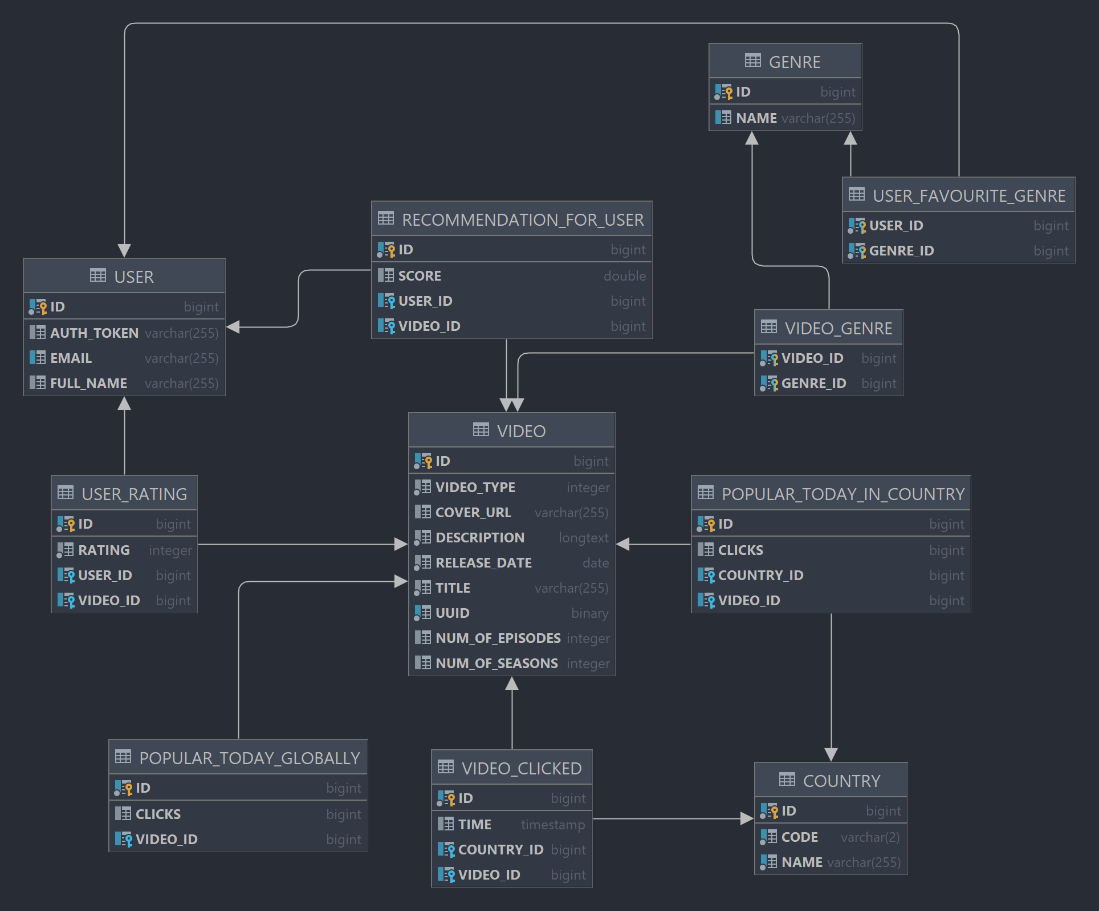
Za frontend se koristio framework React. U njemu se stvorilo po nekoliko različitih komponenti koje su prikazivale filmove, kartice o filmovima, karticu o korisniku, i komponentu u kojoj se biralo žanrove. Komponente za prikaz filmova i serija su generirane temeljem podataka iz baze, dok je komponenta za odabir žanrova hard kodirana. Uz to, koristili su se razni Node.js packages kako bi se moglo poraditi na izgledu same aplikacije, te uspješno oblikovati neke funkcionalnosti. Još jedna bitna komponenta je bio JavaScript library Axios koji se koristio za HTTP komunikaciju između aplikacije, backend-a i raznih externih API-ja.

Za backend se koristio framework Java Spring uz Java verziju 17. Uz to, koristi se i knjižnica RxJava3. Kreirani su modeli za korisnika (User), postavke za networking te video model u kojem je implementiran API za dohvaćanje filmova i serija. Preko tog API-ja, frontend aplikacija dohvaća potrebne filmove, serije i korisnike. U komponenti user, korisnikovi podaci se spremaju u bazu podataka, dok se u komponentama movie i show, dohvaćaju pojedini popularni filmovi i serije bazirani po kodu države.

# Baze podataka

Baza podataka je implementirana kao SQL baza podataka i sastoji se od sljedećih tablica:

1. User, User\_rating, recommendation\_for\_user, user\_rating, genre, video, video\_genre, user\_favourite\_genre, popularity\_totay\_in\_country, popular\_today\_globally, video\_clicked, country



Tablica User modelira korisnika aplikacije koji se prijavio u aplikaciju preko Facebook-a i preko njega prima jedinstveni token.

Tablica User\_rating modelira danu ocjenu korisnika za pojedinog korisnika.

Tablica genre modelira žanrove.

Tablica recommendation\_for\_user modelira pojedinu preporuku korisnika za određeni sadržaj.

Tablica user\_favourite\_genre modelira korisnikov najdraži žanr.

Tablica video\_genre služi za spajanje žanrova s pojedinim videom.

Tablica video modelira sve podatke o videu kao što su ime, opis, datum izlaska, broj epizoda i broj sezona ako se radi o seriji.

Tablica popularity\_today\_in\_country modelira broj pogleda videa u jednom danu za pojedinu državu.

Tablica country modelira ime države i njenu kraticu.

Tablica video\_clicked bilježi vrijeme klika na pojedini video.

Tablica popular\_today\_globally modelira popularnost pojedinog sadržaja na globalnoj razini.

# Upute za instalaciju i korištenje

Kako bi se aplikaciju uspješno koristilo, potrebno je dohvatiti kod iz repozitorija <https://github.com/Drustvene-projekt/Frontend>. Nakon što se kod dohvati potrebno ga je otvoriti u nekom code editoru(VSC npr.) i pokrenuti naredbu **npm install**(korisnik mora imati instalirano Node.js) kako bi se uspješno skinuli svi potrebni moduli za rad aplikacije. Jednom kad je to gotovo aplikacija se pokreće naredbom **npm start** na URL-u <http://localhost:3000> i korisnik može pristupiti početnoj stranici aplikacije. Kako bi se aplikacija uspješno koristila korisnik mora imati Facebook račun. Nakon što se korisnik ulogira sa Facebook računom, dobiva pristup glavnoj stranici aplikacije, i tamo može proučiti informacije o raznim filmovima i serijama, te ima mogućnost davanja ocijene pojedinom sadržaju.

Za potrebe pokretanja backend dijela aplikacije, izvršava se naredba:

nohup java -jar backend.jar 2>&1 >output.log &

Te za generiranje jar datoteke: mvn install

# Mogućnost unaprijeđena i komercijalizacije razvijene aplikacije

Sljedeća etapa u razvoju aplikacije bi bila mogućnost dodavanja prijatelja korisnika kako bi nastala mreža zajedničkih interesa sličnih korisnika. Osim toga, moglo bi se dodati neke sitnice, kao na primjer poveznice na postojeće streaming servise kako bi korisnik odmah mogao pogledati odabrani sadržaj.

Komercijalizacija bi se vjerojatno izvodila ili kroz mjesečno plaćanje članarine ili ograničen pristup mogućnostima uz dodavanje reklama.

Prednosti ove aplikacije u odnosu na neka postojeća rješenja su primjerice ocjene u odnosu na Netflix, koji podržava samo pozitivnu poruku ili pak prepoznavanje korisnika i preporuka sadržaja za njega u odnosu na IMDB.