

# Sustav za preporuku filmova

---

**Marković, Robert**

**Master's thesis / Diplomski rad**

**2024**

*Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj:* **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

*Permanent link / Trajna poveznica:* <https://um.nsk.hr/um:nbn:hr:200:972726>

*Rights / Prava:* [In copyright](#) / [Zaštićeno autorskim pravom.](#)

*Download date / Datum preuzimanja:* **2025-03-06**

*Repository / Repozitorij:*

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU**

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I  
INFORMACIJSKIH TEHNOLOGIJA OSIJEK**

**Sveučilišni diplomski studij Računarstvo**

**Sustav za preporuku filmova**

**Diplomski rad**

**Robert Marković**

**Osijek, 2024.g.**

**FERIT**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMATIJSKIH TEHNOLOGIJA OSIJEK**Obrazac D1: Obrazac za ocjenu diplomskog rada na sveučilišnom diplomskom studiju****Ocjena diplomskog rada na sveučilišnom diplomskom studiju**

<b>Ime i prezime pristupnika:</b>	Robert Marković
<b>Studij, smjer:</b>	Sveučilišni diplomski studij Računarstvo
<b>Mat. br. pristupnika, god.</b>	D-1228R, 08.10.2021.
<b>JMBAG:</b>	0165075187
<b>Mentor:</b>	prof. dr. sc. Josip Job
<b>Sumentor:</b>	
<b>Sumentor iz tvrtke:</b>	
<b>Predsjednik Povjerenstva:</b>	izv. prof. dr. sc. Časlav Livada
<b>Član Povjerenstva 1:</b>	prof. dr. sc. Josip Job
<b>Član Povjerenstva 2:</b>	izv. prof. dr. sc. Ivica Lukić
<b>Naslov diplomskog rada:</b>	Sustav za preporuku filmova
<b>Znanstvena grana diplomskog rada:</b>	<b>Programsko inženjerstvo (zn. polje računarstvo)</b>
<b>Zadatak diplomskog rada:</b>	Kroz teorijski dio rada potrebno je proučiti postojeće sustave za preporuku filmova te napraviti njihovu usporedbu. U okviru praktičnog dijela rada potrebno je izraditi aplikaciju za iOS platformu koja bi korisnicima omogućila ocjenjivanje filmova te predlaganje filmova od interesa na osnovu ocjena korisnika aplikacije kao i ocjena ostalih korisnika sustava. Izrađeni sustav potrebno je ispitati u praksi te napraviti analizu prikupljenih podataka. Tema rezervirana za: Robert Marković
<b>Datum ocjene pismenog dijela diplomskog rada od strane mentora:</b>	23.09.2024.
<b>Ocjena pismenog dijela diplomskog rada od strane mentora:</b>	Dobar (3)
<b>Datum obrane diplomskog rada:</b>	10.10.2024.
<b>Ocjena usmenog dijela diplomskog rada (obrane):</b>	Vrlo dobar (4)
<b>Ukupna ocjena diplomskog rada:</b>	Vrlo dobar (4)
<b>Datum potvrde mentora o predaji konačne verzije diplomskog rada čime je pristupnik završio sveučilišni diplomski studij:</b>	10.10.2024.



**FERIT**

FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA  
I INFORMACIJSKIH TEHNOLOGIJA **OSIJEK**

## IZJAVA O IZVORNOSTI RADA

Osijek, 10.10.2024.

**Ime i prezime Pristupnika:**

Robert Marković

**Studij:**

Sveučilišni diplomski studij Računarstvo

**Mat. br. Pristupnika, godina upisa:**

D-1228R, 08.10.2021.

**Turnitin podudaranje [%]:**

3

Ovom izjavom izjavljujem da je rad pod nazivom: **Sustav za preporuku filmova**

izrađen pod vodstvom mentora prof. dr. sc. Josip Job

i sumentora

moj vlastiti rad i prema mom najboljem znanju ne sadrži prethodno objavljene ili neobjavljene pisane materijale drugih osoba, osim onih koji su izričito priznati navođenjem literature i drugih izvora informacija.

Izjavljujem da je intelektualni sadržaj navedenog rada proizvod mog vlastitog rada, osim u onom dijelu za koji mi je bila potrebna pomoć mentora, sumentora i drugih osoba, a što je izričito navedeno u radu.

Potpis pristupnika:

# Sadržaj

1. UVOD .....	1
1.1. Zadatak diplomskog rada .....	1
2. POSTOJEĆA RJEŠENJA .....	2
2.1. Algoritamski sustavi .....	2
2.2. Napredniji sustavi za preporuku .....	3
2.3. Ostali sustavi .....	4
3. IZRADA SUSTAVA ZA PREPORUKU FILMOVA .....	6
3.1. Razvojno okruženje .....	6
3.2. Arhitektura iOS aplikacije .....	8
3.3. Dobivanje podataka o filmovima .....	11
3.4. Combine .....	13
3.5. Obrada podataka .....	14
3.6. Generiranje filmskih kartica .....	15
3.7. Odabiranje preferiranih filmova .....	15
3.8. Filtriranje .....	15
4. TESTIRANJE .....	17
4.1. Korisnici su ocijenili .....	26
5. ZAKLJUČAK .....	27
LITERATURA .....	28
SAŽETAK .....	29
ABSTRACT .....	30
ŽIVOTOPIS .....	31
PRILOZI .....	32

# 1. UVOD

Gledanje filmova spada u jedan od omiljenih načina zabave mnogih ljudi radilo se o samom odlasku u kino ili gledanju željenog filma iz udobnosti vlastitog doma. Jedino uvijek postavljeno pitanje jest koji film gledati i tu u modernije vrijeme do važnosti dolaze razni sustavi za preporuku filmova. Sustavi za preporuku filmova su većinom dostupni na internetu u obliku web stranice gdje korisnik davajući osobne kriterije kao rezultat dobije određene filmske naslove. Također nerijetko na istom mjestu ima i mogućnost saznati dodatne podatke vezane za odabrani film. Razvojem programiranja mobilnih platformi razvile su se mobilne aplikacije koje korisnicima mogu predložiti film prema traženim kriterijima. U ovom radu opisat će se izrada jedne takve mobilne aplikacije koja će imati ugrađen algoritam preporuke filma, a za realizaciju koristit će se iOS mobilni sustav. Osmišljeno ime opisane aplikacije jest „Cineverse“. A naziv je kreativno osmišljen spajanjem dvije engleske riječi „cinema“ (hrv. kino) i „universe“ (hrv. svemir) – Cinevese.

Cineverse mobilna aplikacija ima zadatak prikazati listu filmova u obliku vertikalnih kartica. Lista se dohvaća u JSON formatu s „The Movie Database“ API-a[1]. Svaka kartica sadržava filmski promotivni plakat, žanr, datum objavljivanja, te kratki opis radnje filma. Ono po čemu je aplikacija zanimljiva jest što kartice nisu prikazane kao retci u tablici koja ima mogućnost pomicanja gore-dolje. Već su kartice generirane i postavljene na stog, tako da je u trenutku vidljiva isključivo jedna, ona najgornja, filmska kartica s njezinim pripadajućim podacima. Takva kartica ima isprogramiranu pomičnu gestu sa sposobnošću pomicanja lijevo-desno i gore-dolje ne mijenjajući oblik i sadržaj. Nađe li se centar kartice blizu lijevog ruba zaslona iPhone-a ili iPad-a i u tom trenutku korisnik otpusti karticu, kartica će biti obrisana. A ako korisnik otpusti karticu blizu desnog ruba zaslona, žanr kartice će biti označena filteru kao željena preferenca pa će aplikacija na temelju žanra tog filma tražiti filmove s istim žanrovima. Mobilna aplikacija pisana je u Swift objektno-orijentiranom programskom jeziku, a radni okvir u kojem je pisana jest UIKit i Combine za asinkrono izvršavanje zadataka. U poglavlju dva opisana su postojeća rješenja sustava za preporuku, te se u nastavku još nalazi tijekom izrade navedenog sustava, programiranje iOS aplikacije, završno testiranje i zaključak.

## 1.1. Zadatak diplomskog rada

Kroz teorijski dio rada potrebno je proučiti postojeće sustave za preporuku filmova te napraviti njihovu usporedbu. U okviru praktičnog dijela rada potrebno je izraditi aplikaciju za iOS platformu koja bi korisnicima omogućila ocjenjivanje filmova te predlaganje filmova od interesa na osnovu ocjena korisnika aplikacije kao i ocjena ostalih korisnika sustava. Izrađeni sustav potrebno je ispitati u praksi te napraviti analizu prikupljenih podataka.

## 2. POSTOJEĆA RJEŠENJA

Sustavi za preporuku filmova prisutni su već dugi niz godina i postali su neizostavan dio svakodnevnog života velikog broja korisnika. Ne samo da su popularni, već ih susrećemo gotovo svuda — od streaming servisa poput Netflix[2], do internetskih platformi za gledanje filmova (poput Filmoviplex[3] platforme), recenzija i preporuka. Osim sustava za preporuku filmova, postoje brojni drugi sustavi za preporuku koji djeluju u različitim područjima. Na primjer, postoje sustavi za preporuku knjiga (poput onih koje koristi Amazon[4]), partnera (kao što su Tinder[5], Badoo[6] i druge aplikacije za upoznavanje), glazbe (npr. Spotify[7] i Apple Music[8]), putovanja (Air Bnb[9]) i oglasa (npr. Njuškalo oglasnika[10] i Facebook Ads[11]). Svi ovi sustavi pomažu korisnicima u odabiru sadržaja koji najbolje odgovaraju njihovim interesima i potrebama. Sustavi za preporuku, bez obzira na njihovu primjenu koriste različite tehnike i algoritme za generiranje preporuka. Ovisno o vrsti podataka s kojima rade i specifičnostima aplikacije, koriste se različiti pristupi. Najpopularnije metode uključuju kolektivno filtriranje, koje koristi povijest ponašanja korisnika za predviđanje budućih preferencija; filtraciju temeljenu na sadržaju, koja analizira kontekst samog sadržaja (kao što su žanr ili opis filma); te hibridne sustave, koji kombiniraju ove dvije metode kako bi se dobili što precizniji rezultati. Sustavi za preporuke nisu ograničeni na jedan sektor — oni su univerzalni i mogu se prilagoditi različitim vrstama podataka i korisničkim odabirima. Na primjer, u sustavima za preporuku glazbe, algoritmi često analiziraju slušane pjesme i izvođače te predlažu slične pjesme ili albume. S druge strane, sustavi za preporuku putovanja uzimaju u obzir ocjene hotela, destinacija, pa čak i osobne preferencije korisnika, kao što su budžet i stil putovanja. Kako bi sustavi za preporuku bili što učinkovitiji, koristi se širok spektar algoritama i tehnika. U sljedećem poglavlju bit će opisane neke od najčešće korištenih tehnika koje čine temelj današnjih sustava za preporuku.

### 2.1. Algoritamski sustavi

Algoritamski sustavi su često jednostavniji za implementaciju jer se temelje na unaprijed definiranim pravilima. Preporuke se nude na temelju određenih kriterija kao što su cijena, popularnost, ime glumca ili žanr filma, a sve se svodi na filtriranje podataka prema jasno definiranim parametrima. Iako ovi sustavi mogu brzo ponuditi preporuke njihova statičnost može dovesti do manje relevantnih prijedloga za korisnika. Na primjer, ako korisnik preferira gledati drame s manje poznatim glumcima, sustav koji preporučuje samo popularne filmove s poznatim imenima neće zadovoljiti njegove potrebe. Ovi sustavi se često primjenjuju u e-trgovini kao što je slučaj s web oglasnikom poput Njuškala, gdje korisnik može filtrirati oglašene artikle prema cijeni, relevantnosti ili datumu objave. Takav pristup omogućuje korisnicima brzu pretragu, ali nije nužno personaliziran, što može rezultirati manje relevantnim preporukama. U filmskim aplikacijama poput Cineverse iOS aplikacije sustav za preporuku filmova također koristi ovu tehniku kako bi korisnicima omogućio da pregledavaju filmove na temelju specifičnih parametara - žanra. Iako je ova metoda korisna za jednostavne pretrage, ona ne uzima u obzir složenije korisničke preferencije, poput povijesti gledanja, emocionalnih sklonosti ili trenutnog raspoloženja. Međutim, upravo zbog te jednostavnosti ovi sustavi su popularni za osnovne aplikacije i one u kojima brzina i jasnoća pretrage imaju prednost nad točnošću preporuka. Kao rezultat toga često se koriste s naprednijim metodama poput strojnog učenja ili kolektivne

filtracije kako bi se korisnicima omogućilo bolje korisničko iskustvo i relevantniji prijedlozi. Na taj način jednostavni sustavi temelji su na kojima se nadograđuju napredniji algoritmi čime se postiže ravnoteža između brzine pretrage i personalizacije preporuka.

## **2.2. Napredniji sustavi za preporuku**

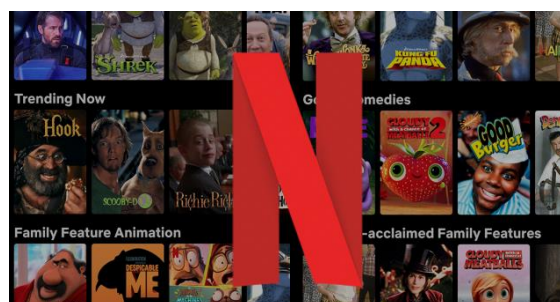
Osim jednostavnih algoritamskih sustava postoje kompleksniji sustavi za preporuku koji koriste modele strojnog učenja. Takvi sustavi omogućuju personaliziranije i točnije preporuke jer analiziraju velike količine podataka i koriste napredne tehnike poput kolektivnog filtriranja, neuronskih mreža i dubokog učenja. Ovi sustavi nazivaju se kolektivni sustavi za preporuku jer koriste podatke od korisnika kako bi generirali preporuke na temelju sličnosti među korisnicima ili stavkama koje korisnici ocjenjuju. Za preporuke su najzastupljenija dva pristupa: item-based[12] (preporuke temeljene na stavkama) i user-based[13] (preporuke temeljene na korisnicima). Item-based kolektivni sustavi fokusiraju se na sličnost među stavkama. Na primjer, ako je korisnik pozitivno ocijenio određeni film sustav će analizirati karakteristike tog filma (žanr, redatelj, ocjene drugih korisnika itd.) i predložiti filmove slične karakteristikama. Ovaj pristup se pokazao vrlo uspješnim u industriji e-trgovine gdje Amazon koristi item-based filtriranje kako bi preporučio različite proizvode. Sustav analizira povijest pretraga, ocjena, broja kupovina ili pregleda od strane korisnika te na temelju toga predlaže slične proizvode koje bi korisnik mogao poželjeti kupiti. Nešto drugačiji pristup predstavljaju user-based kolaborativni sustavi koji se fokusiraju na sličnost između korisnika. Ako sustav prepozna da korisnik A ima slične preferencije kao korisnik B, on će korisniku A preporučiti filmove koje je korisnik B ocijenio pozitivno, a korisnik A ih još nije vidio. Ova metoda omogućuje personalizirane preporuke za korisnike sa specifičnim i zahtjevnim interesima. Na primjer, Netflix koristi user-based kolektivnu filtraciju kako bi preporučivao filmove i serije. Ako korisnici imaju slične obrasce gledanja, Netflix će preporučiti sadržaj temeljen na preferencijama korisnika sličnih interesa. Jedna od ključnih prednosti ovih sustava je njihova vještina da se prilagode promjenjivim preferencijama korisnika. Korištenjem modela strojnog učenja sustavi uče iz podataka koji dolaze u stvarnom vremenu i prilagođavaju svoje preporuke na temelju novih informacija. Na primjer, ako korisnik promijeni svoje navike gledanja i počne više gledati dokumentarce umjesto akcijskih filmova, sustav će brzo prepoznati ovaj obrazac i prilagoditi svoje preporuke. Napredniji sustavi za preporuke također koriste duboko učenje i neuralne mreže za još preciznije predviđanje preferencija korisnika. Ovi modeli mogu analizirati kompleksnije obrasce ponašanja kao što su sekvencijalni odnosi između stavki koje korisnici pregledavaju kako bi pružili još personaliziranije preporuke. Na primjer, sustavi za preporuku na temelju rekurentnih neuronskih mreža mogu pratiti redoslijed u kojem korisnici gledaju filmove i na temelju toga preporučiti sljedeći film koji bi ih mogao zanimati. Osim toga, moderni sustavi koriste grafičke modele koji povezuju korisnike i stavke kroz složene mreže odnosa omogućujući dublje razumijevanje korisničkih preferencija i pružanje preciznijih preporuka. Ovi napredni sustavi omogućuju veću točnost i relevantnost preporuka što ih čini izuzetno korisnima u aplikacijama poput Netflix, Amazona, Spotifyja i drugih platformi koje se oslanjaju na personalizirano korisničko iskustvo.



### 2.3. Ostali sustavi

Još jedan od najpopularnijih pristupa su hibridni sustavi za preporuku koji koriste više tehnika istovremeno kako bi pružili što točnije i relevantnije preporuke. Umjesto da se oslanjaju samo na jednu metodu, hibridni sustavi koriste kombinaciju različitih pristupa poput tehnika temeljenih na sadržaju i kolaborativnog filtriranja čime se smanjuju ograničenja svake od tih metoda. Na primjer, dok sustavi temeljeni na sadržaju analiziraju karakteristike samih filmova (žanr, redatelj, glumci), kolaborativni sustavi se oslanjaju na podatke o ponašanju korisnika. Kombinacijom tih tehnika sustavi mogu pružiti točnije preporuke. Netflix je dobar primjer hibridnog sustava jer koristi oba pristupa koristeći podatke o gledanju sadržaja, ocjenama, preferencijama i ponašanju korisnika kako bi stvorio personalizirane preporuke za svakog korisnika. Sustavi temeljeni na kontekstu su još jedan tip naprednijih sustava za preporuku koji uzimaju u obzir specifične kontekstualne informacije o korisniku. Ovi sustavi koriste podatke kao što su trenutna lokacija korisnika, doba dana, raspoloženje, uređaj koji korisnik koristi ili čak vrijeme. Na primjer, sustav može predložiti filmove s opuštenijom tematikom za večernje gledanje ili komediju kada je korisnik opušten, dok bi ujutro preporučio dokumentarce ili ozbiljnije sadržaje. Google Assistant[14] i Siri[15] su primjeri sustava koji se sve više oslanjaju na kontekst kako bi prilagodili preporuke, iako su još uvijek u fazi razvoja u kontekstu preporuka filmova. Još jedan zanimljiv pristup su sustavi bazirani na skupinama (grupno filtriranje[16]), koji koriste kolektivno znanje i preferencije korisnika kako bi pružili preporuke. Ovi sustavi preporučuju sadržaje na temelju popularnosti među širokom bazom korisnika. Filtriranje prema popularnosti koristi podatke o tome koji su filmovi najviše ocijenjeni ili najviše gledani, uzimajući u obzir „kolektivni ukus” publike. Na primjer, filmski servisi poput IMDB-a koriste ovaj pristup kako bi kreirali popise najpopularnijih filmova, dok YouTube preporučuje videe koji su trenutno viralni među velikim brojem korisnika. Iako je ova metoda korisna za otkrivanje sadržaja koji je široko prihvaćen, može zanemariti specifične interese pojedinaca te često preporučuje najpopularniji, a ne nužno najrelevantniji sadržaj za svakog korisnika. Uz navedene primjere, postoje i napredniji pristupi poput sustava temeljenih na dubokim neuronskim mrežama, koji mogu analizirati složenije obrasce ponašanja i preferencija korisnika, integrirajući podatke iz različitih izvora (npr. ponašanje na društvenim mrežama, povijest pretraga, ocjene filmova). Ovi sustavi imaju potencijal pružiti još preciznije i dinamičnije preporuke, prilagođavajući se specifičnim korisničkim potrebama u realnom vremenu.

U nastavku se nalazi tablica 2.1 u kojoj su prikazane ključne razlike između algoritamskih, naprednih i hibridnih sustava za preporuku filmova, s naglaskom na njihove tehnologije, prednosti i primjere primjene.



Slika 2.1. Naslovna stranica Netflixove filмотеке.

**Tablica 2.1.** Pregled i usporedba različitih sustava za preporuku.

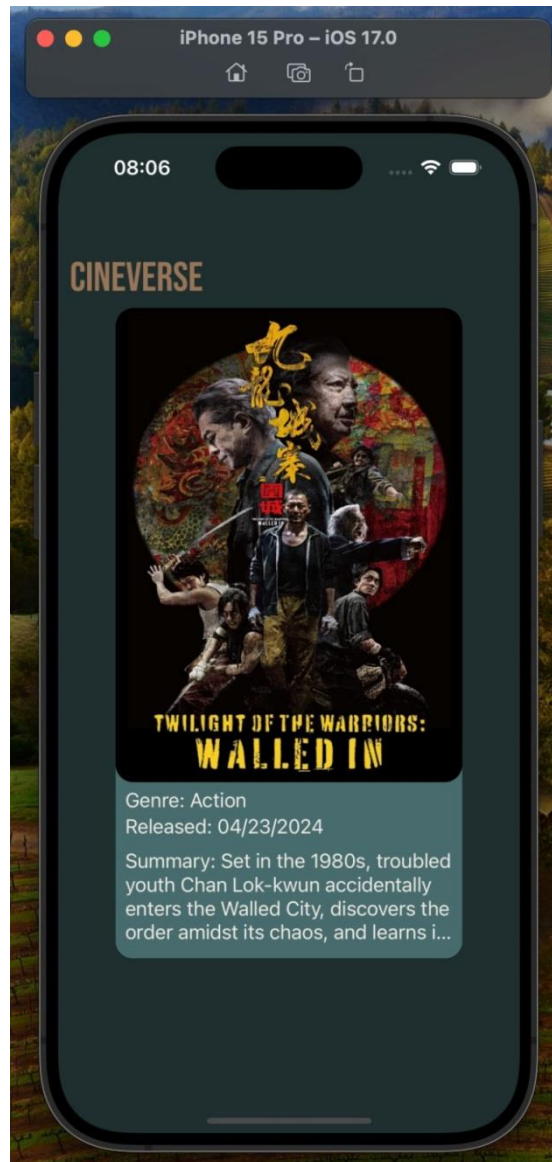
<i>Vrste sustava</i>	<i>Algoritam</i>	<i>Prednosti</i>	<i>Nedostatci</i>	<i>Primjeri</i>
<i>Algoritamski sustavi</i>	Pravila filtriranja (npr. žanr, cijena, popularnost)	Jednostavni za implementaciju, brzi rezultati	Ograničena fleksibilnost, manje precizne preporuke	Njuškalo oglasnik, IMDb
<i>Napredniji sustavi</i>	Strojno učenje, kolektivno filtriranje	Personalizirane i točnije preporuke. bolje skaliranje s velikom količinom podataka	Potrebna veća količina podataka, kompleksnost implementacije	Netflix, Amazon
<i>Hibridni sustavi</i>	Kombinacija sadržajno-temeljenih i kolektivnih tehnika	Kombinira prednosti različitih metoda, smanjuje nedostatke pojedinih sustava	Veća složenost, zahtjeva više resursa za obradu	Netflix (koristi kombinaciju tehnika)
<i>Sustavi temeljeni na kontekstu</i>	Kontekstualni podaci (npr. vrijeme, lokacija, raspoloženje)	Prilagodljivost trenutnim uvjetima korisnika	Potreba za prikupljanjem dodatnih podataka, rizik za privatnost korisnika	Google Assistant, Siri
<i>Grupno filtriranje</i>	Kolektivne preferencije korisnika	Prikladno za zajedničke odluke, uzima u obzir grupne interese	Može zanemariti individualne preferencije unutar grupe	Happy Movie, Netflix Party

### 3. IZRADA SUSTAVA ZA PREPORUKU FILMOVA

Cineverse iOS aplikacija sadrži stog s filmskim karticama. Aplikacija generira navedene kartice i istovremeno vodi računa o broju kartica na zaslonu, također reagira na korisničko pomicanje kartica u svim smjerovima. Ukoliko kartica biva pomaknuta lijevo ili desno do određene granice na zaslonu vrši se filtriranje, a kasnije i filtriranje srodnih filmova po žanru.

#### 3.1. Razvojno okruženje

Xcode je službeno razvojno okruženje (IDE)[17] koje je razvila tvrtka Apple za stvaranje aplikacija za sve njihove iOS, macOS, watchOS i tvOS platforme. Xcode pruža sve nužne alate za razvoj, dizajniranje, testiranje aplikacija, te uklanjanje pogrešaka. Apple aplikacije pišu se u Swift i Objective-C programskim jezicima. Swift je moderniji jezik, dok se Objective-C i dalje koristi u postojećim ali starijim projektima. Postoje dva najkorištenija UI radna okvira; UIKit i SwiftUI. Cineverse je izrađen u UIKit radnom okviru. SwiftUI je noviji radni okvir i tek dobiva popularnost. SwiftUI aplikacije su primjer modernijeg pristupa iOS-u. UIKit je dan danas uveliko u upotrebi za izradu novih aplikacija i za održavanje ali i za nadogradnju već postojećih. Glavna razlika je kodiranje UI elemenata koje se sastoji od drugačijih UI komponenti. Korisničko sučelje može se razviti na dva načina. Prvi je način programatski način i on je korišten u Cineverse aplikaciji. Programatski način zahtjeva kodiranje korisničkog sučelja u programskom jeziku. Drugi način uključuje Xcode Interface Builder i popularno se još zove Storyboard koji služi kao alat za vizualno dizajniranje korisničkih sučelja. Developer koji postavlja elemente poput gumbova, tablica i ostalih pogleda na sučelje promjene može vidjeti u stvarnom vremenu. Za razliku od Storyboard-a, programatski način smatra se nešto ozbiljnijim pristupom iz razloga preciznijeg postavljanja AutoLayout ograničenja kao i mogućnost pristupa nešto većem broju svojstava svih UI pogleda. Nakon isprogramiranog korisničkog sučelja ono se može istestirati na Xcode-ovom ugrađenom simulatoru koji omogućuje testiranje aplikacija bez potrebe za fizičkim Apple uređajem. Simulator podržava različite uređaje, rezolucije ekrana i verzije operacijskog sustava, što pomaže u otkrivanju problema na različitim sustavima. Tko ne želi instalirati, pokrenuti i testirati aplikaciju na XCode simulatoru i ima iPhone mobilni uređaj, može instalirati aplikaciju na svom uređaju.



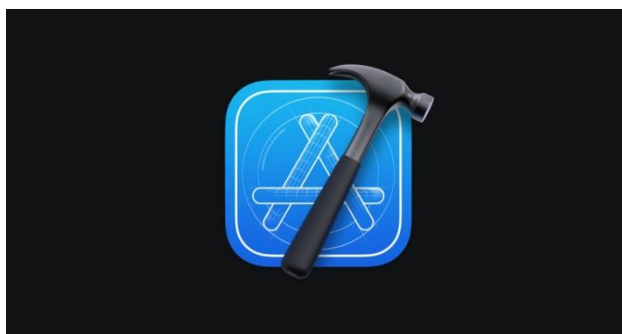
**Slika 3.1.** Cineverse iOS aplikacija pokrenuta na XCode simulatoru koji simulira iPhone 15 Pro.

Xcode nudi i detaljan prikaz performansi aplikacija, praćenje memorije, CPU korištenja i drugih resursa. Xcode također nudi izvrsne alate za otklanjanje pogrešaka (eng. debugging), što olakšava pronalaženje grešaka u kodu.



**Slika 3.2.** Praćenje performansi pokrenute aplikacije u XCode „Debug Navigator“ odjeljku.

Xcode omogućuje i postavljanje aplikacija putem App Store Connect platforme na App Store gdje korisnici mogu vidjeti i preuzeti aplikaciju.



**Slika 3.3.** Xcode logo.

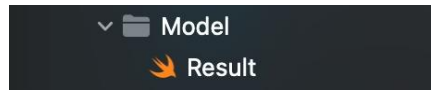
### 3.2. Arhitektura iOS aplikacije

Cineverse iOS aplikacija koristi arhitekturni obrazac MVVM što je skraćeno od Model-View-ViewModel[18]. MVVM jedan je od najpopularnijih i spada u nepočetničke obrasce u razvoju iOS aplikacija (poput MVC obrasca). Ono što MVVM nudi jest jasno odvajanje poslova i odgovornosti između različitih slojeva aplikacije, što omogućuje bolju organizaciju koda, jednostavnije testiranje i kasnije lakše održavanje i nadogradnju aplikacija. MVVM ima tri komponente: model, view i view-model. Na slici 3.4. prikazana je arhitektura MVVM iOS aplikacije, koja ilustrira odvajanje logike poslovnog modela od korisničkog sučelja, te način na koji se podaci prenose između modela, pogleda i view modela.



**Slika 3.4.** MVVM struktura aplikacije s pripadajućim komponentama i odnosima među njima.

Model je predstavljen strukturom koja sadrži osnovne podatke. On upravlja podacima aplikacije, bilo da ih dohvaća iz baze podataka, vanjskog API-ja ili ih računa. Model ne komunicira izravno s korisničkim sučeljem. U aplikaciji za preporuke filmova, model predstavlja datoteka „Model.swift“ i čini ju „Result“ struktura koja implementira attribute filma opcionalog tipa poput naziva filma, žanra, itd. Ujedno struktura predstavlja kodirani objekt dobiven s API-a.



**Slika 3.5.** „Result.swift“ datoteka koja sadrži implementaciju za Model.

```
struct Movie: Codable {
    let adult: Bool?
    let genre_ids: [Int]?
    let id: Int?
    let original_title, overview: String?
    let popularity: Double?
    let poster_path: String?
    let release_date, title: String?
    let video: Bool?
    let vote_average: Double?
    let vote_count: Int?
}

struct Dates {
    let maximum, minimum: String
}
```

**Programski kod 3.1.** „Movie“ struktura koja predstavlja model.

View je sloj koji je odgovoran za prikaz korisničkog sučelja. On ne sadrži nikakvu poslovnu logiku već samo definira izgled aplikacije ili dijela aplikacije. Također View reagira na promjene u podacima koje dolaze iz ViewModela što znači ukoliko se dogodi bilo kakva promjena u podacima koji se prikazuju na zaslonu, View će se ažurirati i prikazati korisniku najnovije podatke. U iOS aplikaciji, View je često implementiran pomoću UIView komponenti, a može uključivati UI elemente poput slika, labela, tablica i još puno toga što je Apple dizajnirao. U Cineverse iOS aplikaciji View je predstavljen „MovieCard.swift“ datotekom koja sadrži MovieCard klasu koja implementira razne UIView elemente poput UIImageView, UIStackView, UILabel ili UIButton.

**Slika 3.6.** „*MovieCard.swift*“ datoteka koja sadrži implementaciju za View.

```
private lazy var stackView: UIStackView = {  
    let stack = UIStackView()  
    stack.axis = .vertical  
    stack.alignment = .leading  
    stack.distribution = .equalSpacing  
    stack.spacing = 2  
    stack.translatesAutoresizingMaskIntoConstraints = false  
    return stack  
}()
```

**Programski kod 3.2.** *UIStackView* koji čini jedan od komponenti *MovieCard View-a*.

ViewModel je posrednik između Modela i View-a. On preuzima podatke iz Modela i transformira ih u format potreban za prikazivanje u korisničkom sučelju. ViewModel obrađuje korisničke interakcije i šalje odgovarajuće podatke View-u, osiguravajući da su podaci uvijek ažurirani i pravilno formatirani. ViewModel ne zna direktno za View, što omogućava bolju testabilnost i modularnost. To također znači da ViewModel može funkcionirati neovisno o tome kako je sučelje implementirano. U Cineverse iOS aplikaciji ViewModel je predstavljen „*SwipeViewModel.swift*“ datotekom i ona sadrži *SwipeViewModel* klasu koja implementira poslovnu logiku poput objavljiivača[19] (eng. publisher), metode za API poziv, metodu za filtriranje filmova i druge.

**Slika 3.7.** „*SwipeViewModel.swift*“ datoteka koja sadrži implementaciju za View Model.

Temeljni dio aplikacije koji ujedno i spaja model, view i view model jest *UIViewController*. *ViewController* klasa je zadužena za interakciju između korisnika i aplikacije. Svaki ekran aplikacije obično se kontrolira jednim View Controller-om, a on ima glavnu ulogu u prikazu, organizaciji i reagiranju na korisničke interakcije. Svaki View Controller prolazi kroz određene faze svog životnog ciklusa, koje omogućuju pravilno rukovanje memorijom i resursima aplikacije. Neke od važnih metoda koje View Controller koristi su: *viewDidLoad()*: metoda koja se poziva kada je view učitana u memoriju. *WillAppear()*: metoda koja se poziva prije nego što View postane vidljiv na ekranu. *DidAppear()*: metoda koja se poziva nakon što je View postao vidljiv i druge. Cineverse iOS aplikacija sadrži jedan glavni *ViewController*.

**Slika 3.8.** „*SwipeViewController.swift*“ datoteka koja sadrži implementaciju za *ViewController*.

### 3.3. Dobivanje podataka o filmovima

The Movie Database (TMDB)[20] jedna je od najpopularnijih baza podataka koja nudi informacije o filmovima, TV emisijama i glumcima. Platforma omogućuje pristup širokom spektru podataka putem svog API-a, poput opisa filmova, filmskih promotivnih plakata i mnogih drugih filmskih podataka. Inače TMDB je i open-source zajednica, što znači da su podaci o filmovima često ažurirani od strane globalne korisničke baze, omogućujući točne i sveobuhvatne informacije.



**Slika 3.9.** *TMDB logo*.

Da bi se TMDB API mogao koristiti, potrebna je registracija korisničkog računa na TMDB web stranici gdje se potom trebaju preuzeti jedinstveni API ključ i Bearer token. Api ključ je jedinstven i omogućuje pristup JSON podacima. Inače služi kao identifikator korisnika i daje određena prava na korištenje API-ja. API ključ je nužan za dobivanje API podataka i potrebno ga je uključiti u URL adresu na koji se upućuje API poziv, inače će zahtjev biti odbijen. Svaka aplikacija ili sustav koji koristi TMDB API mora osigurati da je API ključ siguran i pravilno implementiran kako bi spriječio zloupotrebu, te uspješno bio autoriziran. Bearer token je tip sigurnosnog tokena koji se koristi za autentifikaciju i autorizaciju prilikom slanja zahtjeva prema API-u. Kada aplikacija koristi Bearer token, on se šalje u zaglavlju HTTP zahtjeva kao dokaz da je klijent ovlašten za pristup određenim podacima ili resursima. TMDB API omogućuje skidanje samo dvadeset filmova po API pozivu. Ukoliko želimo još podataka potrebno je URL adresu dinamički ažurirati na sljedeću stranicu s koje se potom dobije lista s novih dvadeset filmova i njihovim pripadajućim podacima.



```

class ApiCaller {

    static let shared = ApiCaller()

    func getMovieList(at page: Int) -> AnyPublisher<[Movie], Error> {
        guard let url = URL(string: URLs.apiUrl + "\(page)") else {
            return Fail(outputType: [Movie].self, failure:
FetchPostsError.wrongUrl).eraseToAnyPublisher()
        }
        var request = URLRequest(url: url)
        request.httpMethod = "GET"
        request.setValue("Bearer \(Keyes.ApiReadAccessToken)", forHTTPHeaderField:
"Authorization")
        request.setValue("application/json", forHTTPHeaderField: "accept")

        return URLSession.shared.dataTaskPublisher(for: request)
            .receive(on: DispatchQueue.main)
            .tryMap { result in
                guard let httpResponse = result.response as? HTTPURLResponse,
                    httpResponse.statusCode >= 200 && httpResponse.statusCode < 300 else {
                    throw FetchPostsError.responseNotOk
                }

                let decoder = JSONDecoder()
                guard let response = try? decoder.decode(Result.self, from: result.data) else {
                    throw FetchPostsError.unknown
                }
                guard let movies = response.results else {
                    throw FetchPostsError.unknown
                }
                return movies
            }
            .eraseToAnyPublisher()
    }
}

```

**Programski kod 3.3.** *Asinkrona metoda API poziva pisana u Swift programskom jeziku.*

### 3.4. Combine

Combine[21] je radni okvir kojeg je Apple uveo u iOS 13, macOS 10.15, watchOS 6 i tvOS 13 za rad s reaktivnim programiranjem. To je alat koji omogućava razvoj aplikacija koristeći deklarativni stil, omogućujući upravljanje događajima poput promjena podataka, korisničkih akcija ili mrežnih zahtjeva kroz kombiniranje i obradu asinkronih tokova podataka. Ključni koncepti Combine-a su objavljiivač (eng. *publisher*) i pretplatnik (eng. *subscriber*). Objavljiivač je objekt koji emitira odnosno šalje podatke. Svaki objavljiivač definira svoj tip podataka kojeg može emitirati. Neki primjeri su Just, Future ili URLSessionDataTaskPublisher za mrežne zahtjeve. Za razliku od objavljiivača, pretplatnik je objekt koji prima podatke od objavljiivača. Pretplatnik se pretplaćuje na objavljiivača i osluškuje primljene vrijednosti ili pogreške. Dva osnovna pretplatnička tipa u Combine-u su Sink i Assign. Combine nudi različite operatore koji omogućuju transformaciju, filtriranje i upravljanje podacima koje emitira objavljiivač. Primjeri operacija su map, filter, reduce, flatMap, itd. Objekti koji upravljaju pretplatama zovu se ukidači (eng. *cancellables*). Kada se pretplata više ne koristi, može se otkazati kako bi se izbjeglo curenje memorije. Combine kao radni okvir vrlo je koristan za rad s asinkronim podacima, kao što su mrežni zahtjevi ili podaci koji se učitavaju u pozadini jer omogućuje pripremu podataka bez da to zna korisnik. Također Combine čini kod urednijim i preglednijim.



**Slika 3.10.** *Combine logo.*

U Cineverse iOS aplikaciji Combine se koristi za asinkrono API pozivanje, te za upravljanje određenim stanjima aplikacije. Stanja u aplikaciji su: `.empty` (niz u koji se spremaju podaci o filmovima je prazan), `.filter` (nakon API poziva, filmovi se filtriraju), `.loading` (filmovi su spremni i generiranje kartica može početi) i `.loaded` stanje (kartice s filmovima su generirane i prikazane na zaslonu, te se vrši pražnjenje niza podataka kako bi sve bilo spremno za spremanje idućih dvadeset filmova kada se stanje postavi ponovno na `.empty`).

```
class SwipeViewModel {  
    @Published var unfilteredMovies: [Movie]  
    @Published var state: State  
    private var bag: Set<AnyCancellable>
```

**Programski kod 3.4.** *Implementacija objavljiivača u SwipeViewModel klasi.*

```

swipeViewModel.$unfilteredMovies
    .combineLatest(swipeViewModel.$state)
    .receive(on: RunLoop.main)
    .sink { [weak self] movies in
        self?.reloadState()
    }
    .store(in: &bag)

```

**Programski kod 3.5.** *Implementacija pretplatnika u `SwipeViewController` klasi.*

### 3.5. Obrada podataka

Obrada podataka u Cineverse iOS aplikaciji koja preporučuje filmove važan je korak u kompletnom radu aplikacije jer bez podataka o filmovima ne bi imali što za prikazivati. Obrada podataka omogućuje pripremu, filtriranje i transformaciju podataka dobivenih od TMDb API-a u format koji aplikacija zna i može koristiti. Ovaj proces obuhvaća dohvaćanje podataka, formatiranje te prilagodbu korisničkom sučelju. Iz tog razloga napravio se „Movie“ model koji predstavlja jedan model filma. Također taj model filma sadrži osnovne atribute o filmovima, poput naslova, linkova promotivnih plakata, žanrova i datuma izlaska filma. Svaki atribut je određen tipom podatka. Primjerice naslov filma je String tip, dok su žanr filma predstavljen nizom cjelobrojnih vrijednosti koji se kasnije dekodiraju u string tip podatka koji se potom prikazuju na zaslonu. Svi tipovi podataka su opcionalnog tipa iz razloga ako se API pozivom iz nekog razloga ne dobije traženi podatak aplikacija taj atribut može označiti s „nil“ vrijednosti bez posljedica za program nakon čega se izvođenje nastavlja. Nakon što su podaci dohvaćeni, oni se moraju pravilno parsirati te se spremaju u memoriju.

Podaci se dohvaćaju pomoću URLSession dataTask objavljiivača, a mogu i s drugim alatima za mrežnu komunikaciju. Zatim se pretvaraju u odgovarajuće modele pomoću „Codable“ protokola. JSON podaci moraju biti parsirani i pretvoreni u instance modela unutar aplikacije. Svaki atribut modela filma, kao što je žanr ili datum izlaska, mapira se na pripadajuća svojstva modela koji aplikacija koristi. Kod parsiranja datuma, JSON vraća datum u formatu "YYYY-MM-DD", koji zatim mora biti pretvoren u format prikladan za prikaz korisniku, kao što je "dd.MM.yyyy". Gdje „d“ označava dan u mjesecu, „M“ kalendarski mjesec, a „yyyy“ godinu.

U idućoj ovoj fazi osigurava se da su svi dohvaćeni podaci ispravni i cjeloviti. Ako neki podaci nedostaju ili su nevaljani, potrebno ih je zamijeniti zadanim vrijednostima kako bi aplikacija nastavila funkcionirati bez problema. Tomu služi opcionalni tip podatka. Nakon validacije, podaci se formatiraju kako bi odgovarali korisničkom sučelju. Na primjer, polja poput popisa žanrova ili ocjena moraju biti pravilno formatirana kako bi se prikazivala na način koji je čitljiv korisnicima. Na primjer, žanrovi. ID broj žanra pretvara se u naziv poput "Akcija", "Komedija" ili slično. Ako film pripada više žanrova, prikazuje se samo prvi, onaj glavni žanr, prema specifičnim pravilima aplikacije. Posljednji korak u obradi podataka je njihovo mapiranje na odgovarajuću poziciju na zaslonu. Kartice su predstavljene MovieCard objektima i u MovieCard klasi se za svaku generiranu karticu točno predaje filmski objekt koji sadržava podatke koji će na

toj kartici biti prikazani. Generiranje kartica će biti detaljnije opisano u sljedećem poglavlju. Ova obrada podataka omogućuje da aplikacija na brz i efikasan način prikaže podatke o filmovima, omogućujući korisnicima jednostavan pregled i interakciju.

### 3.6. Generiranje filmskih kartica

Generiranje filmskih kartica u Cineverse iOS mobilnoj aplikaciji definirano je „GenerateCards()“ metodom. Ona iteracijskom metodom prolazi nizom podataka koji sadržava filmove s TMDb API-a, te za svaki film instancira objekt „MovieCard“ klase s pripadajućim UI atributima. Potom se kartica dodaje kao podpogled na view controller. Prilikom kreiranja objekta za svaki film, svakom objektu se dodaje „UIPanGesture“ prepoznavatelj koji omogućuje pomicanje kartice u svim smjerovima. Pomicanje u svim smjerovima će biti više opisano u sljedećem poglavlju.

### 3.7. Odabiranje preferiranih filmova

Za pomicanje generiranih filmskih kartica u svim smjerovima zadužena je „handlePanGesture()“ metoda koja se nalazi u view controlleru. Metoda prima argument tipa „UIPanGestureRecognizer[22]“ koja kontinuirano prepoznaje i prati horizontalno i vertikalno povlačenje objekta prstom, u ovom slučaju filmske kartice po zaslonu uređaja. Metoda sadržava varijable u koje se spremaju određene izračunate vrijednosti poput udaljenosti centra filmske kartice od ruba zaslona ili veličinu skaliranja kartice prilikom ukljanjanja kartice sa zaslona. Metoda ima dva slučaja. Oba slučaja opisuju događaj kada prst filmsku karticu prestaje povlačiti po zaslonu odnosno kada je gesta završila. Prvi slučaj je ako centar kartice nije došao do lijeve ili desne određene granice postavljene od ruba zaslona i drugi slučaj ukoliko jest. Ako kartica pak nije prešla granicu a pomicanje je završeno, kartica se ponovno centrira na vrh postojećeg stoga filmova. Ukoliko je prešla granicu, kartica se uklanja sa zaslona, briše se i ovisno je li završila na lijevom rubu zaslona kao takva ne ulazi u željene preference filmova. Ako pak završi na desnom rubu zaslona i prijeđe granicu, žanr filma se smatra poželjnim i žanr se sprema za buduće filtriranje. Filtriranje će biti opisano u sljedećem poglavlju. Granica je postavljena sedamdeset i pet jedinica od horizontalnih rubova zaslona.

### 3.8. Filtriranje

Filtriranje vrši „filterMovies()“ metoda iz SwipeViewModel klase. Filmovi se filtriraju na temelju žanrova. Treba napomenuti kako TMDb žanrove daje kao niz cjelobrojnih vrijednosti, pa je potrebno dekodirati ih u string tipove podataka ovisno o kojem žanru se radi (Programski kod 3.6.). Klasa SwipeViewModel sadrži prazan niz u kojeg se spremaju zadnje tri cjelobrojne vrijednosti najpreferiranija filmska žanra koje je korisnik odabrao filmskim karticama. Niz može sadržavati samo jednu istu vrijednost žanra, a maksimalno tri. Ukoliko se odabere četvrti žanr, taj se sprema, a najstariji se žanr briše iz niza. Pomicanjem kartica lijevo ili desno s ruba zaslona SwipeViewController broji koliko generiranih filmskih kartica je preostalo i kada ih preostane šest objavljiivač stanja napravi API poziv. Dohvati se novi niz od 20 filmova koji se potom isfiltrira prema posljednja tri najpreferiranija žanra spremljeni u nizu za preferirane žanrove.

```
enum MovieGenre: Int {  
    case action = 28
```

```

case adventure = 12
case animation = 16
case comedy = 35
case crime = 80
case documentary = 99
case drama = 18
case family = 10751
case fantasy = 14
case history = 36
case horror = 27
case music = 10402
case mystery = 9648
case romance = 10749
case scienceFiction = 878
case tvMovie = 10770
case thriller = 53
case war = 10752
case western = 37

var name: String {
    switch self {
        case .action: return "Action"
        case .adventure: return "Adventure"
        case .animation: return "Animation"
        case .comedy: return "Comedy"
        case .crime: return "Crime"
        case .documentary: return "Documentary"
        case .drama: return "Drama"
        case .family: return "Family"
        case .fantasy: return "Fantasy"
        case .history: return "History"
        case .horror: return "Horror"
        case .music: return "Music"
        case .mystery: return "Mystery"
        case .romance: return "Romance"
        case .scienceFiction: return "Science Fiction"
        case .tvMovie: return "TV Movie"
        case .thriller: return "Thriller"
        case .war: return "War"
        case .western: return "Western"
    }
}

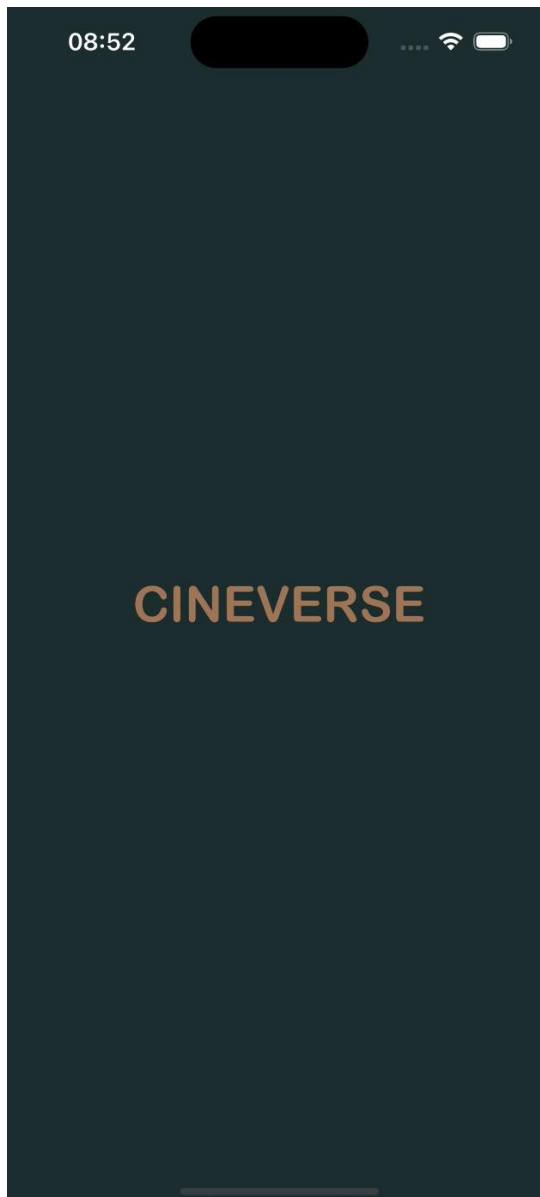
```

**Programski kod 3.6.** Enum za dekodiranje cjelobrojnih žanr vrijednosti u string[23].

## 4. TESTIRANJE

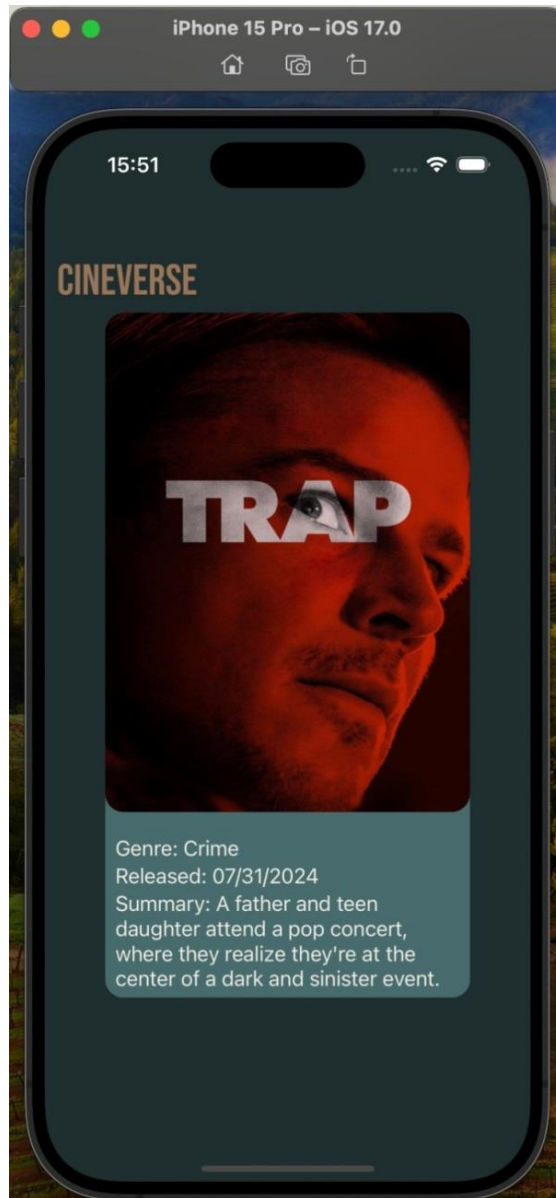
Aplikacija je testirana na dva uređaja. Prvi je iPhone SE treće generacije s iOS verzijom operacijskog sustava 17.5, a drugi uređaj iPhone 15 Pro s verzijom operacijskog uređaja 17.0. Aplikacija uredno radi i uspješno filtrira, a korisničko sučelje je stabilno.

Iako aplikacija koristi algoritamsko filtriranje ono radi zadovoljavajuće. Preporučena nadogradnja bi bila istrenirati strojni model i ugraditi ga u filtriranje aplikacije. Iz razloga jer algoritamsko filtriranje nema mogućnost učenja iz podataka korisnika, ne upravlja složenim podacima i nema mogućnosti potpune personalizacije. U sljedećem dijelu bit će opisan rad aplikacije tijekom testiranja na simulatoru iPhone 15 Pro. Nakon toga će biti opisana i analizirana zastupljenost žanrova TMDb API preporuka, točnost preporuka dobivenih rezultata i upotreba resursa na iPhone 15 Pro simulatoru, a rezultati će biti prikazani grafikinom 4.1. i tablicama 4.1. i 4.2.

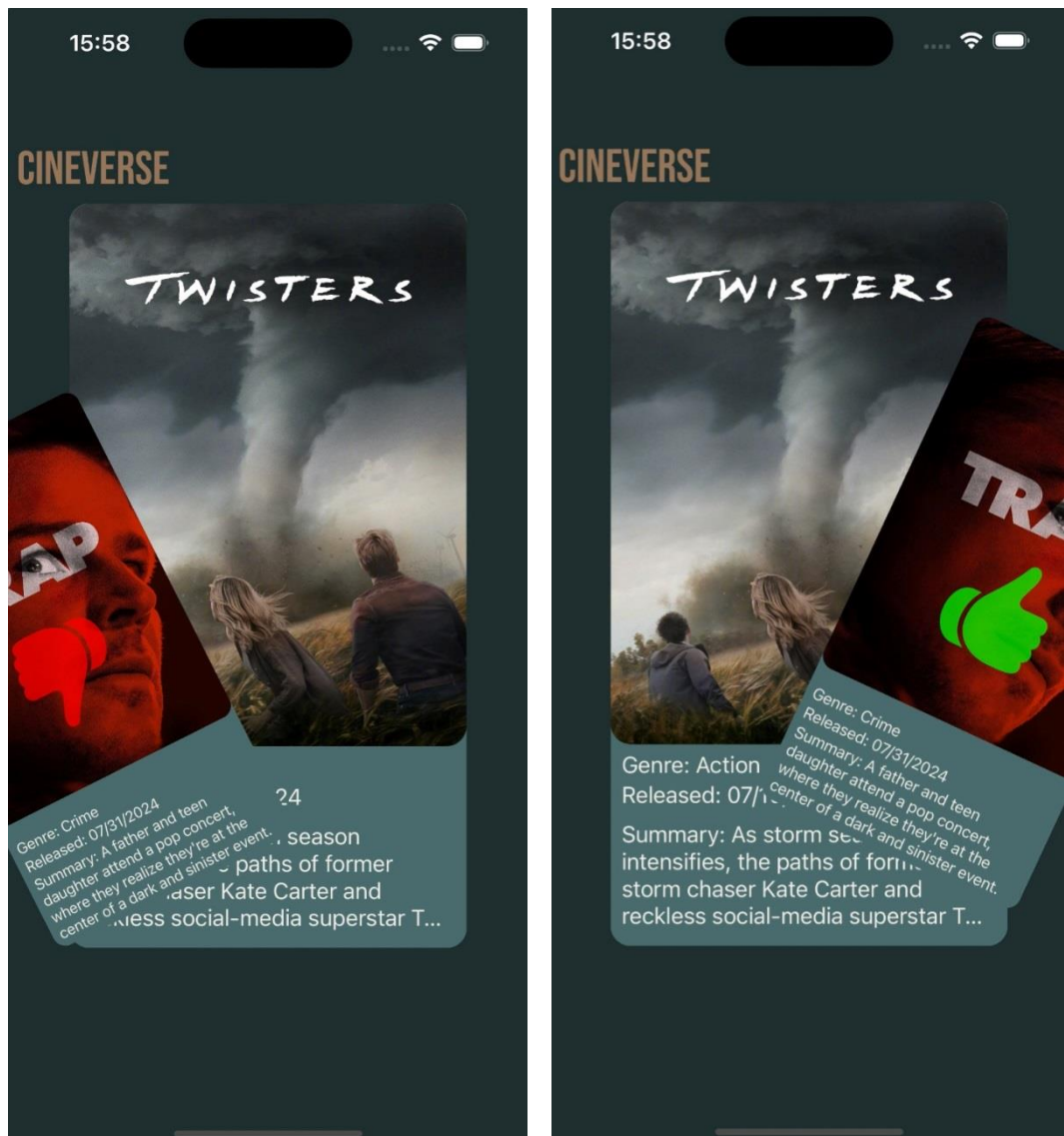


**Slika 4.1.** Početni zaslon Cineverse iOS aplikacije.

Pokretanjem Cineverse mobilne aplikacije otvara se početni zaslon aplikacije koji sadržava naslov aplikacije. Nakon vrlo kratkog vremena vidljiva je prva filmska kartica s pripadajućim informacijama o žanru filma, datumu izlaska, te kratkog opisa radnje filma. Karticu je moguće pomicati u svim smjerovima, a ispod navedene kartice vidljiva je iduća kartica spremna za prikaz korisniku.



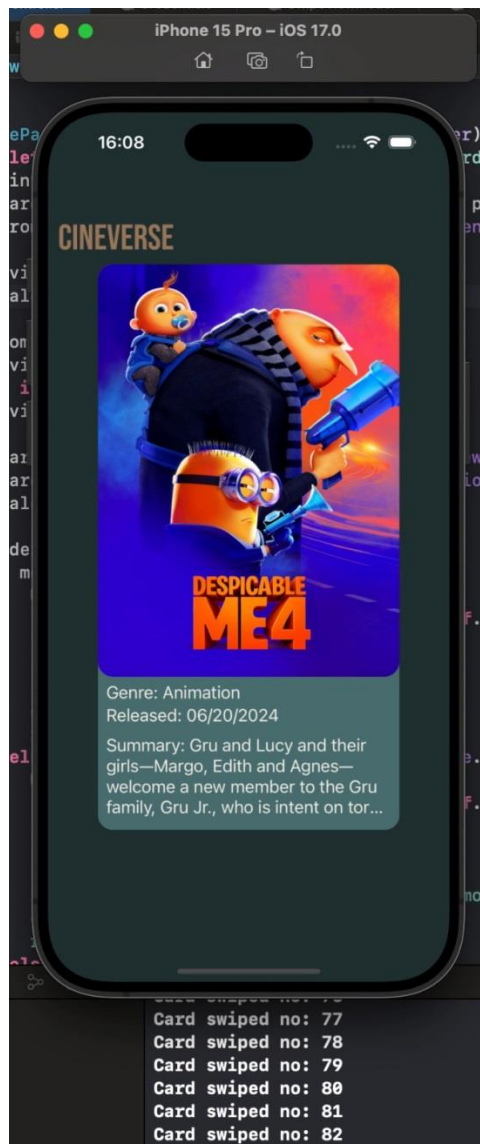
**Slika 4.2.** Početni zaslon Cineverse iOS aplikacije s prikazom prve filmske kartice.



**Slika 4.3.** *Primjer pomicanja filmske kartice na lijevi odnosno desni rub zaslona.*

Skidanjem filmske kartice s ekrana na desnu stranu aplikacija pamti žanr i na nadolazećim filmskim karticama filtrira filmove koji imaju isti žanr. Skidanjem filmske kartice s ekrana na lijevu stranu aplikacija nastavlja dalje s prikazivanjem potencijalno zanimljivih filmova korisniku. Testirano je više od sto pomicanja i skidanja filmske kartice iako TMDB API pruža preko jedan milijun filmova.





**Slika 4.4.** Testiranje broja filmskih kartica, vidljivo je kako se kartice generiraju prema visokom broju filmskih kartica.

U nastavku analizat će se trenutna zastupljenost žanrova TMDB API preporuka. Popis filmova na kojima se temelji ispitivanje najzastupljenijih žanrova prikazan je tablicom 4.1.

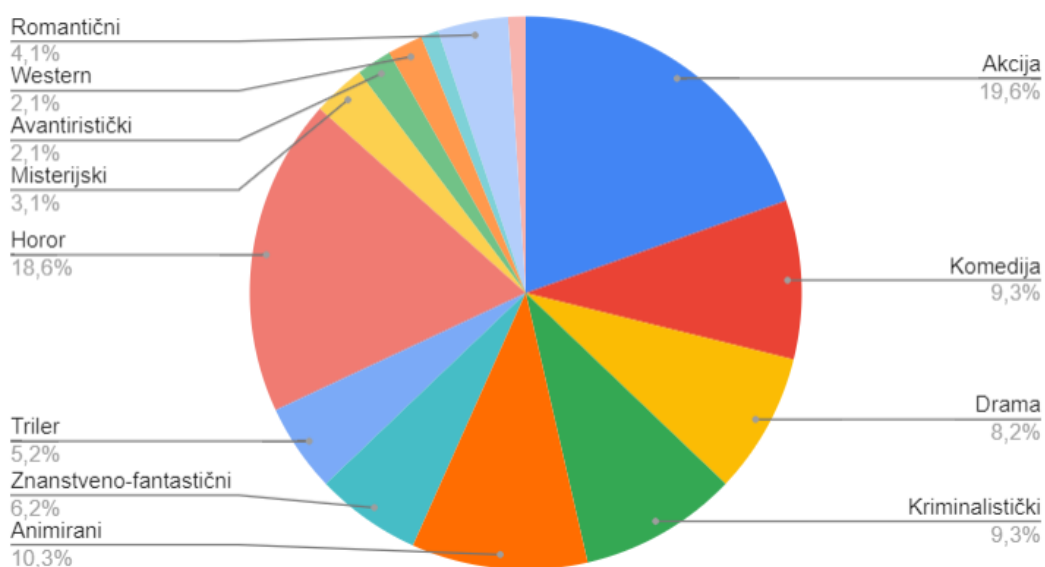
**Tablica 4.1.** Prvih sto trenutnih filmskih rezultata s pripadajućim filmskim žanrom.

<i>Film ID</i>	<i>Naslov filma</i>	<i>Žanr filma</i>	<i>Film ID</i>	<i>Naslov filma</i>	<i>Žanr filma</i>
<b>1.</b>	Borderlands (2024)	Akcijski	<b>51.</b>	Fly me to the Moon (2024)	Romantični
<b>2.</b>	Rebel Ridge (2024)	Kriminalistički	<b>52.</b>	Kinds of kindness	Komedija
<b>3.</b>	El candidato honesto (2024)	Komedija	<b>53.</b>	The Lion King (1994)	Obiteljski
<b>4.</b>	The Crow (2024)	Akcijski	<b>54.</b>	Bandida a numero um (2024)	Akcijski
<b>5.</b>	Beetlejuice (2024)	Komedija	<b>55.</b>	Encanto (2021)	Animirani
<b>6.</b>	Twisters (2024)	Akcijski	<b>56.</b>	Abigail (2024)	Horor
<b>7.</b>	Alien Romulus (2024)	Horor	<b>57.</b>	Spider-Man 3 (2007)	Akcijski
<b>8.</b>	It ends with us (2024)	Romantični	<b>58.</b>	In a violent nature (2024)	Horor
<b>9.</b>	Beetlejuice (2024)	Znanstveno-fantastični	<b>59.</b>	Monkey man (2024)	Akcijski
<b>10.</b>	Uglies (2024)	Znanstvena-fantastični	<b>60.</b>	Lembayung (2024)	Horor
<b>11.</b>	The Garfield Movie (2024)	Animirani	<b>61.</b>	Breathe (2024)	Akcijski
<b>12.</b>	Transformers One (2024)	Animirani	<b>62.</b>	How to make millions before grandma dies (2024)	Drama
<b>13.</b>	The Killer (2024)	Akcijski	<b>63.</b>	Hit man (2024)	Kriminalistički
<b>14.</b>	Longlegs (2024)	Kriminalistički	<b>64.</b>	The Kingtide (2024)	Triler
<b>15.</b>	Kill (2024)	Akcijski	<b>65.</b>	Horizon (2024)	Western
<b>16.</b>	Gunner (2024)	Akcijski	<b>66.</b>	The Well (2024)	Horor
<b>17.</b>	Trap (2024)	Triler	<b>67.</b>	Rosalie (2024)	Drama
<b>18.</b>	The Union (2024)	Akcijski	<b>68.</b>	The boy and the Heron (2024)	Animirani
<b>19.</b>	Speak no evil (2024)	Horor	<b>69.</b>	Didi (2024)	Komedija
<b>20.</b>	The convert (2024)	Akcijski	<b>70.</b>	Shrek 2 (2004)	Animirani
<b>21.</b>	Subservience (2024)	Znanstveno-fantastični	<b>71.</b>	The Forge (2024)	Drama
<b>22.</b>	Prey (2024)	Akcijski	<b>72.</b>	Wolfs (2024)	Kriminalistički
<b>23.</b>	Officer black belt (2024)	Akcijski	<b>73.</b>	Child star (2024)	Dokumentarni
<b>24.</b>	Dragonkeeper (2024)	Akcijski	<b>74.</b>	Hounds of War (2024)	Akcijski
<b>25.</b>	The Deliverance (2024)	Horor	<b>75.</b>	Tatami (2024)	Triler
<b>26.</b>	Cinderellas Curse	Horor	<b>76.</b>	Exhuma (2024)	Misterijski

	(2024)				
<b>27.</b>	Killer Babes (2024)	Komedija	<b>77.</b>	Alienoid return to the future (2024)	Znanstveno-fantastični
<b>28.</b>	Techno Boys (2024)	Komedija	<b>78.</b>	(Un)lucky sisters (2024)	Komedija
<b>29.</b>	The Exorcism (2024)	Horor	<b>79.</b>	Sting (2024)	Horor
<b>30.</b>	Wanted Man (2024)	Akcijski	<b>80.</b>	Never let go (2024)	Horor
<b>31.</b>	Blink Twice (2024)	Misterijski	<b>81.</b>	Matrix (1999)	Akcijski
<b>32.</b>	Farang (2024)	Akcijski	<b>82.</b>	Despicable Me (2024)	Animirani
<b>33.</b>	Crescent City (2024)	Triler	<b>83.</b>	Star wars (2024)	Avanturistički
<b>34.</b>	The mouse trap (2024)	Horor	<b>84.</b>	Tumbbad (2018)	Znanstveno-fantastični
<b>35.</b>	The Wild Robot (2024)	Animirani	<b>85.</b>	Hijack 1971 (2024)	Kriminalistički
<b>36.</b>	Coraline (2024)	Animirani	<b>86.</b>	The bikeriders (2024)	Kriminalistički
<b>37.</b>	The Substance (2024)	Drama	<b>87.</b>	Love lies bleeding (2024)	Kriminalistički
<b>38.</b>	Something in the water (2024)	Triler	<b>88.</b>	The Count of MonteCristo (2024)	Avanturistički
<b>39.</b>	Maxxxine (2024)	Horor	<b>89.</b>	Detective Conan (2024)	Animirani
<b>40.</b>	Oddity (2024)	Horor	<b>90.</b>	The dead do not hurt (2024)	Western
<b>41.</b>	Thelma (2024)	Akcija	<b>91.</b>	Uranus 2324 (2024) (2024)	Romantični
<b>42.</b>	Joker (2024)	Kriminalistički	<b>92.</b>	Uhaw (2024)	Drama
<b>43.</b>	Beautiful wedding (2024)	Romantični	<b>93.</b>	Megalopolis (2024)	Drama
<b>44.</b>	The Strangers (2024)	Horor	<b>94.</b>	Like father like son (2024)	Kriminalistički
<b>45.</b>	Godzilla Minus One (2024)	Znanstveno-fantastični	<b>95.</b>	Robor dreams (2024)	Animacijski
<b>46.</b>	The Jack In The Box Rises (2024)	Horor	<b>96.</b>	Crossing (2024)	Drama
<b>47.</b>	Boxer (2024)	Drama	<b>97.</b>	Nice girls (2024)	Komedija
<b>48.</b>	Cuckoo (2024)	Horor	<b>98.</b>	New life (2024)	Horor
<b>49.</b>	Incoming (2024)	Komedija	<b>99.</b>	Monster (2024)	Misterijski
<b>50.</b>	Kung Fu Games (2024)	Akcija	<b>100.</b>	The wandering eart II (2024)	Znanstveno-fantastični

S obzirom na zastupljenost žanrova u tablici 4.1. može se zaključiti da je akcijski žanr najzastupljeniji s 19 filmova, što čini značajan dio ukupnog broja. Horror filmovi također imaju veliku zastupljenost s 18 naslova što sugerira popularnost ovog žanra u trenutnoj ponudi. Animirani i kriminalistički filmovi su zastupljeni s po 9 do 10 naslova što ukazuje na široku privlačnost ovih žanrova. Manje zastupljeni žanrovi poput western, dokumentarnih i obiteljskih filmova sugeriraju da su specifičnijih žanrova koji nisu toliko dominantni u trenutnom opsegu filmova dostupnih putem TMDB API-a. Treba primjetiti da postoje i stariji filmovi u rezultatima, a razlog je njihova preporučljivost i dan danas. Zaključak je da aplikacija nudi široki izbor, ali s naglaskom na akciju i horor, što može zadovoljiti veći dio publike koja preferira te žanrove. Grafički prikaz udjela svih žanrova prikazan je na grafikonu 4.1.

### Žanr filmova i njihov udio u prvih 100 rezultata



**Grafikon 4.1.** Grafički prikaz trenutne zastupljenosti žanrova TMDB API-a.

Grafikon 4.1. prikazuje zastupljenost različitih filmskih žanrova u uzorku od sto filmova koje je aplikacija dohvatila s TMDB API-ja. Svaki segment grafikona predstavlja udio filmova unutar određenog žanra čime se vizualno prikazuje koliko su pojedini žanrovi zastupljeni u trenutnom skupu podataka. Cilj ovog grafikona je prikazati raznolikost sadržaja u bazi podataka TMDB-a u stvarnom vremenu. Ovakav način prikaza omogućuje brzi uvid u strukturu dostupnih žanrova i može pomoći u daljnjoj analizi preferencija korisnika u aplikaciji.

Tablica 4.2. prikazuje rezultate testiranja točnosti preporuka u aplikaciji koja koristi algoritamsko filtriranje s jasno definiranim pravilima filtriranja prema žanrovima. U ovom pristupu aplikacija temelji preporuke isključivo na korisničkim preferencijama za određene žanrove bez mogućnosti učenja iz prethodnih korisničkih interakcija. Tablica uspoređuje očekivane rezultate sa stvarnim ishodima kako bi se procijenila relevantnost preporuka za svaki pojedini žanr. Algoritamsko filtriranje kao metoda dobro funkcionira za jednostavnije slučajeve

poput prikaza filmova prema specifičnim žanrovima. Testiranje je obuhvatilo različite žanrove, uključujući akcijske, kriminalističke, horor, dramske i znanstveno-fantastične filmove. Rezultati pokazuju da aplikacija u svim slučajevima uspješno pruža relevantne preporuke za navedene žanrove s visokim stupnjem preciznosti. Iako su preporuke za sve žanrove bile točne u većini slučajeva, primijećena je niža preciznost zbog nedostatka personalizacije i sposobnosti aplikacije da uči iz korisničkih preferencija kroz vrijeme. To ukazuje da bi integracija strojnog učenja u aplikaciju mogla poboljšati relevantnost preporuka u složenijim slučajevima.

**Tablica 4.2.** *Tablica testiranja točnosti preporuka.*

<b><i>Testirani slučaj</i></b>	<b><i>Kriterij</i></b>	<b><i>Očekivani rezultat</i></b>	<b><i>Stvarni rezultat</i></b>	<b><i>Stvarna filmska preporuka</i></b>	<b><i>Zadovoljenost kriterija</i></b>
<b><i>Testiranje preporuka</i></b>	Relevantnost prema žanru akcija	Prikaz akcijskih filmova	Prikaz relevantnih akcijskih filmova	Borderlands (2024), Crow (2024), Twisters (2024)	Zadovoljen
<b><i>Testiranje preporuka</i></b>	Relevantnost prema žanru kriminalistički	Prikaz kriminalističkih filmova	Prikaz relevantnih kriminalističkih filmova	Rebel ridge (2024), Longlegs (2024)	Zadovoljen
<b><i>Testiranje preporuka</i></b>	Relevantnost prema žanru horor	Prikaz horor filmova	Prikaz relevantnih horor filmova	Alien romulus (2024), Speak no evil (2024)	Zadovoljen
<b><i>Testiranje preporuka</i></b>	Relevantnost prema žanru drama	Prikaz dramskih filmova	Prikaz relevantnih dramskih filmova	The supstance (2024), Boxer (2024), Kinds of kindness (2024)	Zadovoljen
<b><i>Testiranje preporuka</i></b>	Relevantnost prema žanru znanstvena fantastika	Prikaz znanstveno-fantastičnih filmova	Prikaz relevantnih znanstveno-fantastičnih filmova	Subservience (2024), Ugliers (2024),	Zadovoljen

Tablica 4.3. prikazuje rezultate testiranja upotrebe resursa aplikacije tijekom rada mobilne aplikacije i filtriranja prema žanrovima. Aplikacija koristi jasno definirana pravila filtriranja filmova na temelju žanrova što zahtijeva procesiranje jednostavnih podataka, poput korisničkih odabira. Algoritamsko filtriranje nije resursno zahtjevno jer ne uključuje složene procese poput strojnog učenja ili analize korisničkih obrazaca ponašanja. Testiranje je provedeno na dva uređaja, iPhone 15 Pro i iPhone SE treće generacije. Ključne faze analize uključuju: Učitavanje aplikacije: uočeno je brzo učitavanje aplikacije jer se podaci o filmovima i žanrovima brzo dohvaćaju i obrađuju. Biranje i pregledavanje filmova: aplikacija koristi minimalne resurse tijekom pregledavanja kartica s filmovima, jer filtriranje prema žanrovima ne zahtijeva značajnu računalnu snagu. Filtriranje prema žanrovima: CPU i memorija korišteni za filtriranje su minimalni jer filtriranje zahtijeva minimalne resurse, a aplikacija pokazuje stabilne performanse na oba testirana uređaja. Rezultati pokazuju da aplikacija vrlo efikasno koristi resurse prilikom korištenja algoritamskog filtriranja. Budući da ovaj pristup ne uključuje intenzivno procesiranje podataka ili personalizaciju, potrošnja memorije i CPU-a ostaje niska. Razlike u performansama između starijih i novijih uređaja bile su minimalne, što ukazuje na to da aplikacija radi optimalno čak i na hardverski slabijim uređajima.

**Tablica 4.3.** *Tablica iskorištenosti i upotrebe resursa.*

<i>Faza korištenja aplikacije</i>	<i>Uređaj</i>	<i>Iskorištenost CPU</i>	<i>Iskorištenost radne memorije</i>
<i>Učitavanje aplikacije</i>	iPhone 15 Pro (iOS 17.0)	22%	21,3 MB
	iPhone SE (3rd generation) (iOS 17.5)	26%	19,5 MB
<i>Biranje i pregledavanje filmova lijevo-desno</i>	iPhone 15 Pro (iOS 17.0)	1%	98,6 MB
	iPhone SE (3rd generation) (iOS 17.5)	3%	69,6 MB
<i>Filtriranje prema žanrovima</i>	iPhone 15 Pro (iOS 17.0)	6%	19,2 MB
	iPhone SE (3rd generation) (iOS 17.5)	9%	17,8 MB

#### 4.1. Korisnici su ocijenili

Za potpunije mišljenje o radu aplikacije sedam anonimnih dobrovoljaca instaliralo je Cineverse iOS aplikaciju na način da kroz nekoliko dana isprobaju aplikaciju, te nakon toga ispune kratki upitnik, te daju svoje mišljenje na pitanje „Sviđa li vam se Cineverse, zašto i što biste voljeli promijeniti?“ o aplikaciji i sustavu. U nastavku nalaze se ukupno sedam komentara na postavljeno pitanje.

Korisnik A: „*Aplikacija izgleda moderno. Sviđa mi se minimalistički dizajn i tamno korisničko sučelje. Swajpanje kroz filmove je brzo. Volio bih vidjeti više informacija o svakom filmu poput detaljnijih podataka o glumcima i nagradama.*“

Korisnik B: „*Volim što aplikacija koristi TMDB API za preporuke. Već nakon nekoliko swajpova, otkrio sam filmove za koje nikad nisam čuo. Možda bi bilo dobro dodati mogućnost prilagođavanja filtara, npr. prema žanru ili ocjeni.*“

Korisnik C: „*Ako volite gledati filmove, ovo je aplikacija za vas. Preporuke su pogođene i algoritam radi odlično. Ipak, mislim da bi funkcionalnost mogla biti proširena dodavanjem TV serija u bazi podataka.*“

Korisnica D: „*Aplikacija radi glatko, bez zamrzavanja. Preporuke su točne. Jedini problem je što ne postoji opcija za spremanje filmova koje želite pogledati kasnije. To bi bila korisna funkcionalnost.*“

Korisnik E: „*Sviđa mi se što aplikacija koristi Tinder-style swajpanje za odabir filmova. To je zabavan način za pretraživanje. No, ponekad algoritam preporuča filmove istog žanra previše puta zaredom, bilo bi bolje kad bi algoritam bio malo raznovrsniji.*“

Korisnik F: „*Sviđa mi se jer je aplikacija jednostavna za korištenje. Swajpanje je također jednostavno i sviđa mi se thumb up/down feature, ali primijetio sam da neki manje poznati filmovi nisu uključeni u bazu. Dodavanje više indie filmova bi bilo odlično.*“

Korisnik G: „*Ova aplikacija je pravi dragulj za svakog filmofila. Volim način na koji se filmovi predstavljaju kroz jednostavno swajpanje, jer ne poznam niti jednu filmsku aplikaciju da ima takav način pretraživanja. Uglavnom su to dating aplikacije poput Badoo ili Tindera. Preporuke su precizne, a vizualni dizajn aplikacije moderan. Svakako mi se sviđa kako aplikacija povlači relevantne podatke o filmovima putem interneta i daje brze i točne rezultate. Jedino što bih volio vidjeti u budućim verzijama je možda mogućnost personaliziranih filtara za još bolju selekciju. Sve u svemu zadovoljan.*“

## 5. ZAKLJUČAK

Izrađeni sustav korisniku može preporučiti filtrirane filmove na osnovu njegovih osobnih preferenci na temelju žanra. Sustav za preporuku uklopljen je u iOS mobilnu aplikaciju pisana u Swift programskom jeziku u UIKit radnom okviru. Opisana je detaljna izrada mobilne aplikacije kao i metoda kojom se filmovi filtriraju. Testirana je kako bi se prikazao njezin rad. Za daljnji napredak sustava za preporuku moguće je istrenirati model strojnog učenja koji bi na drugačiji način pronašao korisniku željene preporuke, ali i dodavanja mogućnosti da se vidi više informacija o odabranom filmu prilikom odabiranja određene filmske kartice. Za sustav je još odlučeno testirati ga u praksi na način što je ukupno sedam anonimnih iOS korisnika instaliralo Cineverse mobilnu aplikaciju na svoj iPhone. Korisnici su također napisali svoje mišljenje u anonimnoj anketi koja je ispunila isčekivanja i dala do znanja da su korisnici zadovoljni. Također su predložili određene ideje i za usavršavanje aplikacije.



## LITERATURA

- [1] „The Movie Database (TMDB)“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://www.themoviedb.org/>
- [2] „Netflix official website“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://www.netflix.com/hr-en/>
- [3] „Filmoviplex - Online filmovi s prijevodom“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://www.filmoviplex.com/>
- [4] „Amazon movie recommendation site“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://www.amazon.com/gp/video/storefront>
- [5] „Tinder - Dating, Make Friends & Meet New People“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://tinder.com/>
- [6] „Badoo - Free Online Dating App“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://badoo.com/>
- [7] „Spotify“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://open.spotify.com/>
- [8] „Apple Music - Web Player“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://music.apple.com/us/new>
- [9] „AiRBnB search destinations and make a booking“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://www.airbnb.com/>
- [10] „Njuškalo; Croatian no.1. ad advertiser“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://www.njuskalo.hr/>
- [11] „Facebook Ads Manager“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: [https://www.facebook.com/business/tools/ads-manager?content\\_id=Xxyfzyu4h03cCYL&ref=sem\\_smb&utm\\_term=facebook%20ads&gclid=CjwKCAjw0aS3BhA3EiwAKaD2ZUrKq97WIG9\\_eE15tbA-W6OOaIDa6GO0D0aveX3D9smPb80uQOLAjhoCioqQAvD\\_BwE&gad\\_source=1](https://www.facebook.com/business/tools/ads-manager?content_id=Xxyfzyu4h03cCYL&ref=sem_smb&utm_term=facebook%20ads&gclid=CjwKCAjw0aS3BhA3EiwAKaD2ZUrKq97WIG9_eE15tbA-W6OOaIDa6GO0D0aveX3D9smPb80uQOLAjhoCioqQAvD_BwE&gad_source=1)
- [12] „Sustavi za preporuku temeljeni na stavkama - Medium“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://medium.com/@veribilimi35/item-based-recommendation-e2181d716962>
- [13] „Sustavi za preporuku temeljeni na korisnicima - Medium“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://medium.com/@deepapandithu/recommender-system-user-collaborative-filtering-37613f0c6a9>
- [14] „Google Assistant“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://assistant.google.com/>
- [15] „Apple Siri“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://www.apple.com/siri/>
- [16] „Sustavi za preporuku temeljeni na grupnom filtriranju - Medium“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://towardsdatascience.com/an-introduction-to-group-recommender-systems-8f942a06db56>
- [17] „XCode IDE“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://developer.apple.com/xcode/>
- [18] „MVVM iOS architectural pattern“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://medium.com/@zebayasmeen76/mvvm-in-ios-swift-6afb150458fd>
- [19] „Combine publishers“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://developer.apple.com/documentation/combine/publisher>
- [20] „The Movie Database faq“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://developer.themoviedb.org/docs/faq>
- [21] „Combine framework for iOS“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://developer.apple.com/documentation/combine>
- [22] „UIPanGestureRecognizer“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://developer.apple.com/documentation/uikit/uipangesturerecognizer>
- [23] „TMDB genre values“. Pristupljeno: 09. rujan 2024. [Na internetu]. Dostupno na: <https://www.themoviedb.org/talk/5daf6eb0ae36680011d7e6ee>

## SAŽETAK

U ovom diplomskom radu opisan je proces izrade iOS mobilne aplikacije Cineverse koja ima zadaću preuzimanja filmova s web stranice TMDB, obrade i prikazivanja filmova i pripadajućih podataka te ih obraditi i ponuditi korisniku film kao prijedlog. Korisnik potom ima mogućnost pregledati podatke o filmovima te određeni film osobnom preferencijom pogledati na nekom od drugih postojećih filmskih servisa u kinu ili u svom domu. Nakon gledanja film treba ocijeniti negativno ili pozitivno te si tako omogućiti preporuke sličnih filmova prema žanrovima. Mobilna aplikacija razvijena je u računalnom programu XCode na prijenosnom računalu MacBook Pro 2019. Rezultat je opravdao očekivanja i mobilna aplikacija se može koristiti, a također ima veliki potencijal za daljnji razvoj.

Ključne riječi: film, iOS, mobilna aplikacija, sustav za preporuku

## **ABSTRACT**

**Title:** Movie recommendation system

This master's thesis describes the process of creating an iOS mobile application Civerse, which has the task of downloading movies from the TMDB website, processing and displaying movies and associated data, and offering them to the user. The user then has the option to choose movies based on personal preference and after choosing and watching the movie, rate it negatively or positively and thus enable recommendations of similar movies based on genres. The mobile application was developed in the XCode computer program on a MacBook Pro 2019 laptop. The result has made expectations and the mobile application can be used, and also contains great potential for further development.

**Keywords:** iOS, mobile application, movie, recommendation system

## ŽIVOTOPIS

Robert Marković rođen je u Osijeku 7. listopada 1998. Osnovnu školu pohađao je također u Osijeku. 2013. godine upisuje Elektrotehničku školu Osijek, smjer elektrotehničar. Za vrijeme pohađanja srednje škole sudjeluje na brojnim školskim natjecanjima, međuškolskim natjecanjima na Fakultetu elektrotehnike i računarstva u Zagrebu, sajmu inovacija „iNova“, sajmu inovacija „Inventum“ s radovima u području elektrotehnike, elektronike i računarstva te Maker Fair izložbama radova u Elektrotehničkoj i prometnoj školi Osijek. 2016. godine sudjeluje na Erasmus+ razmjeni učenika te odlazi na dvotjednu stručnu praksu na South Devon University Technical College u Velikoj Britaniji. 2017. godine upisuje preddiplomski sveučilišni studij računarstva na Fakultetu elektrotehnike, računarstva i informacijskih tehnologija u Osijeku. 2021. godine na istom sveučilištu upisuje diplomski sveučilišni studij računarstva, modul DRD - Informacijske i podatkovne znanosti. Iste godine počinje se zanimati za iOS development u kojem nastavlja raditi i usavršavati se. Diplomski sveučilišni studij završava 2024.godine.

---

## **PRILOZI**