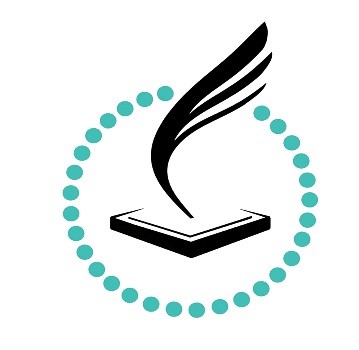
بسم الله الرحمن الرحیم



دانشگاه صنعتی جندی شاپور دزفول

دانشکده برق و کامپیوتر

پایان‌نامه کارشناسی رشته کامپیوتر

طراحی و پیاده‌سازی شبکه اجتماعی سینمایی شامل سیستم پیشنهاددهی

نگارش

**صادق همدانی‌پور**

استاد راهنما

**دکتر مریم چینی‌پرداز**

استاد مشاور

**دکتر مریم چینی‌پرداز**

اردیبهشت 1401

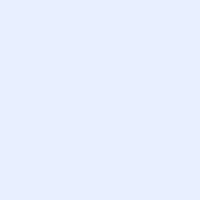
**برگ اصالت و مالکیت اثر**

این‌جانب صادق همدانی‌پور دانش‌آموختة کارشناسی رشتۀ کامپیوتر پدیدآور پایان‌نامه با عنوان " طراحی و پیاده‌سازی شبکه اجتماعی سینمایی شامل سیستم پیشنهاددهی" با راهنمایی دکتر مریم چینی‌پرداز گواهی و تعهد می‌کنم که بر پایۀ قوانین و مقررات، از جمله «دستورالعمل نحوۀ بررسی تخلفات پژوهشی» و همچنین «مصادیق تخلفات پژوهشی» مصوب وزارت علوم، تحقیقات و فناوری (۲۵ اسفند ۱۳۹۳):

* این پایان‌نامه دستاورد پژوهش این‌جانب و محتوای آن از درستی و اصالت برخوردار است؛
* حقوق معنوی همۀ کسانی را که در به‌دست‌‌آمدن نتایج اصلی پایان‌نامه تأثیرگذار بوده‌اند، رعایت کرده‌ام و هنگام کاربرد دستاورد پژوهش‌های دیگران در آن، با دقت و به‌درستی به آن‌ها استناد کرده‌ام؛
* این پایان‌نامه و محتوای آن را تاکنون این‌جانب یا کس دیگری برای دریافت هیچ‌گونه مدرک یا امتیازی در هیچ‌‌جا ارائه نکرده‌ایم؛
* همۀ حقوق مادی این پایان‌نامه از آن دانشگاه صنعتی جندی شاپور دزفول است و آثار برگرفته از آن با وابستگی سازمانی دانشگاه صنعتی جندی شاپور دزفول منتشر خواهد شد؛
* در همۀ آثار برگرفته از این پایان‌نامه، نام استاد(ان) راهنما و اگر استاد راهنمای نخست تشخیص دهد، نام استاد(ان) مشاور و نشانی رایانامۀ سازمانی آنان را می‌آورم؛
* در همۀ گام‌های انجام این پایان‌نامه، هرگاه به اطلاعات شخصی افراد یا اطلاعات سازمان‌ها دسترسی داشته یا آن‌ها را به‌کار برده‌ام، رازداری و اخلاق پژوهش را رعایت کرده‌ام.

تاریخ و امضا

**برگ تأیید هیئت داوران/ صورت‌جلسۀ دفاع (به زبان فارسی)**



**چکیده**

امروزه با افزایش محبوبیت فیلم‌های سینمایی بین مخاطبان، تنوع ژانری و محتوایی گسترده‌ای در میان فیلم‌های تولیدی ایجاد شده است. این تنوع محتوا، پروسه یافتن فیلم مناسب برای مخاطبان را به امری دشوار تبدیل کرده است. وجود یک سیستم هوشمند که توانایی شناسایی، تفکیک و رتبه‌بندی کاربران و فیلم‌ها بر اساس طبع و سلایق مختلف را داشته باشد، کمک شایانی به تسهیل پروسه فیلم‌یابی مخاطبان خواهد کرد. این سیستم هوشمند کاربران را قادر خواهد ساخت تا پیشنهادات مناسب و شخصی‌سازی شده‌ای را از سیستم دریافت و با کاربرانی با طبع و سلیقه مشابه خود آشنا شوند.

در این پروژه با استفاده از الگوریتم‌های پالایش گروهی، یک سیستم پیشنهاددهنده هوشمند را طراحی و پیاده‌سازی کرده و سپس با استفاده از فریم‌ورک‌های متداول طراحی وب نظیر Django و ‌VueJs یک محیط تعاملی تحت وب برای تعامل کاربران با سیستم و یکدیگر، طراحی و پیاده‌سازی می‌نماییم.

**کلیدواژه‌ها:**

**فهرست نوشتار**

[۱. مقدمه 17](#_Toc103948152)

[۱-۱. اهمیت موضوع 17](#_Toc103948153)

[۱-۲. تشریح پروژه 17](#_Toc103948154)

[۱-۳. توصیف رویکرد پیشنهاددهی 18](#_Toc103948155)

[۱-۳-۱. دسته‌بندی افراد 18](#_Toc103948156)

[۱-۳-۲. چالش موجود 18](#_Toc103948157)

[۱-۳-۳. راهکار 18](#_Toc103948158)

[۱-۳-۴. انواع فیلترینگ 19](#_Toc103948159)

[۱-۳-۵. نحوه استفاده در پروژه 19](#_Toc103948160)

[۲. پیاده‌سازی 21](#_Toc103948161)

[۲-۱. معماری نرم‌افزار 21](#_Toc103948162)

[۲-۲. ‌Backend (سرور) 21](#_Toc103948163)

[۲-۲-۱. مدل داده‌ها 21](#_Toc103948164)

[۲-۲-۲. Celery 23](#_Toc103948165)

[۲-۲-۳. سیستم پیشنهاددهنده 23](#_Toc103948166)

[۲-۲-۴. API 26](#_Toc103948167)

[۲-۳. Frontend (client app) 30](#_Toc103948168)

[۳. شرح عملکرد 31](#_Toc103948169)

[۳-۱. لاگین 31](#_Toc103948170)

[۳-۱-۱. ورود ناموفق 31](#_Toc103948171)

[۳-۱-۲. ورود موفق 32](#_Toc103948172)

[۳-۲. عضویت 32](#_Toc103948173)

[۳-۲-۱. عضویت ناموفق 32](#_Toc103948174)

[۳-۲-۲. عضویت موفق 33](#_Toc103948175)

[۳-۳. داشبورد 33](#_Toc103948176)

[۳-۳-۱. ویرایش اطلاعات 33](#_Toc103948177)

[۳-۳-۲. لیست کاربران مشابه 34](#_Toc103948178)

[۳-۴. پروفایل کاربر 34](#_Toc103948179)

[۳-۴-۱. لیست دنبال‌کنندگان یا دنبال‌شوندگان 35](#_Toc103948180)

[۳-۴-۲. لیست کامنت‌ها 35](#_Toc103948181)

[۳-۴-۳. لیست نقدها 36](#_Toc103948182)

[۳-۴-۴. لیست تماشا 36](#_Toc103948183)

[۳-۴-۵. لیست مورد علاقه 37](#_Toc103948184)

[۳-۵. صفحه اصلی 37](#_Toc103948185)

[۳-۶. ساید‌بار 38](#_Toc103948186)

[۳-۷. صفحه فیلم 39](#_Toc103948187)

[۳-۷-۱. لیست فیلم‌های مشابه 40](#_Toc103948188)

[۳-۷-۲. ثبت نقد 40](#_Toc103948189)

[۳-۷-۳. لیست نقدهای موجود 41](#_Toc103948190)

[۳-۸. صفحه نقد 41](#_Toc103948191)

[۳-۸-۱. ویرایش نقد 42](#_Toc103948192)

[۳-۸-۲. کامنت‌ها 42](#_Toc103948193)

[۳-۹. صفحه لیست فیلم‌ها 44](#_Toc103948194)

[۴. فهرست منابع 45](#_Toc103948195)

[برگ تأیید هیئت داوران/ صورت‌جلسۀ دفاع (به زبان انگلیسی) 46](#_Toc103948196)

۱. مقدمه

۱-۱. اهمیت موضوع

امروزه با افزایش بی‌رویه علاقه‌مندان به سینما، تماشاگران بی‌شماری با سلایق گوناگونی و متفاوتی وارد این حوزه شده‌اند. این تنوع طبع وسلیقه به صنعت فیلم‌سازی نیز وارده شده و هر ساله در حوزه سینما فیلم‌های بیشماری در ژانرها و موضوعات مختلفی ساخته و منتشر می‌شود که هر فیلم و هر ژانر تماشاگر خاص خود را دارد. این تنوع و تعدد گزینه‌ها و ژانرها و سبک‌های مختلف، بر تماشاگران سینما نیز تاثیر گذاشته و پروسه‌ی یافتن فیلمی مناسب با طبع و سلیقه خود را تبدیل به کار دشواری کرده است.

در این پروژه قصد داریم تا با طراحی یک نرم‌افزار هوشمند، به تسهیل این پروسه برای کاربران کمک کنیم. بدین منظور با استفاده از الگوریتم‌های فیلترینگ یک سیستم پیشنهاد‌دهی طراحی و با استفاده از آن فیلم‌هایی مناسب طبع کاربر به او پیشنهاد و همچنین کاربران با سلیقه مشابه را به یکدیگر معرفی کنیم.

۱-۲. تشریح پروژه

هدف در این پروژه، طراحی یک شبکه اجتماعی سینما محور شامل یک سیستم پیشنهاددهی است. در این برنامه کاربران قادر خواهند بود تا علاوه بر تعامل با یکدیگر، به نقدوبررسی و امتیازدهی به فیلم‌های سینمایی مختلف بپردازند. همچنین سیستم پیشنهاددهی با به کارگیری امتیازات داده شده توسط هر کاربر، توانایی این را دارد که امتیاز فیلم‌های دیده نشده توسط کاربر را تخمین زده، فیلم‌های مناسب و مطابق سلیقه کاربر به او پیشنهاد داده و کاربران با سلایق مشترک را به یکدیگر معرفی نماید.

امکانات پیش‌بینی شده در این نرم‌افزار به شرح زیر می‌باشد:

* ثبت امتیاز (۱ تا ۱۰) و نقد و بررسی کاربر برای یک فیلم
* جست‌و‌جو و یافتن اطلاعات یک فیلم
* معرفی لیستی از کاربران مشابه و هم‌سلیقه با کاربر به او
* معرفی لیستی از فیلم‌های مشابه به یک فیلم
* دنبال‌کردن کاربران مورد نظر
* دریافت فهرست پیشنهادات و تخمین امتیاز فیلم بر اساس کاربران دنبال شده
* دریافت فهرست پیشنهادات و تخمین امتیاز فیلم بر اساس امتیازات شخصی کاربر
* مشاهده نقدها و امتیازات ثبت شده از کاربران
* قابلیت کامنت‌گذاری کاربران در صفحه نقد ثبت‌شده‌ی یک کاربر
* قابلیت لایک کردن نقدها و کامنت‌های کاربران

۱-۳. توصیف رویکرد پیشنهاددهی

یکی از بدیهی‌ترین شیوه‌های ارزیابی و دریافت پیشنهادات در هر حوزه‌ای اعم از سینما، کتاب و ... ، پرس‌وجو و درخواست پیشنهاددهی از دوستان یا افراد مورد اعتماد که دارای سلیقه مشابه با شما هستند، می‌باشد. طبیعتا این تعداد محدود از افراد از تمامی گزینه‌های موجود آگاه نبوده و قادر نخواهند بود تا پیشنهادات متنوعی را ارائه دهند. بدین منظور مجموعه‌ای از تکنیک‌ها به نام پالایش گروهی (collaborative filtering) توسعه داده شده‌اند.

یک الگوریتم پالایش گروهی معمولا با جستجو در گروه بزرگی از آدم‌ها و یافتن گروهی کوچکتر از آدم‌ها با سلیقه مشابه کار می‌کند. این الگوریتم با مبنا قرار دادن آیتم‌هایی که اعضای این گروه پسندیده‌اند، یک لیست مرتب از پیشنهادات را جمع‌آوری و ارائه می‌کند.

۱-۳-۱. دسته‌بندی افراد

به منظور شناسایی و دسته‌بندی گروهی آدم‌ها بر اساس سلایقشان، به یک الگوریتم برای مقایسه‌ی دو به دوی آن‌ها و تخصیص یک امتیاز تشابه (similarity score) بین این دو فرد نیاز داریم. الگوریتم‌های متنوعی در این زمینه وجود دارند که یکی از ساده‌ترین آن‌ها الگوریتم فاصله اقلیدسی (Euclidean distance) می‌باشد.

در ریاضیات، فاصله اقلیدسی بین دو نقطه در فضای اقلیدسی طول یک قطعه خطی بین دو نقطه است. برای محاسبه امتیاز تشابه بین دو فرد، آیتم‌های مشابه‌ای را که این دو فرد به آن‌ها امتیاز داده‌اند را جمع‌آوری و از آن‌ها به عنوان محور مختصاتمان استفاده می‌کنیم. سپس با محاسبه فاصله بین این دو نقطه، امتیاز تشابه این فرد به دست آورده می‌شود.

۱-۳-۲. چالش موجود

با وجود ساده و کارا بودن الگوریتم فاصله اقلیدسی، به دلیل مشکل تورم نمره‌ای (grade inflation) که ذاتا در محیط ما وجود دارد، نمی‌توانیم با استفاده از آن امتیازات دقیق و قابل‌اعتمادی را به دست بیاوریم. این مشکل ناشی از تفاوت ذاتی کاربران در استفاده از رنج امتیازی موجود می‌باشد و باعث غیرنرمال شدن اطلاعات می‌شود.

فرض کنید علی و اکبر دو فرد مورد نظر ما هستند و ما می‌خواهیم امتیاز تشابه بین آن‌ها را محاسبه کنیم. علی فیلم A را پسندیده و به آن امتیاز ۹ را می‌دهد. اما از فیلم B راضی نبوده و به آن امتیاز ۴ را می‌دهد. عباس نیز مانند علی فیلم A را پسندیده اما به آن امتیاز ۷ را می‌دهد و فیلم B دوست نداشته و به آن ۱ امتیاز می‌دهد. در تئوری عباس و علی سلیقه مشابهی داشته و اما با استفاده از الگوریتم فاصله اقلیدسی، امتیاز تشابه پایینی برای آن‌ها به دست خواهند آمد.

۱-۳-۳. راهکار

در این پروژه برای رفع این مشکل و نرمال‌سازی اطلاعات امتیازی کاربران از ضریب همبستگی پیرسون (Pearson correlation coefficient) استفاده خواهیم کرد. این الگوریتم همبستگی دو مجموعه از اطلاعات را بر روی یک خط راست بررسی کرده و به این همبستگی یک عدد بین (-۱ تا ۱) اختصاص می‌دهد که ۱ به معنای تشابه کامل و -۱ به معنی عدم‌ تشابه کامل است.

با استفاده از الگوریتم پیرسون خواهیم توانست تا افراد را بر اساس سلایقشان دسته‌بندی و با به‌کارگیری الگوریتم‌های فیلترینگ آیتم‌های مناسب را به آن‌ها پیشنهاد دهیم. در این پروژه از دو نوع الگوریتم فیلترینگ استفاده خواهیم کرد: ۱. فیلترینگ کاربر محور (user-based filtering) ۲. فیلترینگ آیتم محور (item-based filtering)

۱-۳-۴. انواع فیلترینگ

در فیلترینگ کاربر محور، پس از محاسبه امتیازات تشابه کاربران و دسته‌بندی آن‌ها در گروه‌های با سلیقه مشترک، با به‌کارگیری امتیازات داده شده به هر آیتم توسط هم‌گروهی‌های هر کاربر، امتیاز احتمالی آن کاربر برای آن آیتم را تخمین زده و یا با محاسبه امتیاز تخمینی بر تمام آیتم‌ها، آیتم‌های برتر را جمع‌آوری و به کاربر پیشنهاد می‌دهیم.

در فیلترینگ آیتم محور، به جای تخصیص امتیاز تشابه بین کاربران، این پروسه را به ازای هر آیتم موجود و بر اساس امتیازاتی که کاربران مختلف به این آیتم‌ها داده‌اند، انجام می‌دهیم. بدین منظور و برای تخصیص امتیاز تشابه بین آیتم A و آیتم B امتیازاتی که هر کاربر به این دو آیتم داده را جمع‌آوری کرده و با مقایسه این امتیازات، یکی امتیاز تشابه به این دو آیتم تخصیص می‌دهیم. پس از تخصیص امتیاز بین تمامی آیتم‌ها، خواهیم توانست تا امتیاز احتمالی یک کاربر به آن آیتم را تخمین زده و یا آیتم‌های مشابه به این آیتم را به کاربر معرفی نماییم.

فیلترینگ آیتم‌محور ذاتا و به علت کمتر بودن اطلاعات مورد نیاز، علی رغم نتایج یکسان اما بسیار سریع‌تر از فیلترینگ کاربر محور می‌باشد. همچنین در تئوری این نوع فیلترینگ نتایج با ثبات‌تری نسبت به فیلترینگ کاربرمحور به ما خواهد داد.

منطقا می‌توان تنها با استفاده از فیلترینگ آیتم‌محور، پروسه پیشنهاددهی را با سرعت بیشتر و ثبات بهتر انجام داد. اما به دلیل نیاز به دسته‌بندی و معرفی کاربران مشابه و هم‌سلیقه به یکدیگر، ما نیاز داریم تا از فیلترینگ کاربرمحور نیز در این پروژه استفاده کنیم.

۱-۳-۵. نحوه استفاده در پروژه

در این پروژه با استفاده از الگوریتم پیرسون، امتیاز تشابه بین کاربران و فیلم‌های مختلف را محاسبه می‌کنیم. سپس یک سیستم پیشنهاددهنده شامل دو بخش زیر پیاده‌سازی می‌کنیم: ۱. پیشنهاددهنده عمومی (global recommender) ۲. پیشنهاددهنده دوستان (friends recommender)

بخش پیشنهاددهنده عمومی بر اساس فیلترینگ نوع کاربرمحور و با استفاده امتیازات ثبت‌شده توسط خود کاربر، سعی می‌کند تا امتیاز احتمالی یک فیلم را تخمین زده و لیستی از فیلم‌های مناسب را به کاربر پیشنهاد دهد.

بخش پیشنهاددهنده دوستان زیر بر اساس فیلترینگ آیتم‌محور پیاده‌سازی شده و با استفاده از امتیازات ثبت شده توسط کاربران دنبال شده توسط کاربر، سعی می‌کند تا امتیاز احتمالی یک فیلم را تخمین زده و لیستی از فیلم‌های مناسب را به کاربر پیشنهاد دهد.

۲. پیاده‌سازی

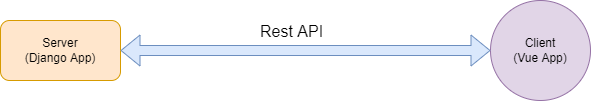
در این فصل به بررسی و تشریح مدل نرم‌افزاری، معماری سیستم و فریم‌ورک‌های استفاده شده می‌پردازیم.

۲-۱. معماری نرم‌افزار

امروزه شیوه مرسوم طراحی و پیاده‌سازی نرم‌افزارهای کاربردی، معماری کلاینت-سرور (client-server) می‌باشد. در این پروژه نیز همانگونه که در تصویر ۱-۱ نشان داده شده است با استفاده از همین معماری، نرم‌افزار خود را به دو بخش سرور و کلاینت تقسیم، و این دو بخش را با استفاده از یک واسط به نام API به یکدیگر متصل می‌کنیم.

در این پروژه، بخش سرور را با استفاده از فریم‌ورک جنگو (Django) پیاده‌سازی می‌کنیم. جنگو یک ابزار ساخته شده به زبان پایتون (Python) است که برای پیاده‌سازی برنامه‌های وب استفاده می‌شود.

همچنین بخش کلاینت به عنوان یک وب اپ (Web App) و با استفاده از فریم‌ورک VueJs پیاده‌سازی می‌شود. Vue یک فریم‌ورک به زبان جاواسکریپ (JavaScript) است که به منظور ساخت رابط‌های کاربری تحت وب استفاده می‌شود.



تصویر ۱-۱: معماری کلاینت-سرور

۲-۲. ‌Backend (سرور)

بخش سرور وظیفه پاسخ‌دهی به درخواست‌های کاربران، دریافت و ارسال اطلاعات مورد نیاز از پایگاه داده و همچنین جمع‌آوری و ارسال اطلاعات تولید شده توسط سیستم پیشنهاددهنده را دارد.

همچنین در این بخش با به‌کارگیری برنامه Celery در کنار جنگو، پروسه تولید اطلاعات مورد نیاز سیستم پیشنهاددهنده را انجام می‌دهیم.

۲-۲-۱. مدل داده‌ها

در این پروژه از پایگاه داده postgresql برای ثبت و بازیابی اطلاعات استفاده می‌شود. مدل داده‌های موجود در این پایگاه داده در تصویر ۲-۲ نشان داده شده و موجودیت‌های آن به شرح زیر است:

User: این موجودیت برای ذخیره اطلاعات کاربران استفاده می‌شود. صفات آن به شرح زیر است:

Id: شناسه مربوط به کاربر

Gender: جنسیت کاربر

Avatar: تصویر پروفایل کاربر

Friends: لیست دوستان کاربر

Watchlist: لیست فیلم‌های ذخیره شده برای تماشا

Liked: لیست فیلم‌های محبوب

Movie: این موجودیت برای ذخیره اطلاعات فیلم‌ها استفاده می‌شود. صفات آن به شرح زیر است:

Id: شناسه مربوط به فیلم

Title: عنوان فیلم

Year: سال ساخت فیلم

Genres: لیست ژانرهای فیلم

Imdbid: آی‌دی پروفایل فیلم در سایت imdb

Tmdbid: آی‌دی پروفایل فیلم در سایت tmdb

Plot: خلاصه‌ای از داستان فیلم

Poster: آدرس تصویر پوستر فیلم

Review: از این موجودیت برای ذخیره اطلاعات نقدهای کاربران استفاده می‌شود. صفات آن به شرح زیر است:

Id: شناسه مربوط به نقد

Userid: شناسه کاربری که نقد را نوشته

Movieid: شناسه فیلمی که برای آن نقد نوشته شده است

Rating: امتیازی که کاربر به این فیلم داده است

Text: متن نقد نوشته شده توسط کاربر

Comments: لیست کامنت‌های نقد

Likes: لیست کاربرانی که این نقد را لایک کرده‌اند

Created\_at: زمانی که نقد ساخته شده است

Updated\_at: آخرین زمانی که نقد آپدیت شده است

Comment: از این موجودیت برای ذخیره اطلاعات کامنت‌های کاربران استفاده می‌شود. صفات آن به شرح زیر است:

Userid: شناسه کاربری که این کامنت را نوشته است

Object: شی مربوط به موجودیتی که این کامنت برایش نوشته شده است (کامت یا نقد)

Text: متن کامنت

Comments: لیست کامنت‌هایی که برای این کامنت نوشته شده‌اند

Likes: لیست کاربرانی که این کامنت را لایک کرده‌اند

Created\_at: زمانی که این کامنت نوشته شده است

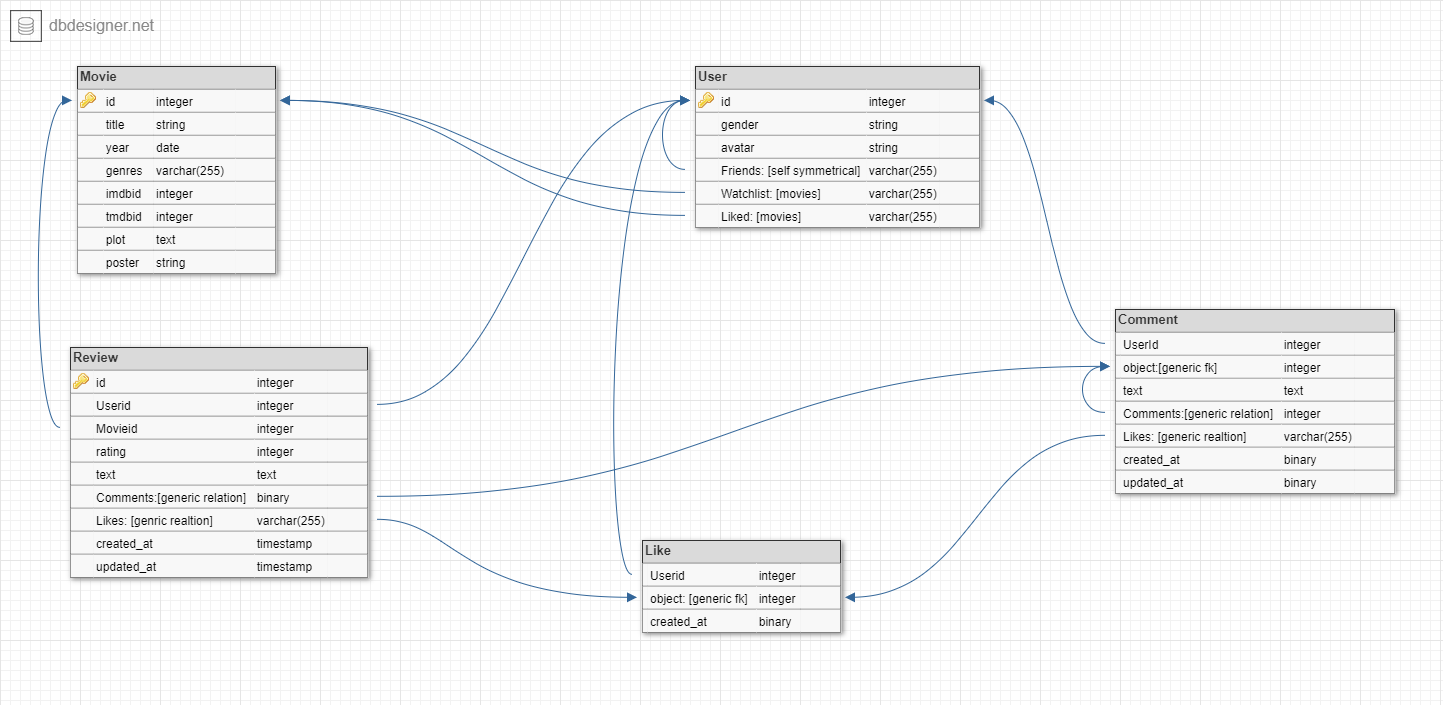
Updated\_at: آخرین زمانی که این کامنت آپدیت شده است

Like: از این موجودیت برای ذخیره اطلاعات لایک‌های کاربران استفاده می‌شود. صفات آن به شرح زیر است:

Userid: شناسه کاربری که لایک کرده است

Object: شی مربوط به موجودیتی که لایک شده است (فیلم، نقد یا کامنت)

Created\_at: زمانی که این لایک انجام شده است



تصویر ۲-۲: مدل داده‌های پایگاه داده

۲-۲-۲. Celery

در این پروژه از برنامه celery به منظور تولید متناوب و روزانه اطلاعات مورد نیاز سیستم پیشنهاددهنده استفاده می‌شود. Celery یک سیستم task manager بوده و در آن قادر خواهیم مجموعه‌ای از task های دوره‌ای را تعریف و آن‌ها را در بازه‌های زمانی مشخص(روزانه) در پس‌زمینه اجرا کنیم.

در این پروژه دو task برای محاسبه امتیازات تشابه و همچنین بروزرسانی اطلاعات مربوط به فیلم‌ها در فایل های `backend/recommender/tasks.py` و همچنین در `backend/app/tasks.py` تعریف می‌کنیم.

Task اول به صورت روزانه اجرا و امتیازات تشابه بین تمامی فیلم‌ها و همچنین بین تمامی کاربران را محاسبه کرده و در پایگاه داده ذخیره می‌کند.

Task دوم نیز به صورت روزانه اجرا، لیستی از فیلم‌ها با اطلاعات ناقص را جمع‌آوری و تلاش کرده تا این اطلاعات را با استفاده از پروفایل فیلم در سایت imdb تکمیل کند. سپس این اطلاعات را در پایگاه داده ذخیره می‌کند.

۲-۲-۳. سیستم پیشنهاددهنده

در این پروژه پس از محاسبه امتیازات تشابه بین کاربران و بین فیلم‌ها، با استفاده از دو نوع پیشنهاد‌دهنده خود به نام‌های عمومی و دوستان، اطلاعات مورد نیاز را تولید و با استفاده از API به سمت کلاینت ارسال خواهیم کرد.

۲-۲-۳-۱. امتیازات تشابه

همانگونه که در فصل پیش ذکر شد، در این پروژه برای محاسبه امتیازات تشابه از الگوریتم پیرسون استفاده خواهیم کرد. این الگوریتم با حل مشکل تورم نمره به ما کمک کرده تا نتایج دقیق‌تر و قابل‌اعتماد‌تری به دست آوریم.

برای محاسبه امتیاز تشابه دو فرد، از الگوریتم زیر استفاده می‌شود. در این الگوریتم Pi معادل با فرد اول و دوم، n معادل با تعداد کل فیلم‌های امتیاز داده شده مشترک بین فرد اول و دوم، و همچنین PR لیستی از امتیازات مشترک کاربر اول و دوم به فیلم‌ها می‌باشد. سایر مقادیر مورد نیاز به نحوه زیر محاسبه می‌شوند:

*در نهایت، مقدار عددی امتیاز تشابه بین فرد اول و دوم، از طریق فرمول زیر محاسبه می‌شود:*

*امتیاز تشابه r عددی در بازه (-۱ تا ۱) خواهد بود، که مقدار ۱ معادل با تشابه کامل و مقدار -۱ معادل با عدم‌تشابه کامل می‌باشد.*

*برای محاسبه امتیاز تشابه بین دو فیلم نیز از همین فرمول استفاده شده. با این تفاوت که مقدار Pi معادل است با فیلم اول و دوم، مقدار n تعداد افراد مشترکی که به هر دو فیلم امتیاز داده‌اند، و PR معادل با لیست امتیازاتی است که توسط افراد مشترک برای این دو فیلم ثبت شده است.*

۲-۲-۳-۲. پیشنهاددهنده عمومی

در این پیشنهاددهنده، محاسبات با استفاده از امتیازات تشابه بین فیلم و همچنین لیست امتیازاتی که کاربر برای فیلم‌ها ثبت کرده است انجام می‌شود. این پیشنهاددهنده اطلاعاتی مانند لیست کاربران مشابه، امتیاز تخمین‌زده شده، لیست فیلم‌های مشابه و همچنین لیستی از فیلم‌های پیشنهادی را به ازای هر کاربر تولید می‌کند.

۲-۲-۳-۲-۱ لیست کاربران مشابه

برای تولید لیست کاربران مشابه، کافیست لیست تمام امتیازات تشابهی که برای کاربر محاسبه شده است را از پایگاه داده جمع‌آوری کرده، این لیست را به ترتیب امتیاز تشابه با کاربر به صورت نزولی مرتب کرده، و ۱۰۰ نفر اول این لیست را جمع‌آوری و به کاربر ارسال کنیم. این لیست شامل اطلاعات هر کاربر به علاوه امتیاز تشابه او با کاربر فعلی خواهد بود.

۲-۲-۳-۲-۲. لیست فیلم‌های مشابه

برای تولید لیست فیلم‌های مشابه نیز مانند لیست کاربران مشابه، کافیست لیست تمام امتیازات تشابهی که برای یک فیلم محاسبه شده است را از پایگاه داده جمع‌آوری کرده، این لیست را به ترتیب امتیاز تشابه با فیلم به صورت نزولی مرتب کرده، و ۱۰ فیلم اول این لیست را جمع‌آوری و به کاربر ارسال کنیم. این لیست شامل اطلاعات هر فیلم به علاوه امتیاز تشابه آن نسبت به این فیلم خواهد بود.

۲-۲-۳-۲-۳. امتیاز تخمین‌زده شده

برای تخمین امتیاز یک فیلم، کافیست تمام امتیازات ثبت شده توسط کاربر به ازای هر فیلم را در امتیاز تشابه آن فیلم ضرب کرده و مجموع آن‌ها را تقسیم بر مجموع امتیازات تشابه آن‌ها کرد. فرمول اینکار به صورت زیر می‌باشد. در اینجا PR لیست امتیازاتی است که کاربر به ازای هر فیلم ثبت کرده و sim لیست امتیازات تشابه ‌آن فیلم‌ها نسبت به فیلم مذکور می‌باشد.

۲-۲-۳-۲-۴. لیست فیلم‌های پیشنهادی

برای تولید یک لیست از فیلم‌های پیشنهادی، بایستی به ازای تمامی فیلم‌های موجود در پایگاه داده، با استفاده از الگوریتم معرفی شده در بخش قبلی، امتیاز تخمینی آن فیلم را برای کاربر محاسبه کنیم. سپس این لیست به ترتیب امتیازات تخمین‌زده‌شده به صورت نزولی مرتب کرده و ۱۰ فیلم برتر این لیست را جمع‌آوری کرده و به کاربر ارسال کنیم. این لیست شامل اطلاعات هر فیلم به علاوه امتیاز تخمین‌زده‌شده آن فیلم برای کاربر خواهد بود.

۲-۲-۳-۳. پیشنهاددهنده دوستان

این پیشنهاددهنده اطلاعات خود را با استفاده از افراد حاضر در لیست دوستان کاربر ایجاد می‌کند. در این پیشنهادهنده محاسبات با استفاده از لیست امتیازات تشابه بین کاربر و دوستان، و همچنین لیست امتیازاتی که دوستان کاربر به ازای هر فیلم ثبت کرده‌اند، انجام می‌شود.

۲-۲-۳-۳-۱. امتیاز تخمین‌زده‌شده

برای تخمین امتیاز یک فیلم، کافیست امتیازاتی که دوستان کاربر برای این فیلم مذکور ثبت کرده‌اند را در امتیاز تشابه آن فرد به کاربر مورد نظر ضرب کرده و مجموع آن‌ها را تقسیم بر مجموع امتیازات تشابه دوستانی که آن فیلم را دیده‌اند، کرد. فرمول اینکار به صورت زیر می‌باشد. در اینجا PR لیست امتیازاتی است که دوستان کاربر برای این فیلم خاص ثبت کرده‌اند و sim لیست امتیازات تشابه بین کاربر و دوستان او می‌باشد.

۲-۲-۳-۳-۲. لیست فیلم‌های پیشنهادی

برای تولید یک لیست فیلم‌های پیشنهادی، بایستی به ازای تمامی فیلم‌های موجود در پایگاه داده، با استفاده از الگوریتم معرفی شده در بخش قبلی، امتیاز تخمینی آن فیلم را برای کاربر محاسبه کنیم. سپس این لیست به ترتیب امتیازات تخمین‌زده‌شده به صورت نزولی مرتب کرده و ۱۰ فیلم برتر این لیست را جمع‌آوری کرده و به کاربر ارسال کنیم. این لیست شامل اطلاعات هر فیلم به علاوه امتیاز تخمین‌زده‌شده آن فیلم برای کاربر خواهد بود.

۲-۲-۴. API

در برنامه‌های وب‌محور از API به عنوان واسطی برای تبادل اطلاعات بین دو برنامه استفاده می‌شود. در این پروژه واسط API بر مبنای معماری REST و با استفاده از کتابخانه Django Rest Framework (DRF) پیاده‌سازی شده است.

مسیر اصلی دسترسی به API پروژه به صورت "localhost/api/" می‌باشد. ساختار سایر زیرمسیرهای پیاده‌سازی برای این پروژه به شرح زیر می‌باشد:

۲-۲-۴-۱. اعتبارسنجی

به منظور اعتبارسنجی کاربران از این مسیر "api/auth/" استفاده می‌شود.

۲-۲-۴-۱-۱. لاگین

با ارسال یک درخواست POST به مسیر "auth/login/" کاربران می‌توانند لاگین کرده و توکن خود را دریافت کنند. سپس با استفاده از این توکن کاربران می‌توانند درخواست‌های خود را اعتبارسنجی کنند.

۲-۲-۴-۱-۲. لاگ‌آوت

با ارسال یک درخواست POST به مسیر "auth/logout" کاربران می‌توانند از سیستم خارج شده و توکن آن‌ها حذف می‌شود.

۲-۲-۴-۱-۳. ثبت‌نام

با ارسال یک درخواست POST شامل اطلاعات مورد نیاز به مسیر "auth/register/" کاربران می‌توانند در سیستم ثبت‌نام کنند.

۲-۲-۴-۲. کامنت‌ها

به منظور دریافت و ارسال اطلاعات مورد نیاز کامنت از مسیر "api/comments/" استفاده می‌شود.

۲-۲-۴-۲-۱. ویرایش

با ارسال یک درخواست PATCH به مسیر "comments/<id>/" کاربران می‌توانند کامنت‌های خود را ویرایش کنند. هر کاربر فقط کامنت خود را می‌تواند ویرایش کند.

۲-۲-۴-۲-۲. حذف

با ارسال یک درخواست DELETE به مسیر "comments/<id>/" کاربران می‌توانند کامنت‌های خود را حذف کنند.

۲-۲-۴-۲-۳. لایک‌ها

با ارسال یک درخواست GET به مسیر "comments/<id>/likes/ لیستی از لایک‌کنندگان یه کامنت دریافت می‌شود.

۲-۲-۴-۲-۴. کامنت‌ها

با ارسال یک درخواست GET به مسیر "comments/<id>/comments/" لیستی از کامنت‌های مربوط به یک کامنت دریافت می‌شوند.

۲-۲-۴-۳. لایک‌ها

به منظور دریافت و ارسال اطلاعات مورد نیاز لایک‌ها از مسیر "api/likes/ /" استفاده می‌شود.

۲-۲-۴-۳-۱. حذف

با ارسال یک درخواست DELETE به مسیر "likes /<id>/" کاربران می‌توانند لایک‌های خود را حذف کنند.

۲-۲-۴-۴. فیلم‌ها

به منظور دریافت اطلاعات مورد نیاز فیلم‌ها از مسیر "api/movies/" استفاده می‌شود.

۲-۲-۴-۴-۱. بازیابی

با ارسال یک درخواست GET به مسیر "movies/<id>/" اطلاعات کامل یک فیلم دریافت خواهد شد.

۲-۲-۴-۴-۲. فیلترینگ

با ارسال یک درخواست GET به فرمت "movies/?genres=comedy&year=1995" کاربر می‌تواند نسبت به فیلتر کردن نتایج بر حسب ژانر یا سال تولید اقدام کند.

۲-۲-۴-۴-۳. جستجو

با ارسال یک درخواست GET به فرمت "movies/?search=toy%20story" کاربر می‌تواند در لیست عناوین فیلم‌ها جستجو کند.

۲-۲-۴-۴-۴. لیست تماشا

با ارسال یک درخواست GET به مسیر "movies/<id>/watchlist/" لیست کاربرانی که این فیلم را در لیست تماشای خود قرار داده‌اند دریافت می‌شود.

۲-۲-۴-۴-۵. لایک‌ها

با ارسال یک درخواست GET به مسیر "movies/<id>/likes /" لیست کاربرانی که این فیلم را لایک کرده‌اند دریافت می‌شود.

۲-۲-۴-۵. نقدها

به منظور دریافت و ارسال اطلاعات مورد نیاز نقدها از مسیر "api/reviews/" استفاده می‌شود.

۲-۲-۴-۵-۱. ایجاد

با ارسال یک درخواست POST به مسیر "reviews/" حاوی اطلاعات مورد نیاز، کاربر می‌تواند یک نقد ایجاد کند.

۲-۲-۴-۵-۲. ویرایش

با ارسال یک درخواست PATCH به مسیر "reviews/<id>/" کاربر می‌تواند نقد خود را ویرایش کند. هر کاربر فقط نقدهای خود را می‌تواند ویرایش کند.

۲-۲-۴-۵-۳. حذف

با ارسال یک درخواست DELETE به مسیر "reviews/<id>/" کاربر می‌تواند نقد خود را حذف کند. هر کاربر فقط نقدهای خود را می‌تواند حذف کند.

۲-۲-۴-۵-۴. نقدهای اخیر

با ارسال یک درخواست GET به مسیر "reviews/recent/" لیستی ۱۰ تایی از آخرین نقدهای ایجاد شده در سیستم دریافت خواهد شد.

۲-۲-۴-۵-۵. نقدهای اخیر دوستان

با ارسال یک درخواست GET به مسیر "reviews/friends/" لیستی ۱۰ تایی از آخرین نقدهای ایجاد شده توسط دوستان کاربر در سیستم دریافت خواهد شد.

۲-۲-۴-۵-۶. لایک‌ها

با ارسال یک درخواست GET به مسیر "reviews/<id>/likes/" لیستی از کاربرانی که این نقد را لایک کرده‌اند دریافت خواهد شد.

۲-۲-۴-۵-۷. کامنت‌ها

با ارسال یک درخواست GET به مسیر "reviews/<id>/comments/" لیستی از کامنت‌هایی که برای این نقد ایجاد شده‌اند دریافت خواهد شد.

۲-۲-۴-۶. کاربران

به منظور دریافت و ارسال اطلاعات کاربران از مسیر "api/users/" استفاده می‌شود.

۲-۲-۴-۶-۱. بازیابی

با ارسال یک درخواست GET به مسیر "users/<id>/" اطلاعات یک کاربر دریافت می‌شود.

۲-۲-۴-۶-۲. ویرایش

با ارسال یک درخواست PATCH به مسیر "users/<id>/" کاربر می‌تواند اطلاعات خود را ویرایش کند. هر کاربر فقط اطلاعات خود را می‌تواند ویرایش کند.

۲-۲-۴-۶-۳. دنبال‌شوندگان

با ارسال یک درخواست PATCH به مسیر "users/<id>/followings/" لیستی از دنبال‌شوندگان توسط کاربر دریافت می‌شود.

۲-۲-۴-۶-۴. دنبال‌کنندگان

با ارسال یک درخواست PATCH به مسیر "users/<id>/followers/" لیستی از دنبال‌کنددگان کاربر دریافت می‌شود.

۲-۲-۴-۶-۵. isFollowing

با ارسال یک درخواست GET به مسیر "users/<id>/isfollowing/" وضعیت دنبال شدن یا نشدن این کاربر دریافت می‌شود.

۲-۲-۴-۶-۶. IsFollows

با ارسال یک درخواست GET به مسیر "users/<id>/isfollows/" وضعیت دنبال کردن یا نکردن این کاربر دریافت می‌شود.

۲-۲-۴-۶-۷. follow

با ارسال یک درخواست POST به مسیر "users/<id>/follow/" کاربر می‌تواند این کاربر را follow کند.

۲-۲-۴-۶-۸. unfollow

با ارسال یک درخواست POST به مسیر "users/<id>/unfollow/" کاربر می‌تواند این کاربر را unfollow کند.

۲-۲-۴-۶-۹. لیست تماشا

با ارسال یک درخواست GET به مسیر "users/<id>/watchlist/" فهرست لیست تماشای این کاربر دریافت می‌شود.

۲-۲-۴-۶-۱۰. لیست مورد علاقه

با ارسال یک درخواست GET به مسیر "users/<id>/favorites/" فهرست لیست فیلم‌های محبوب این کاربر دریافت می‌شود.

۲-۲-۴-۶-۱۱. اطلاعات اضافه

با ارسال یک درخواست GET به مسیر "users/<id>/extras/" اطلاعات اضافی یه کاربر از قبیل تعداد دنبال‌کنندگان، تعداد دنبال‌شوندگان، تعداد لیست تماشا و تعداد لیست مورد علاقه ارسال می‌شود.

۲-۲-۴-۷. پیشنهاددهنده دوستان

به منظور دریافت اطلاعات از پیشنهادهنده دوستان از مسیر "api/rc/friends/" استفاده می‌شود.

۲-۲-۴-۷-۱. لیست پیشنهادات

با ارسال یک درخواست GET به مسیر "friends/recommend/" لیستی ۱۰ تایی از پیشنهادات دریافت خواهد شد.

۲-۲-۴-۸. پیشنهاددهنده عمومی

به منظور دریافت اطلاعات از پیشنهاددهنده عمومی از مسیر "api/rc/global/" استفاده می‌شود.

۲-۲-۴-۸-۱. لیست پیشنهادات

با ارسال یک درخواست GET به مسیر "global/recommend/" لیستی ۱۰ تایی از پیشنهادات دریافت خواهد شد.

۲-۲-۴-۸-۲. لیست کاربران مشابه

با ارسال یک درخواست GET به مسیر "global/similar-users/" لیستی ۱۰۰ تایی از کاربران مشابه دریافت خواهد شد.

۲-۲-۴-۸-۳. لیست فیلم‌های مشابه

با ارسال یک درخواست GET به مسیر "global/similar-movies/<id>/" لیستی ۱۰ تایی از فیلم‌های مشابه فیلم مذکور دریافت خواهد شد.

۲-۳. Frontend (client app)

از بخش کلاینت به منظور فراهم آوردن بستری برای تعامل کاربر با سیستم استفاده می‌شود. در این پروژه بخش کلاینت به عنوان یک برنامه تحت وب (web app) و با استفاده از فریم‌ورک VueJs پیاده‌سازی می‌شود. Vue یکی از فریم‌ورک‌های محبوب جاواسکریپ بوده که به منظور توسعه رابط‌های کاربری تحت وب می‌باشد.

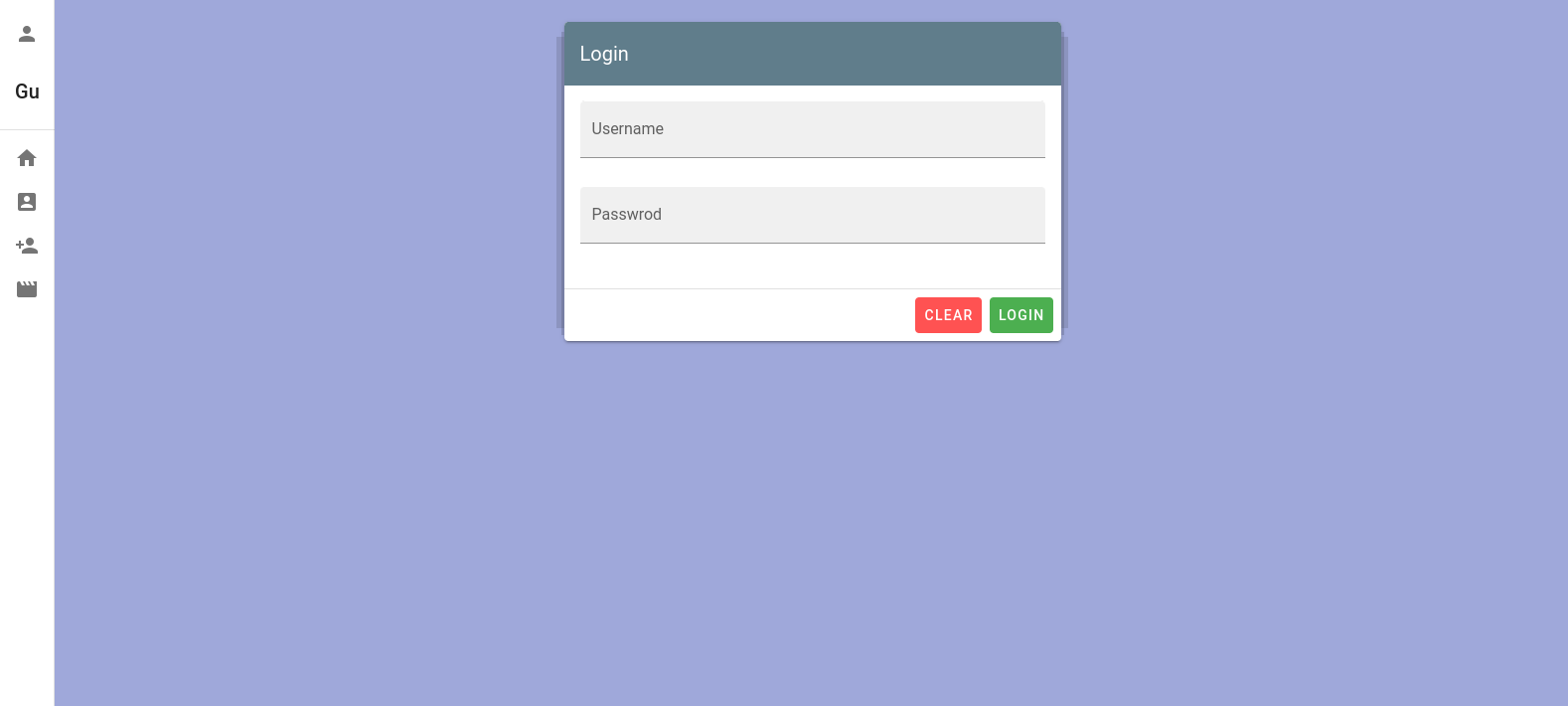
در این پروژه علاوه بر فریم‌ورک استاندارد Vue از سایر کتاب‌خانه‌های موجود به منظور فراهم‌سازی امکانات لازم استفاده شده است. این کتاب‌خانه‌ها به شرح زیر می‌باشند:

* Vuetify: یک UI Library که با فراهم کردن کامپوننت‌های آماده تحت دیزاین Material Design پروسه ساخت رابط کاربری را تسهیل می‌کند.
* Vue-router: از این کتابخانه به منظور مسیریابی (Routing) و نمایش صفحات مختلف سایت استفاده می‌شود.
* Vuex: کتابخانه رسمی Vue که به منظور مدیریت وضعیت‌های سیستم (State Manegment) و ذخیره‌سازی وضعیت تعاملات کاربر استفاده می‌شود.

۳. شرح عملکرد

در این فصل به تشریح و نمایش امکانات مخلتف پیاده‌سازی شده در بخش کلاینت خواهیم پرداخت. بخش کلاینت این پروژه یک وب اپ پیاده‌سازی شده بر پایه فریم‌ورک VueJs می‌باشد.

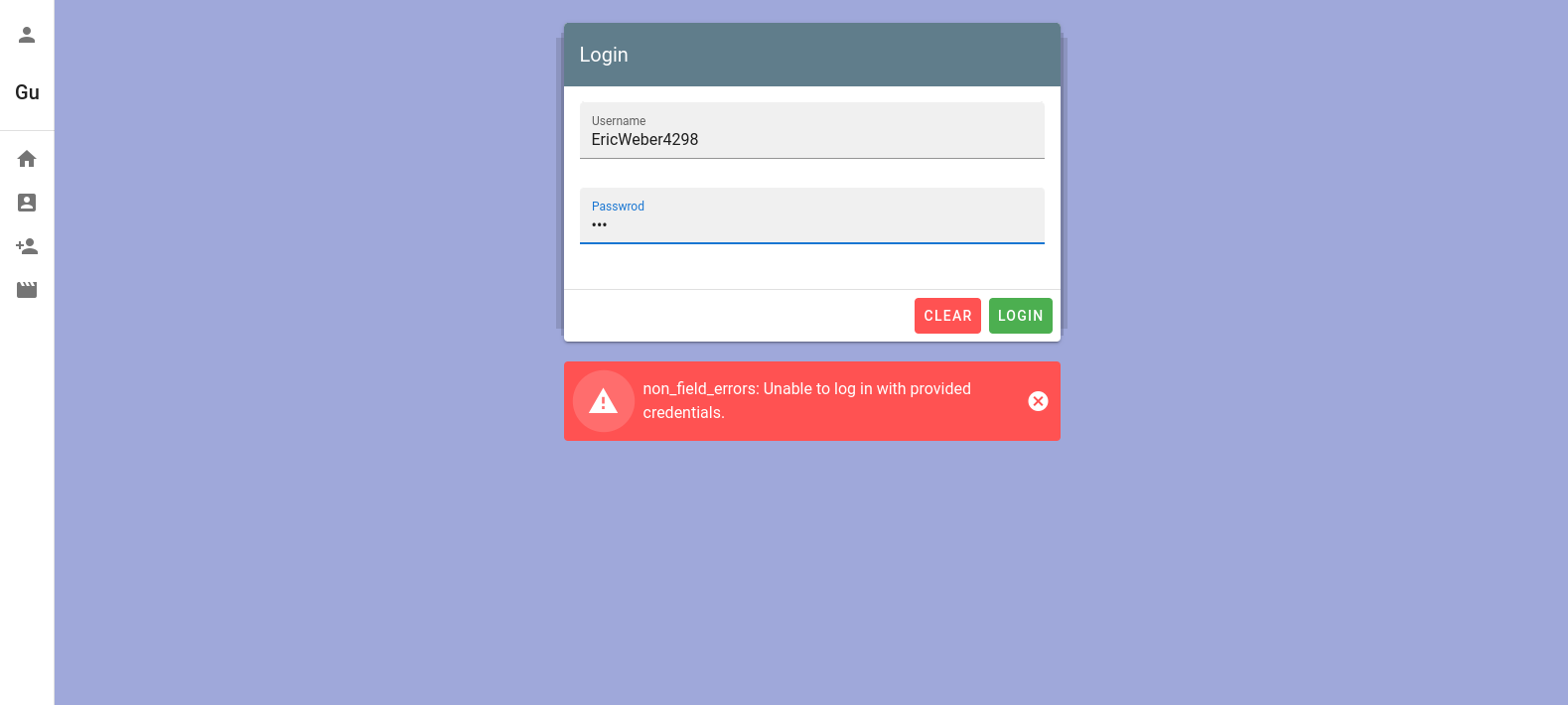
۳-۱. لاگین

در این صفحه کاربر قادر خواهد بود تا با وارد کردن مشخصات خود، وارد سیستم شده و از امکانات سایت استفاده کند. شمایل صفحه لاگین در تصویر ۱-۲ نشان داده شده است.

تصویر ۱-۲: صفحه لاگین

۳-۱-۱. ورود ناموفق

در صورت اشتباه وارد کردن مشخصات و ورود ناموفق، خطاهای مربوطه به کاربر نمایش داده خواهد شد. نمونه‌ای از این خطاها در تصویر ۲-۲ نشان داده شده است.

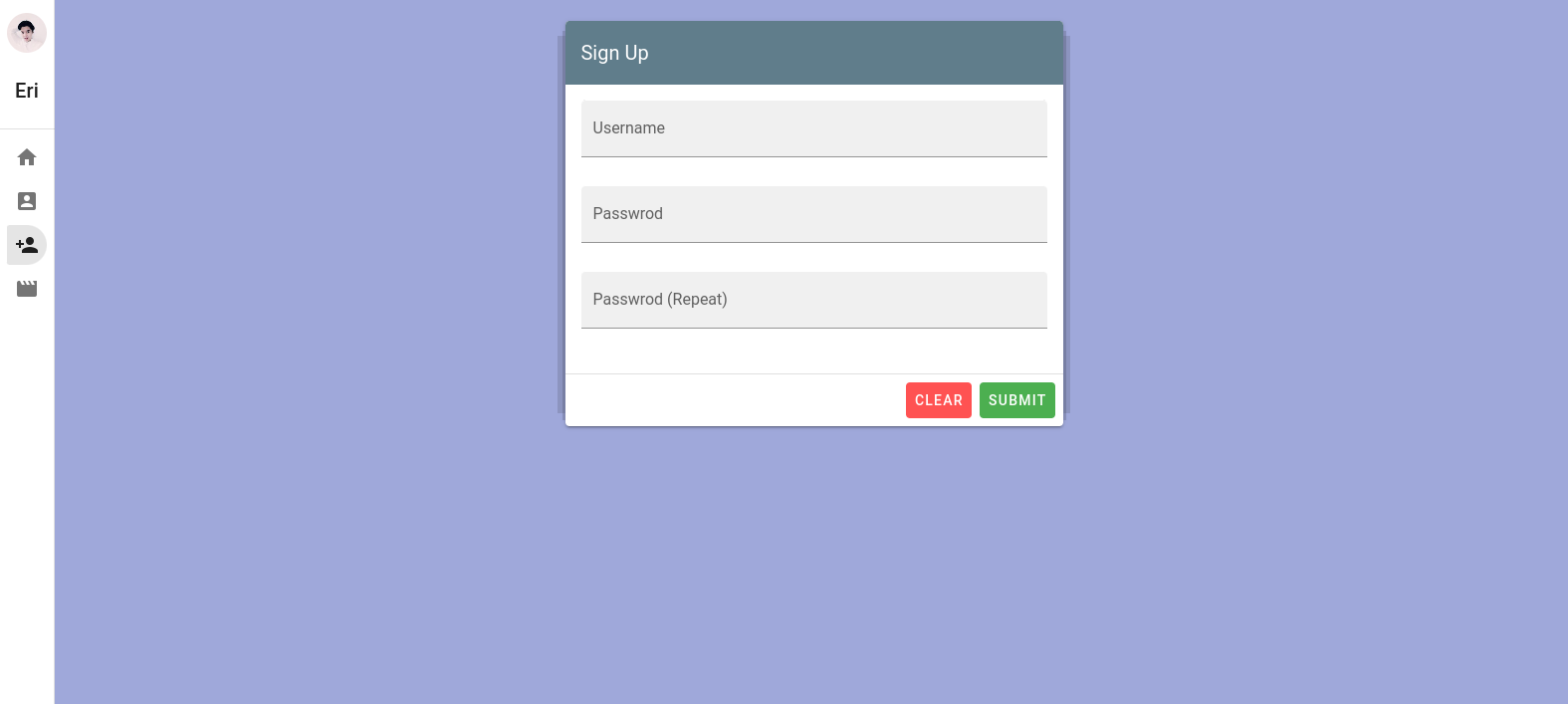


تصویر ۲-۲: لاگین ناموفق

۳-۱-۲. ورود موفق

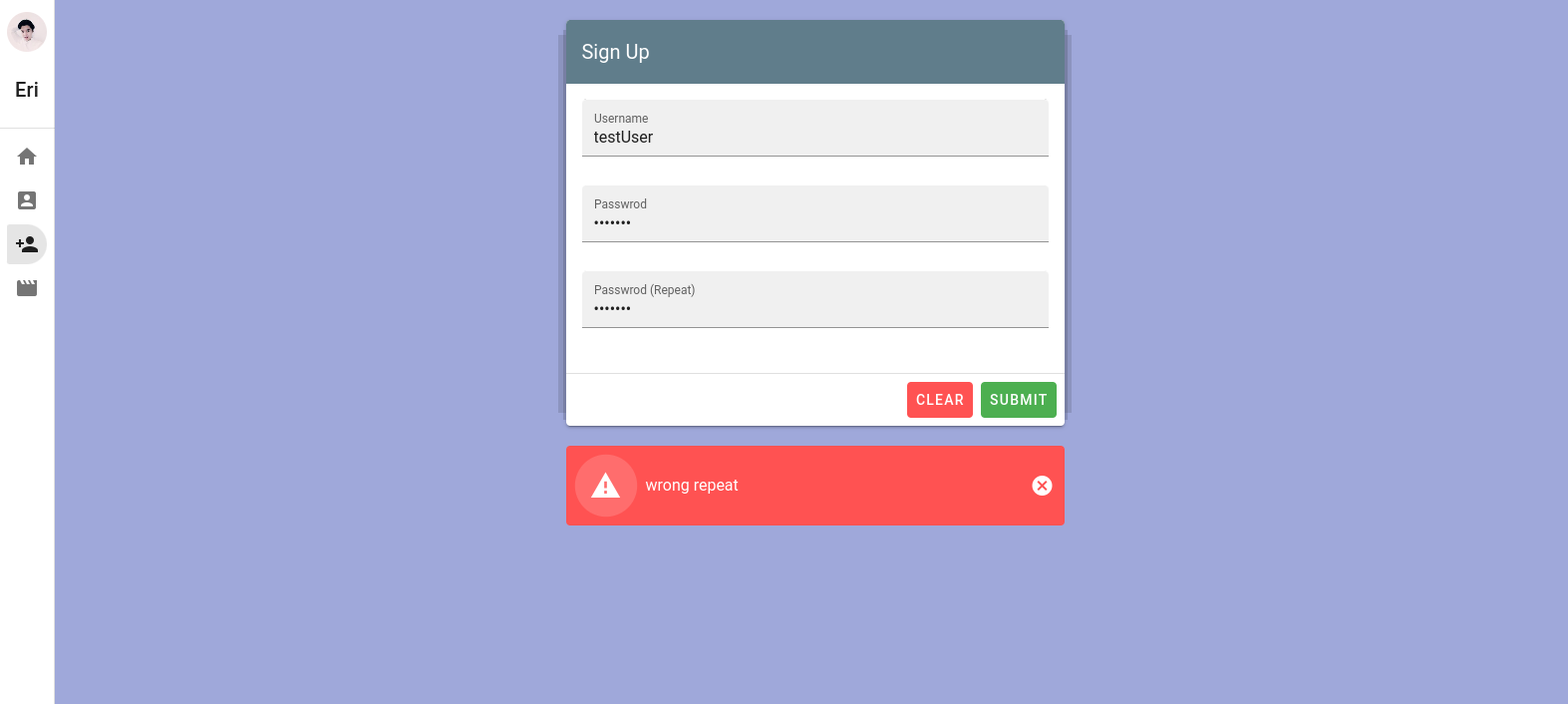
در صورت ورود موفق، کاربر به صفحه داشبورد خود منتقل خواهد شد.

۳-۲. عضویت

در این صفحه کاربر قادر خواهد بود تا با فراهم کردن اطلاعات مورد نیاز در سیستم عضو شود. شمایل صفحه عضویت کاربران در تصویر ۳-۲ نمایش داده شده است.

تصویر ۳-۲: صفحه عضویت کاربران

۳-۲-۱. عضویت ناموفق

در صورت وارد کردن اطلاعات ناصحیح، سیستم فرآیند عضویت را متوقف کرده و خطاهای مربوطه را به کاربر نمایش خواهد داد. نمونه‌ای از این خطاها در تصویر ۴-۲ نمایش داده شده است.

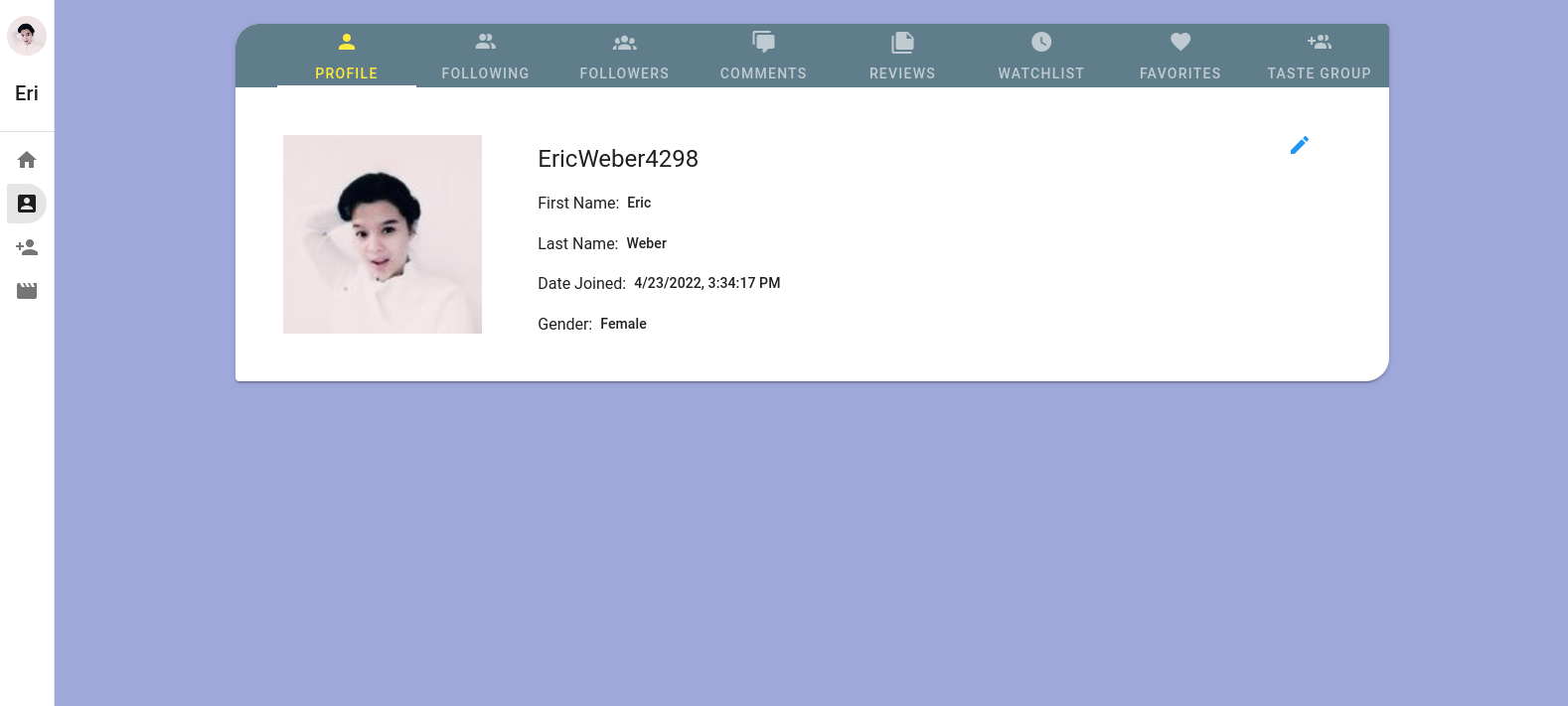
تصویر ۴-۲: عضویت ناموفق

۳-۲-۲. عضویت موفق

در صورت وارد کردن اطلاعات صحیح و فاقد مشکل، عضویت کاربر با موفقیت انجام شده و کاربر به صفحه داشبورد خود منتقل خواهد شد.

۳-۳. داشبورد

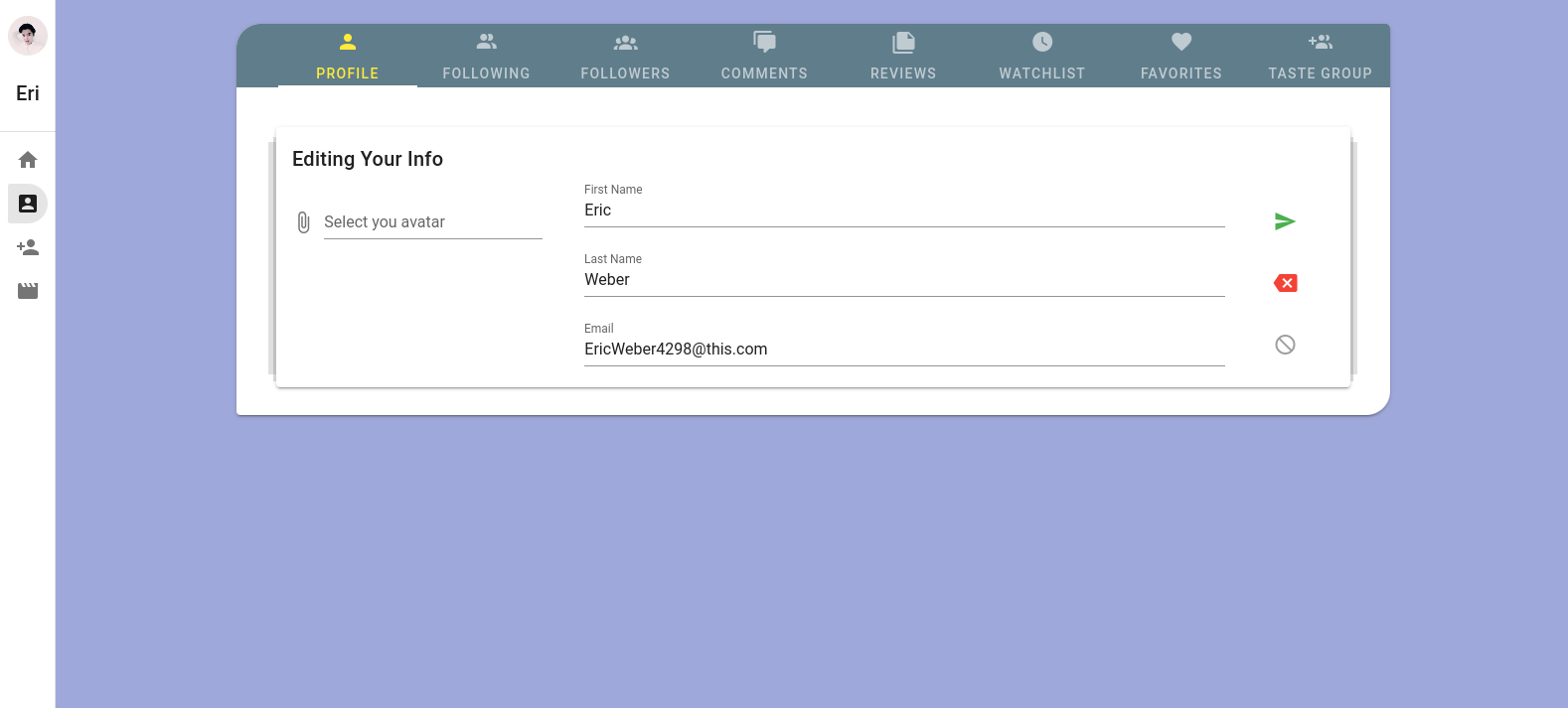
در این صفحه کاربر قادر خواهد بود تا مشخصات خود را مشاهده و نسبت به ویرایش آن‌ها اقدام کند. شمایل این صفحه در تصویر ۵-۲ نمایش داده شده است.



تصویر ۵-۲: صفحه داشبورد کاربر

۳-۳-۱. ویرایش اطلاعات

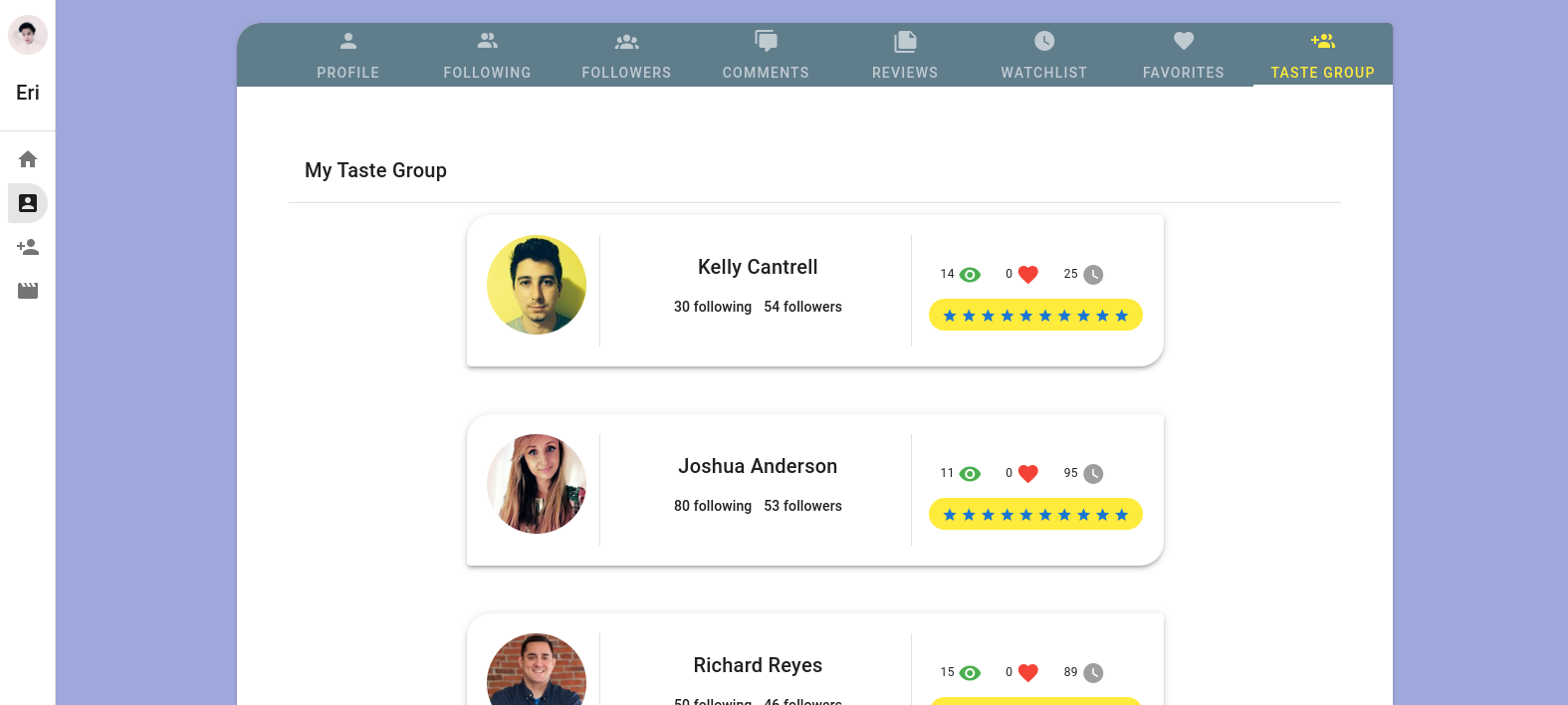
در پنل ویرایش اطلاعات کاربر قادر خواهد بود تا اطلاعاتی مانند نام و نام خانوادگی، ایمیل و تصویر پروفایل خود را ویرایش کند. این پنل در تصویر ۶-۲ نمایش داده شده است.



تصویر ۶-۲: پنل ویرایش اطلاعات کاربر در صفحه داشبورد

۳-۳-۲. لیست کاربران مشابه

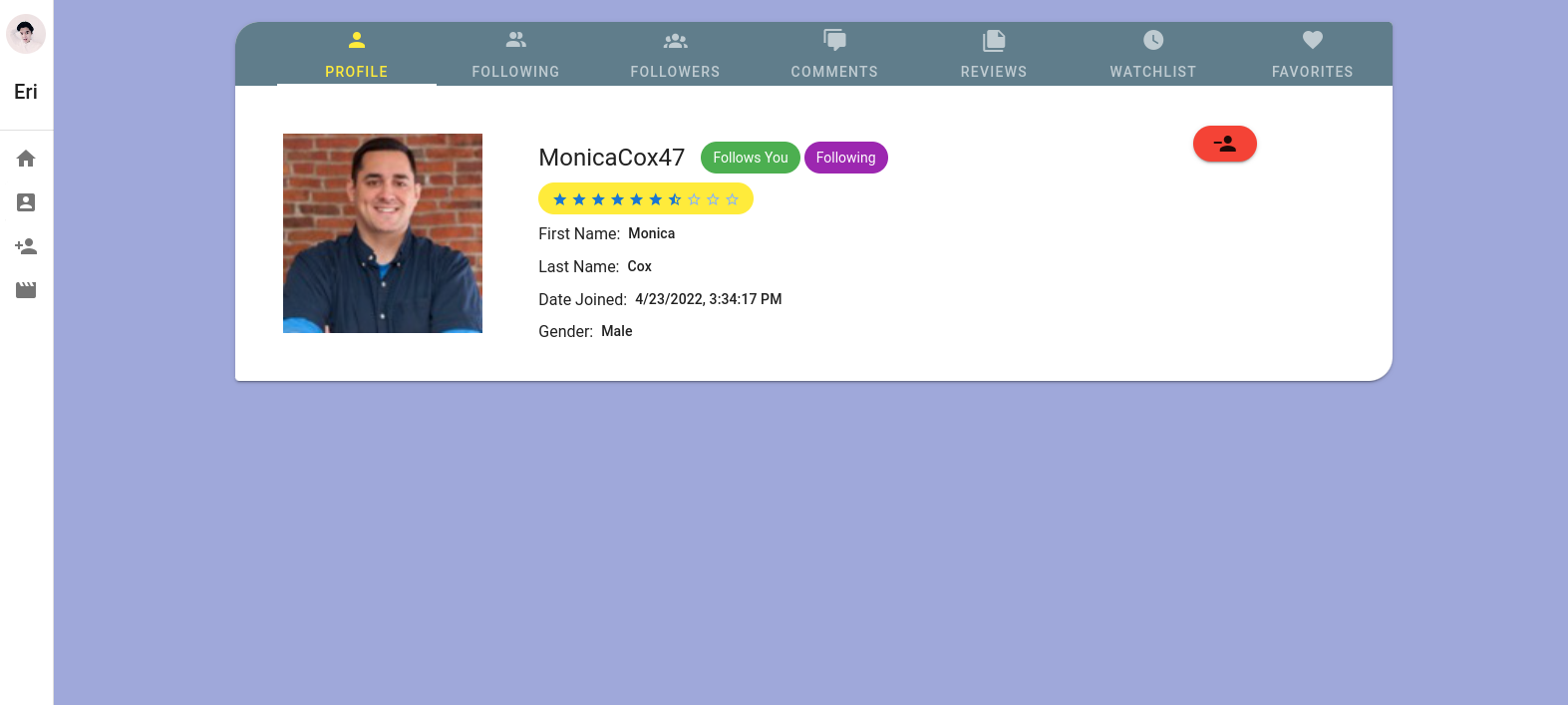
در این صفحه کاربر می‌تواند لیستی از ۱۰۰ کاربر برتر را که بیشترین شباهت را به او داشته مشاهده، وارد پروفایل آن‌ها و در صورت لزوم آن‌ها را دنبال کند. در این لیست علاوه بر مشخصات کاربر، امتیاز تشابه آن‌ها به کاربر نیز نمایش داده شده است. این پنل در تصویر ۷-۲ نمایش داده شده است.



تصویر ۷-۲: لیست کاربران مشابه در صفحه داشبورد کاربر

۳-۴. پروفایل کاربر

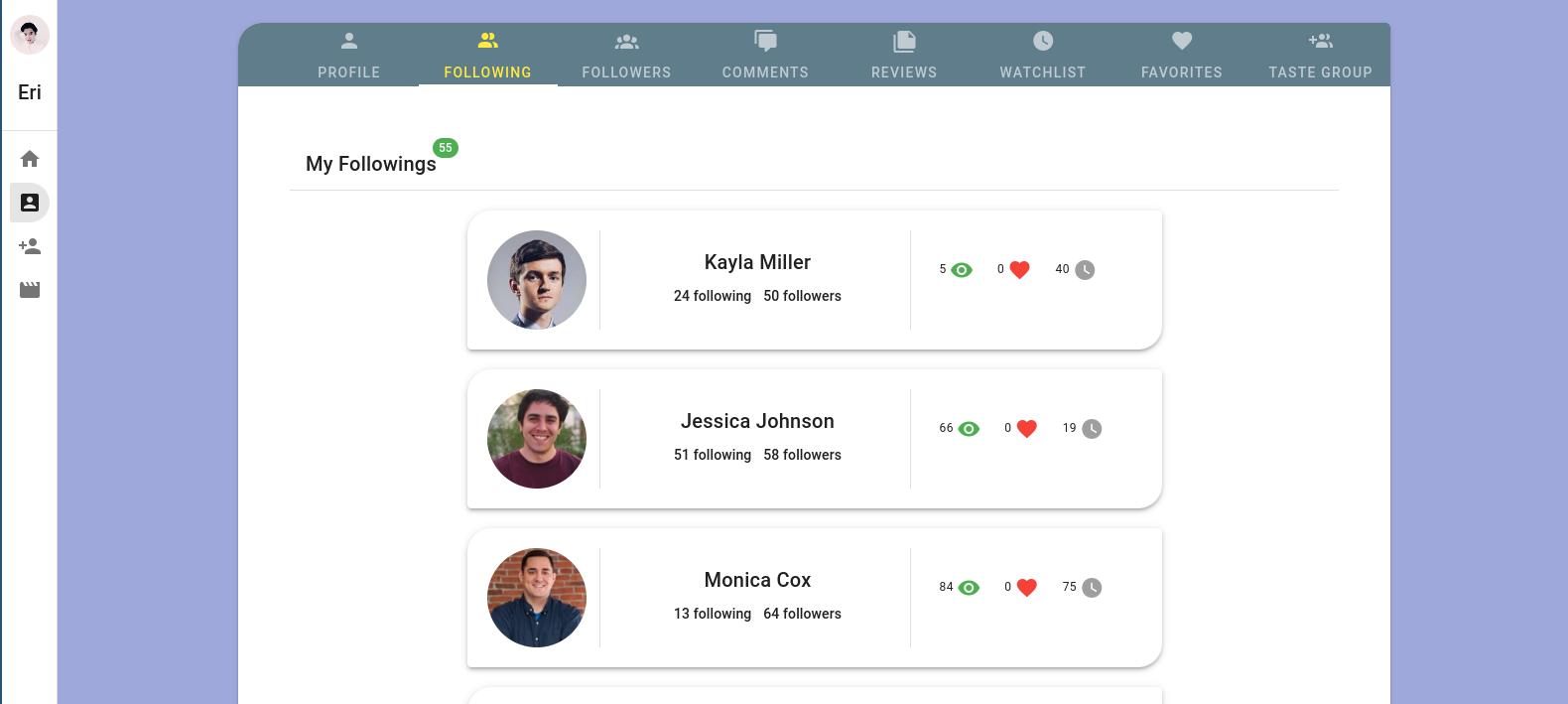
در این صفحه کاربر می‌تواند اطلاعات سایر کاربران را مشاهده یا آن‌ها را دنبال کند. این صفحه در تصویر ۸-۲ نمایش داده شده است.



تصویر ۸-۲: صفحه پروفایل شامل مشخصات کاربر

۳-۴-۱. لیست دنبال‌کنندگان یا دنبال‌شوندگان

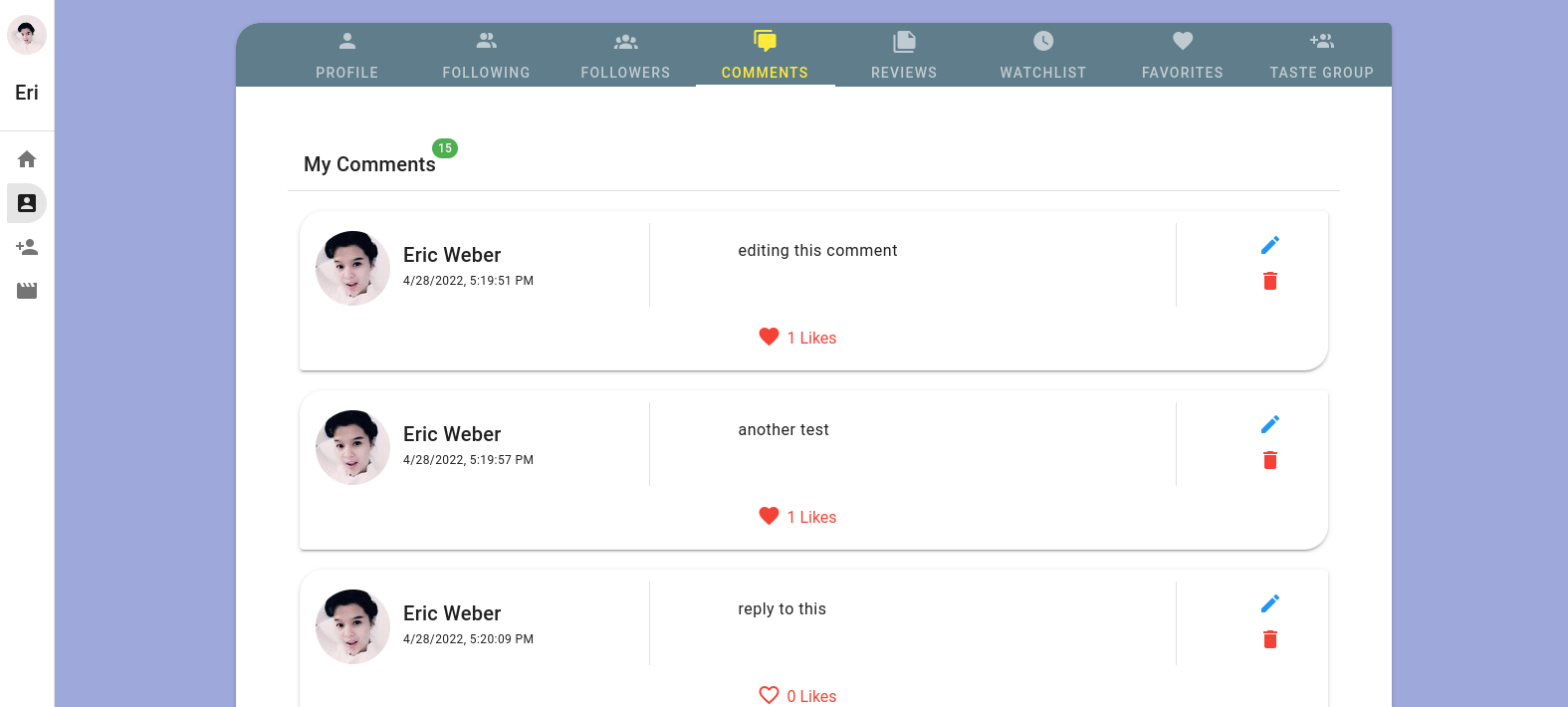
در این صفحه لیستی از کاربرانی را یک کاربر دنبال کرده یا او را دنبال می‌کنند نمایش داده می‌شود. نمونه‌ای از این لیست در تصویر ۹-۲ نمایش داده شده است.



تصویر ۹-۲: لیست دنبال‌شوندگان کاربر

۳-۴-۲. لیست کامنت‌ها

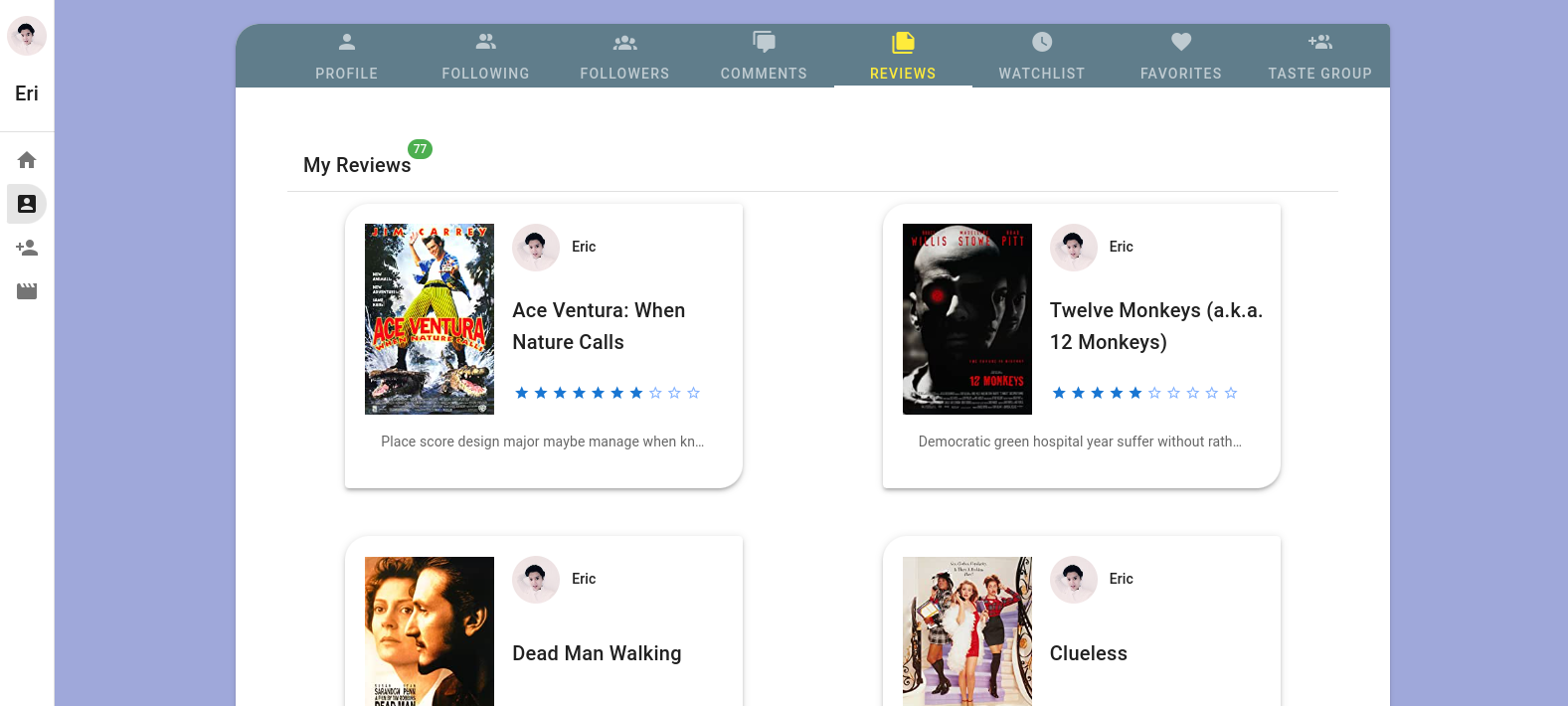
در این صفحه لیستی از تمامی کامنت‌هایی که کاربر در سایت درج کرده نمایش داده می‌شود. نمونه‌ای از این لیست در تصویر ۱۰-۲ نمایش داده شده است.



تصویر ۱۰-۲: لیست کامنت‌های کاربر در پروفایل وی

۳-۴-۳. لیست نقدها

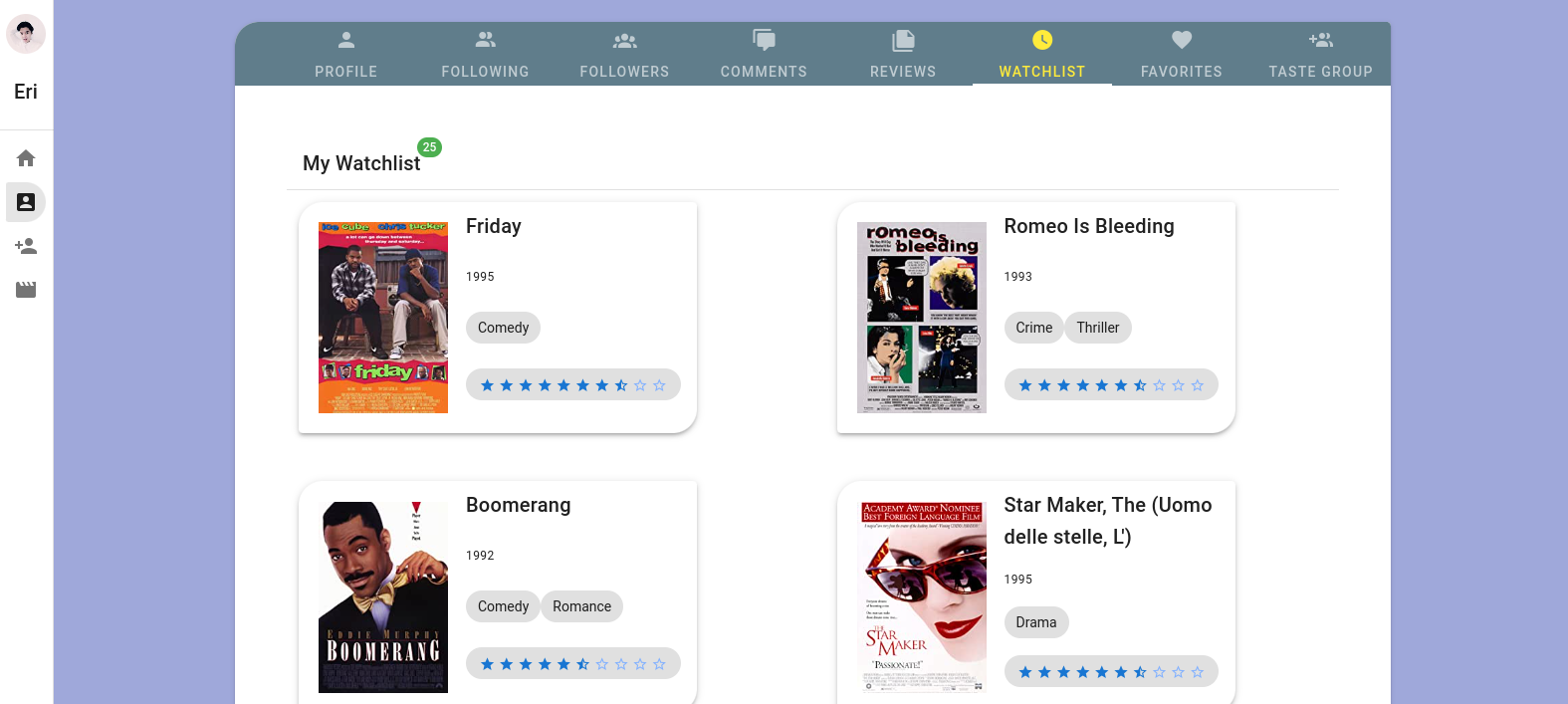
در این صفحه لیست تمامی امتیازات و نقدهایی که کاربر درج کرده نمایش داده می‌شود. نمونه‌ای از این لیست در تصویر ۱۱-۲ نمایش داده شده است.



تصویر ۱۱-۲: لیست نقدهای کاربر در صفحه پروفایل وی

۳-۴-۴. لیست تماشا

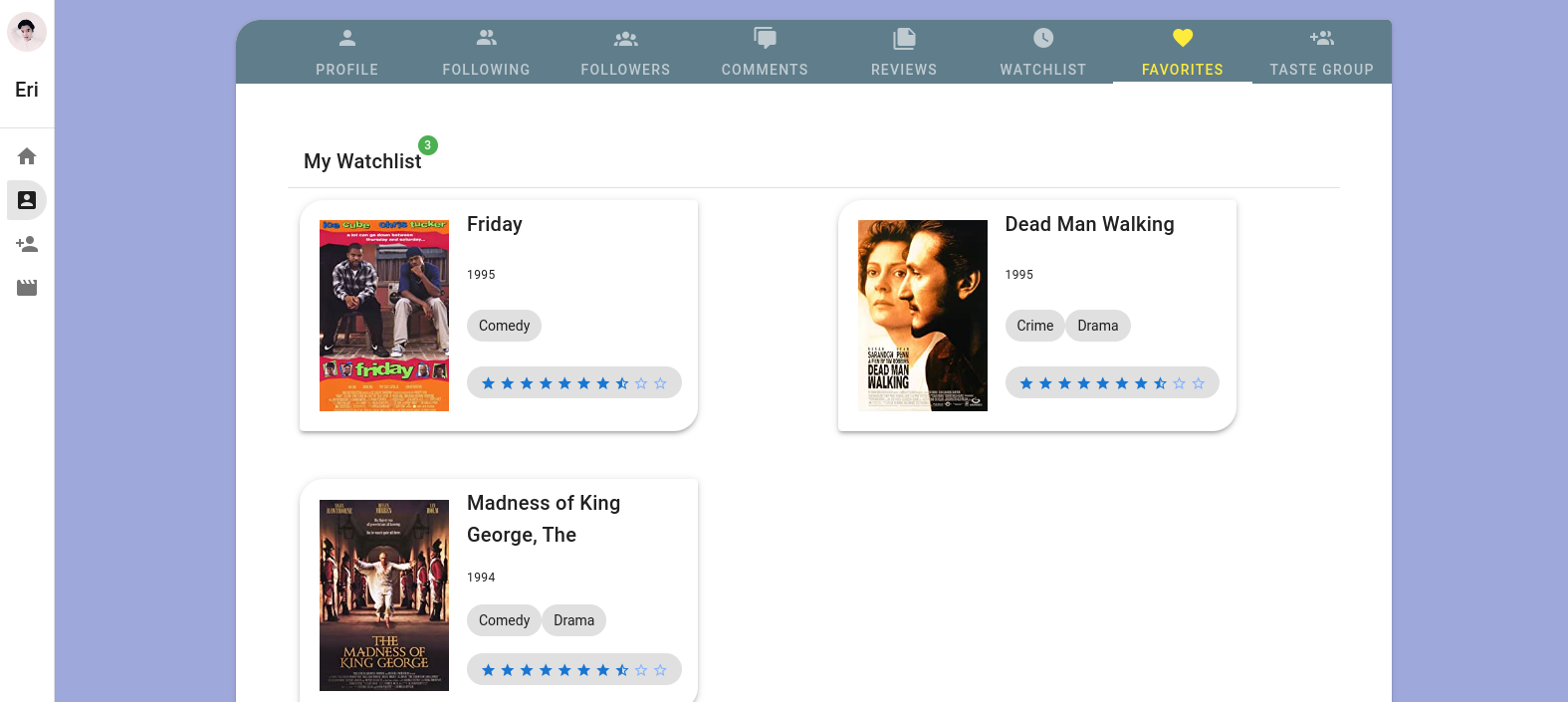
در این صفحه لیستی از فیلم‌هایی که کاربر در لیست تماشای خود قرار داده نشان داده می‌شود. نمونه ای از این لیست در تصویر ۱۲-۲ نشان داده شده است.



تصویر ۱۲-۲: لیست تماشای کاربر در صفحه پروفایل وی

۳-۴-۵. لیست مورد علاقه

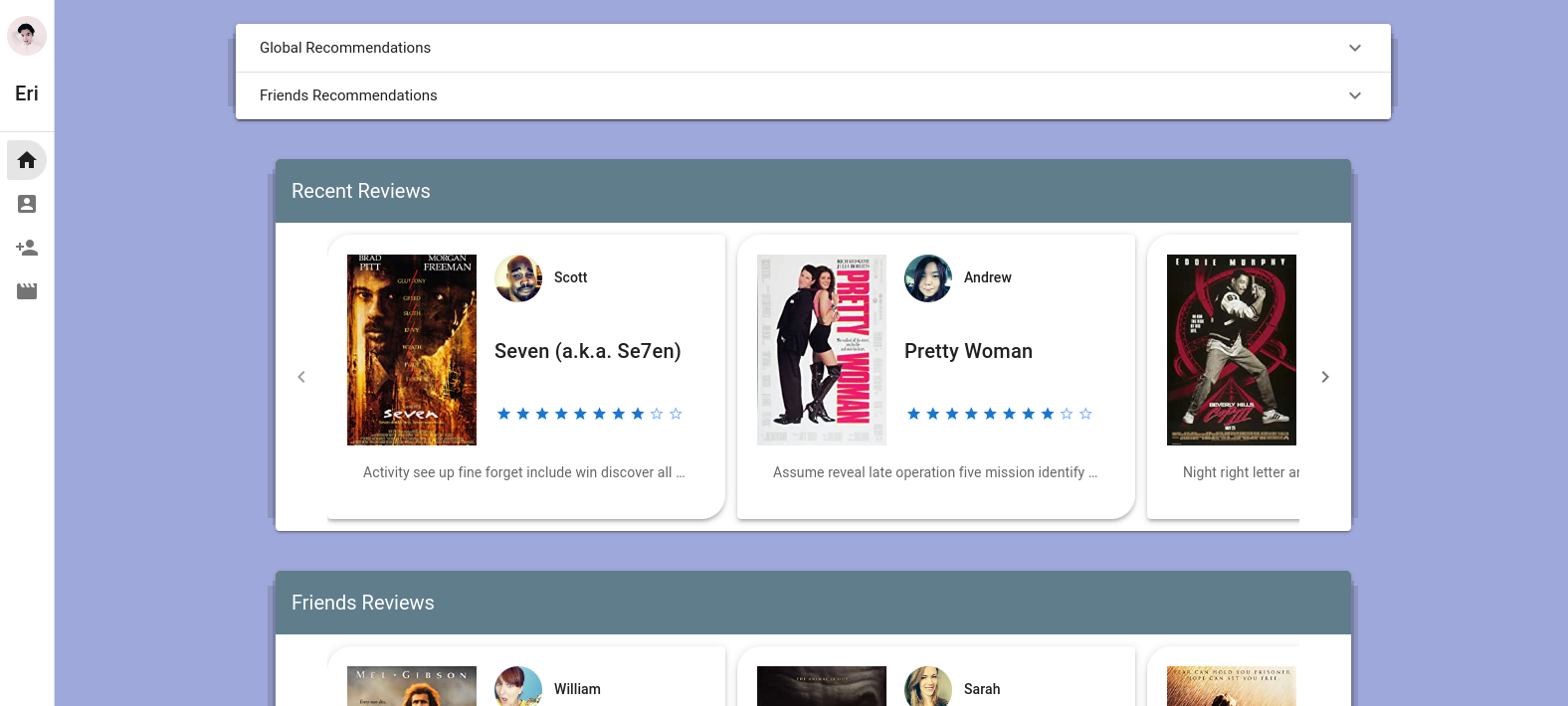
در این صفحه لیستی از فیلم‌های مورد علاقه کاربر نمایش داده می‌شود. نمونه‌ای از این لیست در تصویر ۱۳-۲ نشان داده شده است.



تصویر ۱۳-۲: لیست فیلم‌های مورد علاقه کاربر در صفحه پروفایل وی

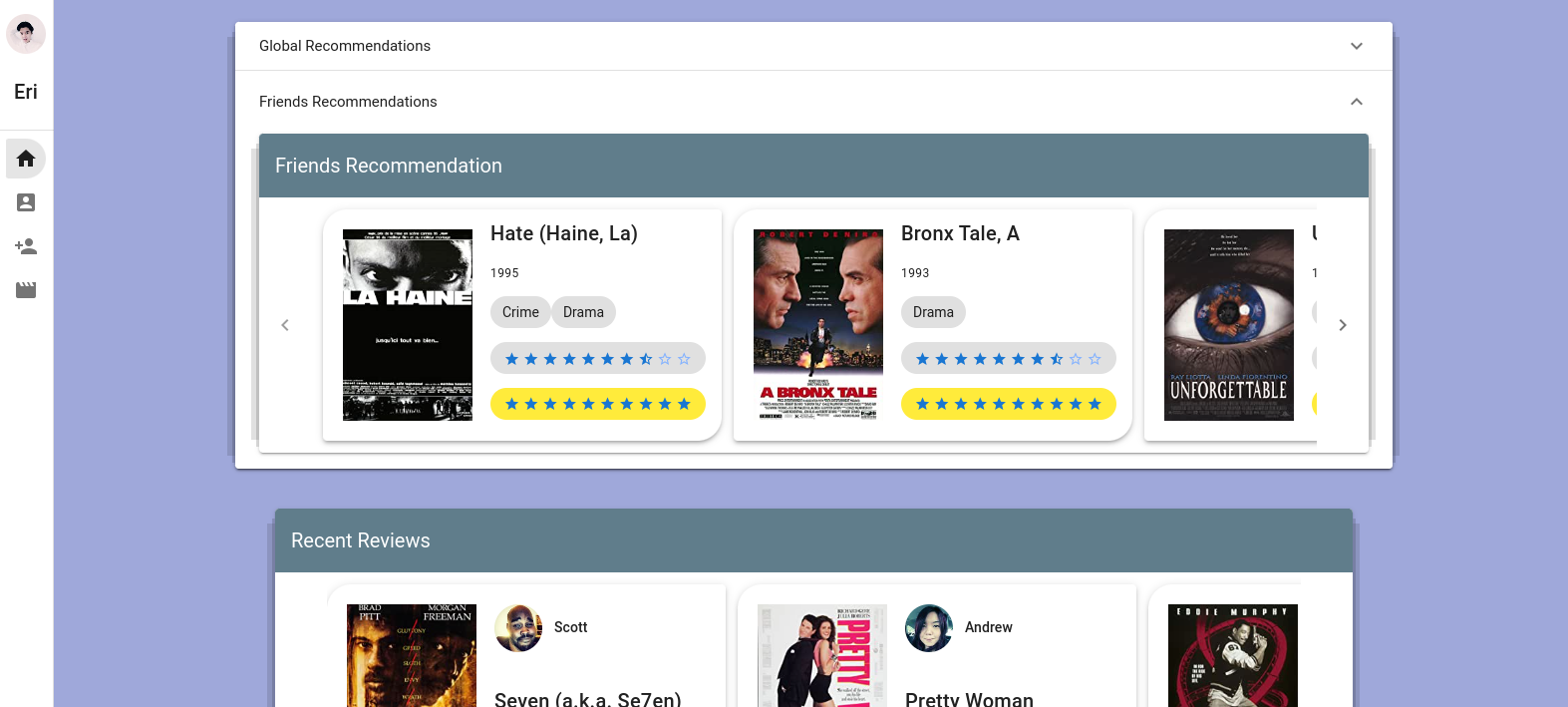
۳-۵. صفحه اصلی

در صفحه اصلی سایت، دو لیست ۱۰ تایی از آخرین نقدهای ثبت شده و همچنین آخرین نقدهایی که دوستان کاربر ثبت کرده‌اند نمایش داده می‌شود. شمایل این صفحه در تصویر ۱۴-۲ نمایش داده شده است.



تصویر ۱۴-۲: صفحه اصلی سایت

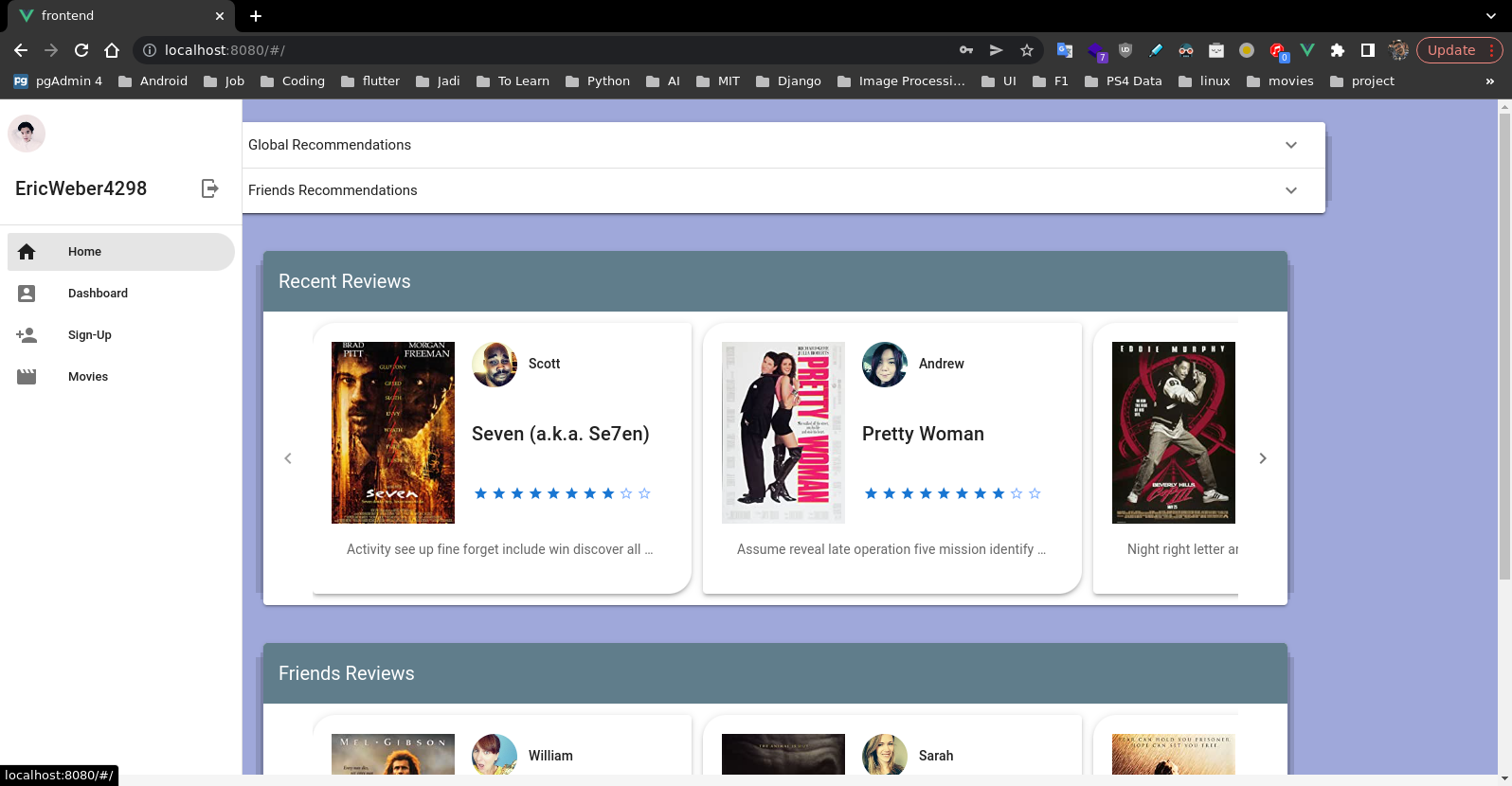
همچنین پنلی برای نمایش لیست فیلم‌های پیشنهادی وجود دارد. با باز کردن این پنل، لیست پیشنهادات محاسبه و برای کاربر نمایش داده می‌شود. نمونه‌ای از این پنل در تصویر ۱۵-۲ نشان داده شده است.



تصویر ۱۵-۲: پنل فیلم‌های پیشنهادی در صفحه اصلی سایت

۳-۶. ساید‌بار

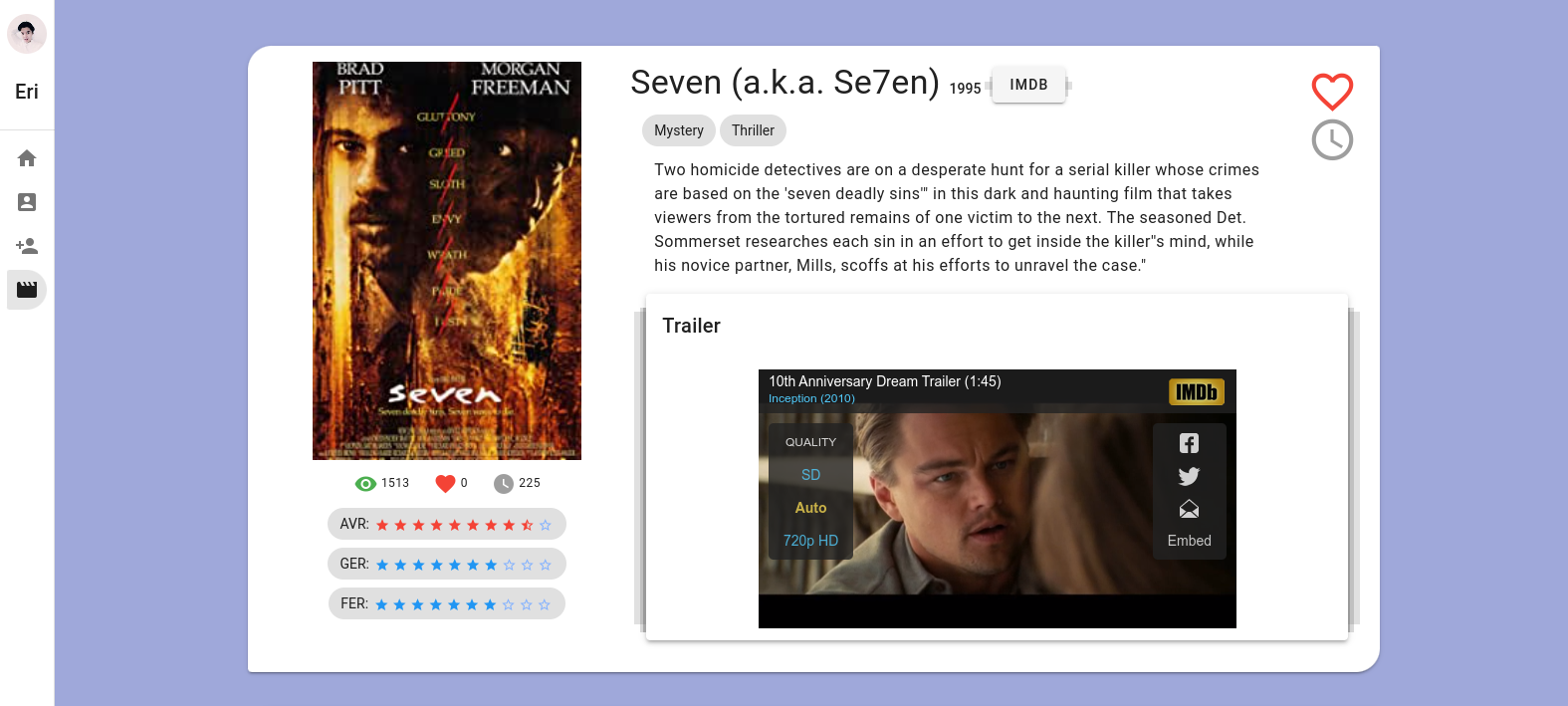
در تمامی صفحات سایت، بخشی در سمت چپ صفحه به عنوان سایدبار(Side Bar) وجود دارد. در این بخش کاربر می‌تواند علاوه بر مشاهده مشخصات خود، از سایت خارج یا بین صفحات موجود در سایت جابه‌جا شود. شمایل این بخش در تصویر ۱۶-۲ نمایش داده شده است.



تصویر ۱۶-۲: پنل سایدبار در سمت چپ تمامی صفحات سایت

۳-۷. صفحه فیلم

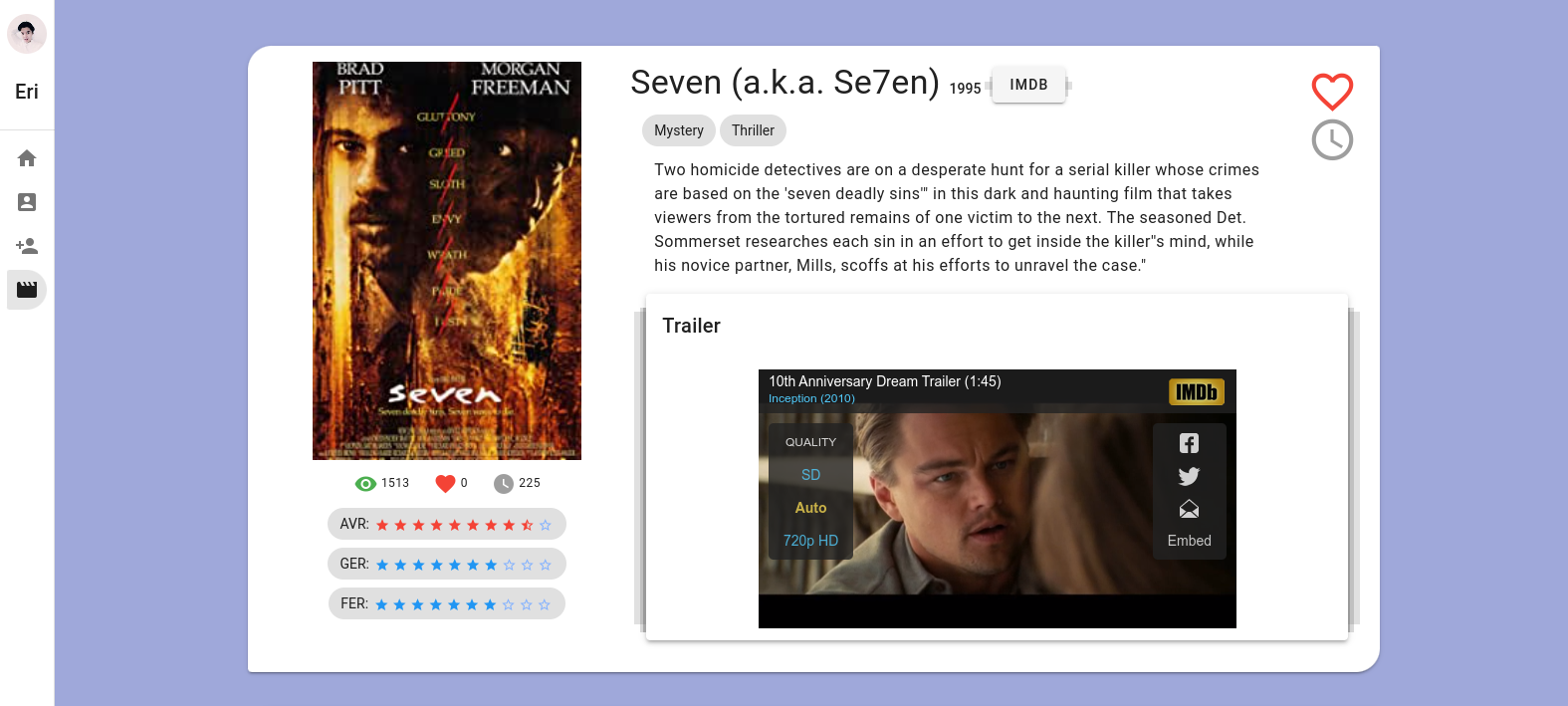
در این صفحه مشخصات فیلم برای کاربر نمایش داده می‌شود. نمونه‌ای از این پنل در تصویر ۱۷-۲ نمایش داده شده است.



تصویر ۱۷-۲: پنل مشخصات فیلم در صفحه هر فیلم

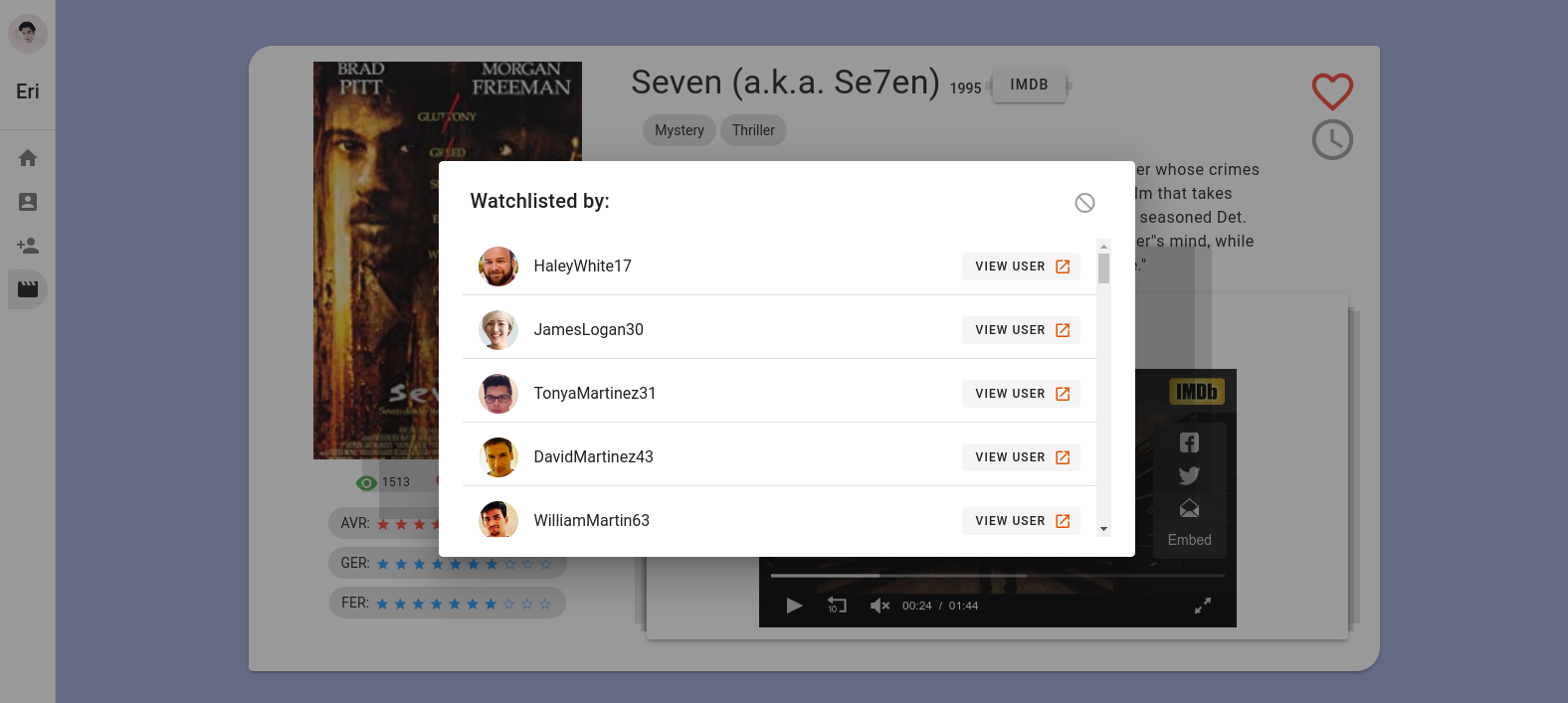
در این بخش کاربر می‌تواند اطلاعات موجود برای فیلم نظیر عنوان و سال تولید، لیست ژانرهای فیلم، خلاصه‌ای از داستان فیلم، لینک پروفایل imdb فیلم یا تریلر فیلم را مشاهده کند. همچنین با دو دکمه‌ای که در سمت راست بالای پنل وجود دارد، این فیلم را لایک کرده یا به لیست تماشای خود اضافه کند. این دو دکمه در تصویر ۱۷-۲ قابل مشاهده می‌باشند.

در پنلی که سمت چپ پایین پوستر فیلم وجود دارد، کاربر می‌تواند تعداد افرادی که فیلم را تماشا کرده، لایک کرده یا در لیست تماشای خود قرار داده‌اند را مشاهده کند. همچنین امتیازات تخمین زده شده برای فیلم نیز در این بخش قابل مشاهده است. شمایل این پنل در تصویر ۱۸-۲ نمایش داده شده است.



تصویر ۱۸-۲: بخش امتیازات مربوط به فیلم در پنل مشخصات فیلم

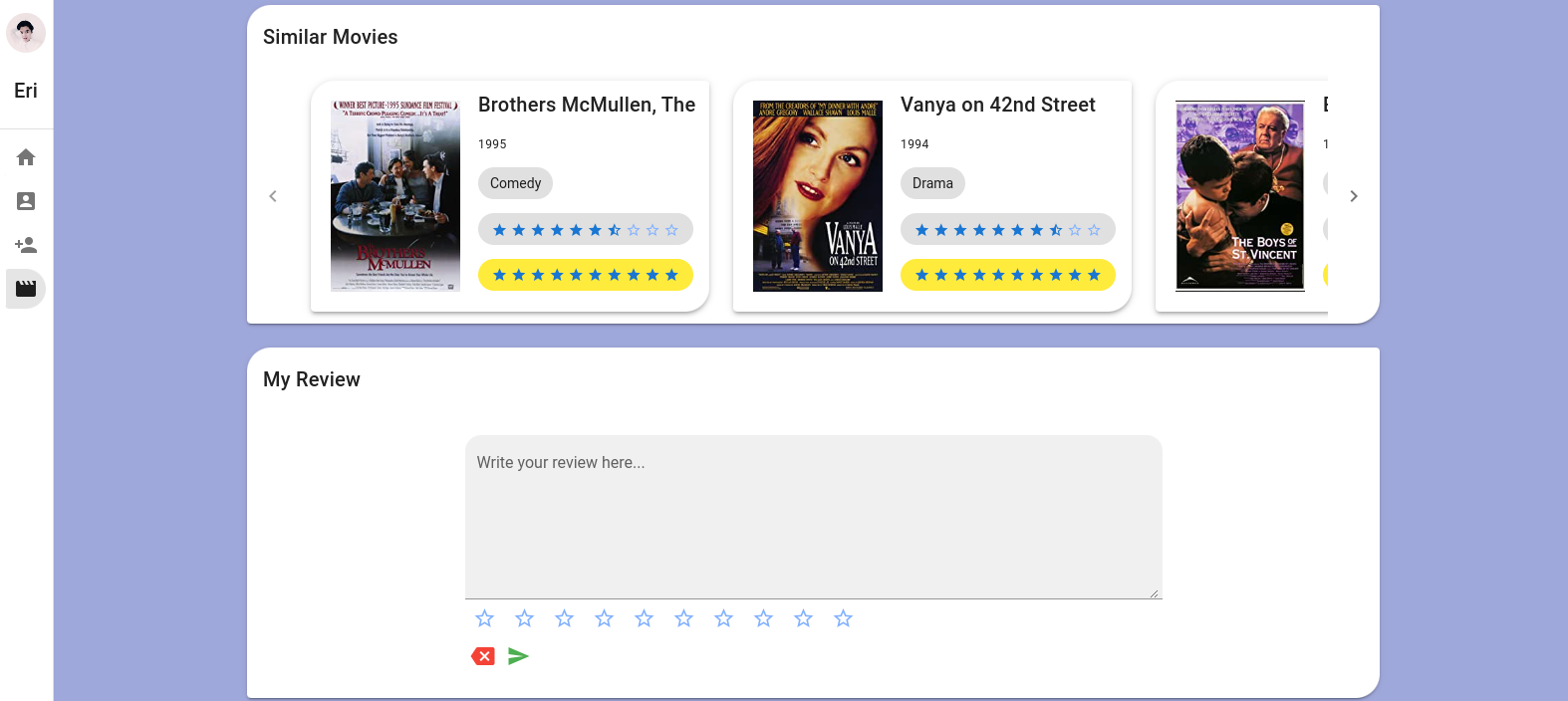
همچنین با کلیک بر روی تعداد لایک‌ها و تعداد لیست‌های تماشا، لیستی از کاربرانی که این فیلم لایک کرده یا در لیست تماشای خود قرار داده‌اند قابل مشاهده خواهد بود. نمونه‌ای از این لیست در تصویر ۱۹-۲ نشان داده شده است.



تصویر ۱۹-۲: نمونه‌ای از لیست کاربرانی که این فیلم را در لیست تماشای خود قرار داده‌اند

۳-۷-۱. لیست فیلم‌های مشابه

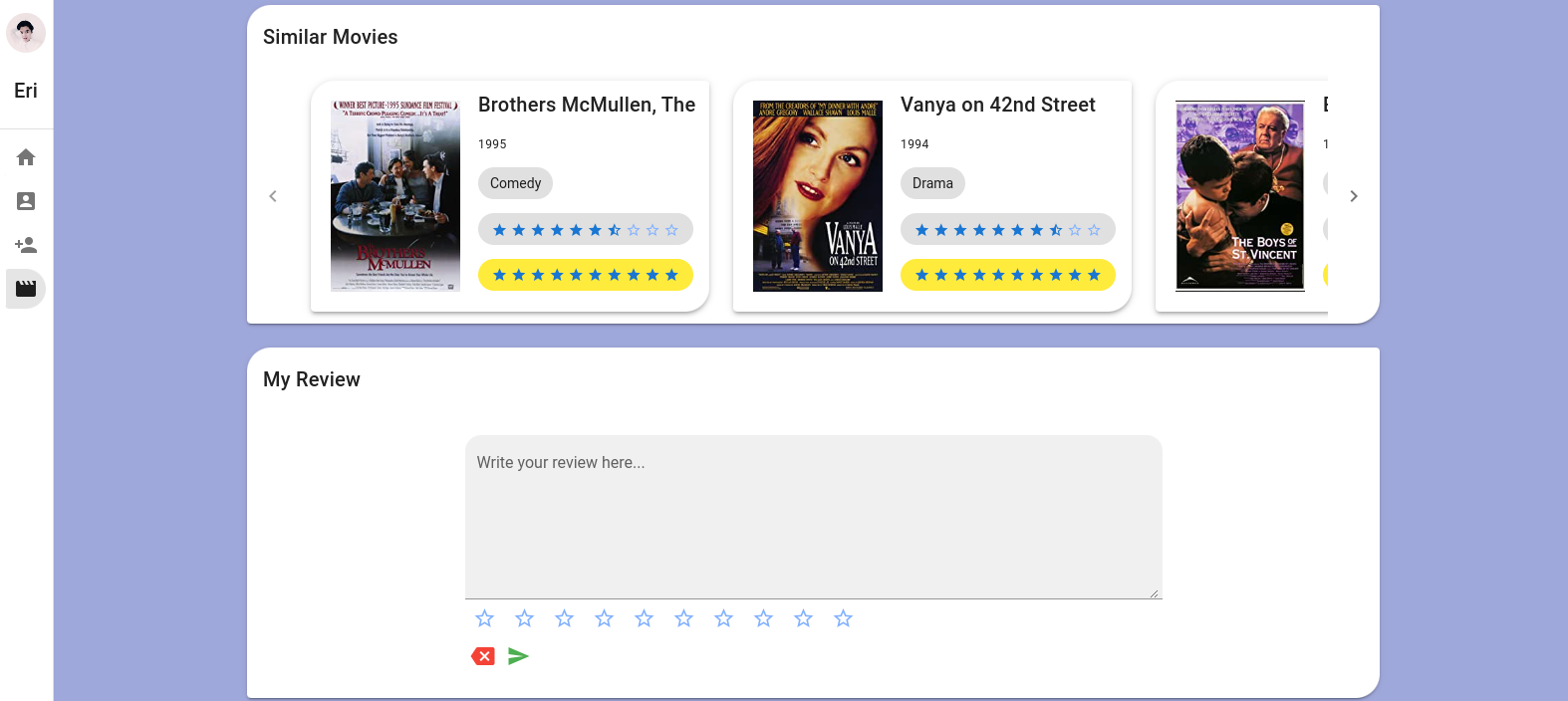
در پایین پنل، لیستی ۱۰ تایی از فیلم‌هایی که بیشترین شباهت را به این فیلم داشته به علاوه امتیاز تشابه آن‌ها قابل مشاهده خواهد بود. شمایل این پنل در تصویر ۲۰-۲ نشان داده شده است.



تصویر ۲۰-۲: پنل فیلم‌های مشابه هر فیلم

۳-۷-۲. ثبت نقد

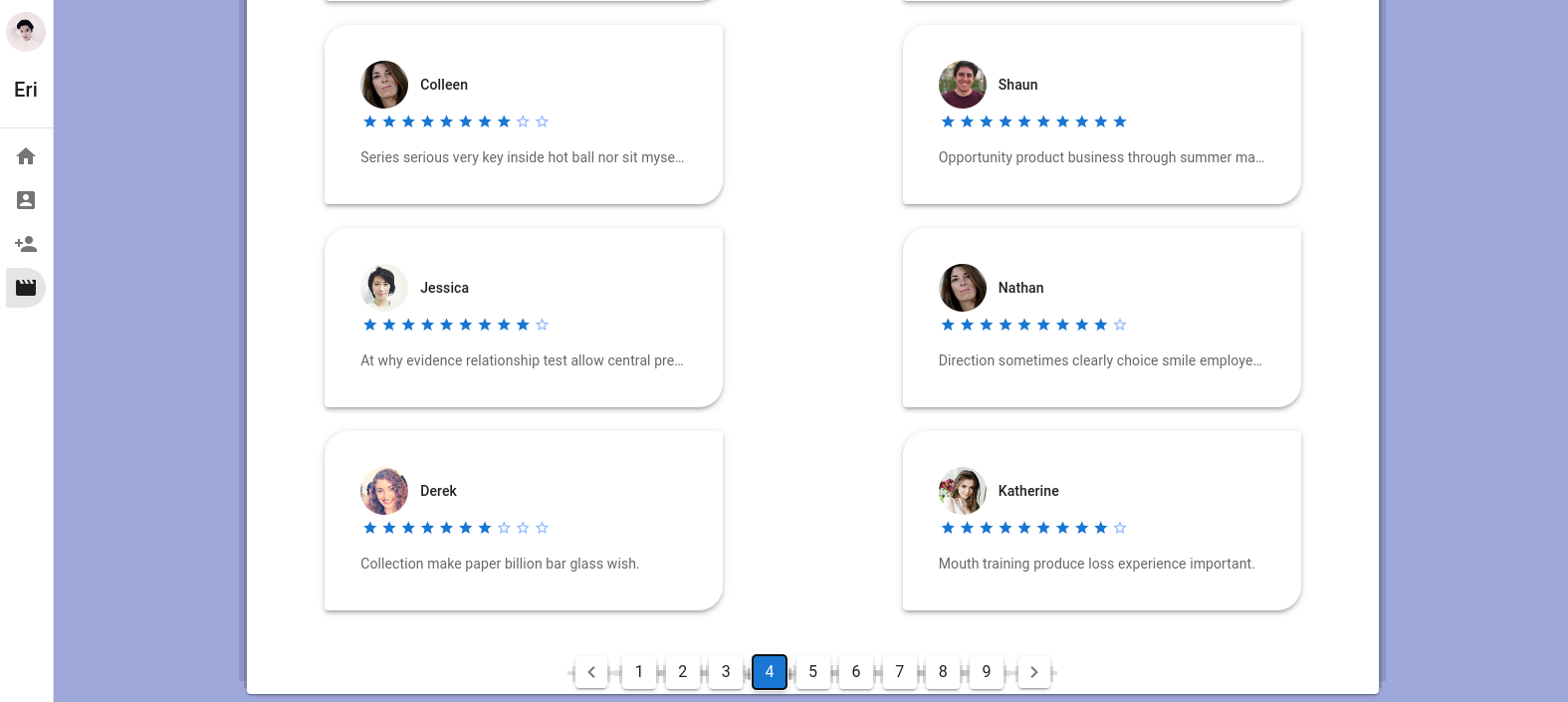
در پایین بخش فیلم‌های مشابه، پنلی برای ثبت نقد برای کاربر وجود دارد. چنانچه پیش از این نقدی برای این فیلم ثبت شده باشد، در این پنل قابل مشاهده خواهد بود. شمایل این پنل در تصویر 21-۲ نمایش داده شده است.



تصویر ۲۱-۲: پنل ثبت نقد در صفحه فیلم

۳-۷-۳. لیست نقدهای موجود

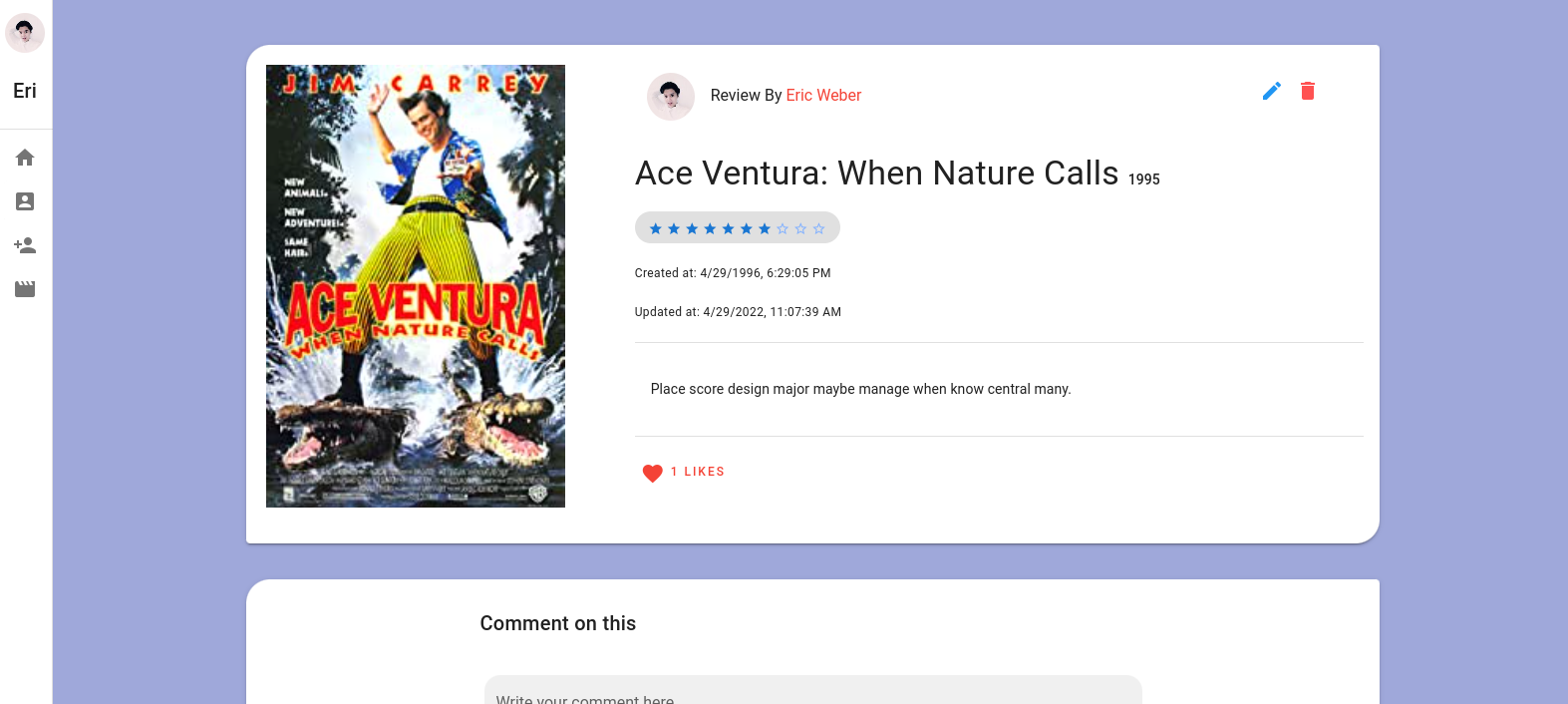
در انتها لیستی از تمامی نقدهای موجود برای این فیلم وجود دارد. نمونه‌ای از این لیست در تصویر ۲۲-۲ نشان داده شده است.



تصویر ۲۲-۲: لیست نقدهای ثبت شده برای این فیلم

۳-۸. صفحه نقد

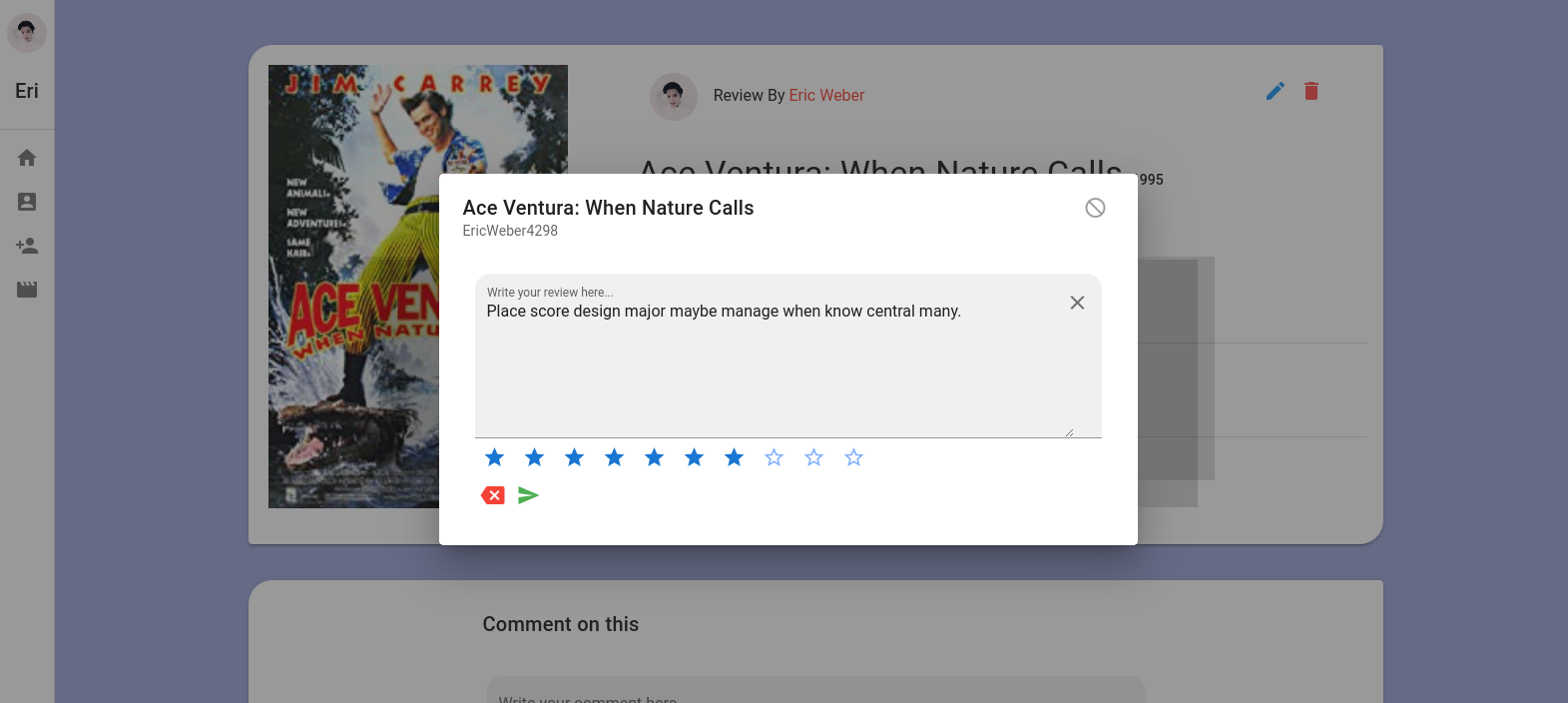
در صورت کلیک بر روی هر نقد، کاربر وارد صفحه مشخصات آن نقد شده. در این صفحه کاربر می‌تواند تمامی اطلاعات مربوط به این نقد را مشاهده کند و در صورت لزوم آن را لایک کرده یا لیست از افرادی که این نقد را لایک کرده‌اند را مشاهده کند. شمایل این صفحه در تصویر ۲۳-۲ نمایش داده شده است.



تصویر ۲۳-۲: پنل مشخصات نقد در صفحه نقد

۳-۸-۱. ویرایش نقد

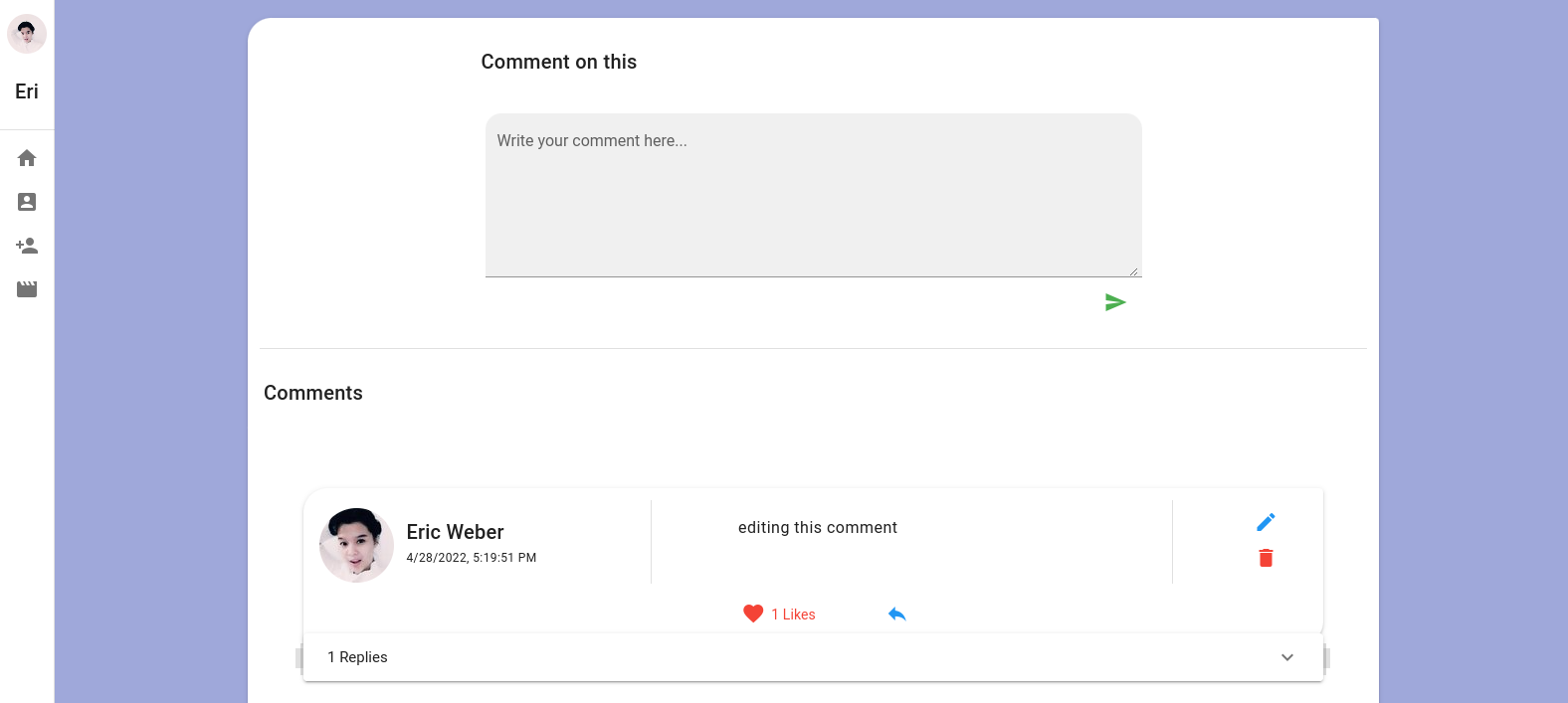
در صورتی که نقد موجود متعلق به کاربر فعلی باشد، این کاربر خواهد بود تا این نقد را در همین صفحه ویرایش یا حذف کند. نمونه‌ای از پنل ویرایش نقد در تصویر ۲۴-۲ نشان داده شده است.



تصویر ۲۴-۲: پنل ویرایش اطلاعات نقد

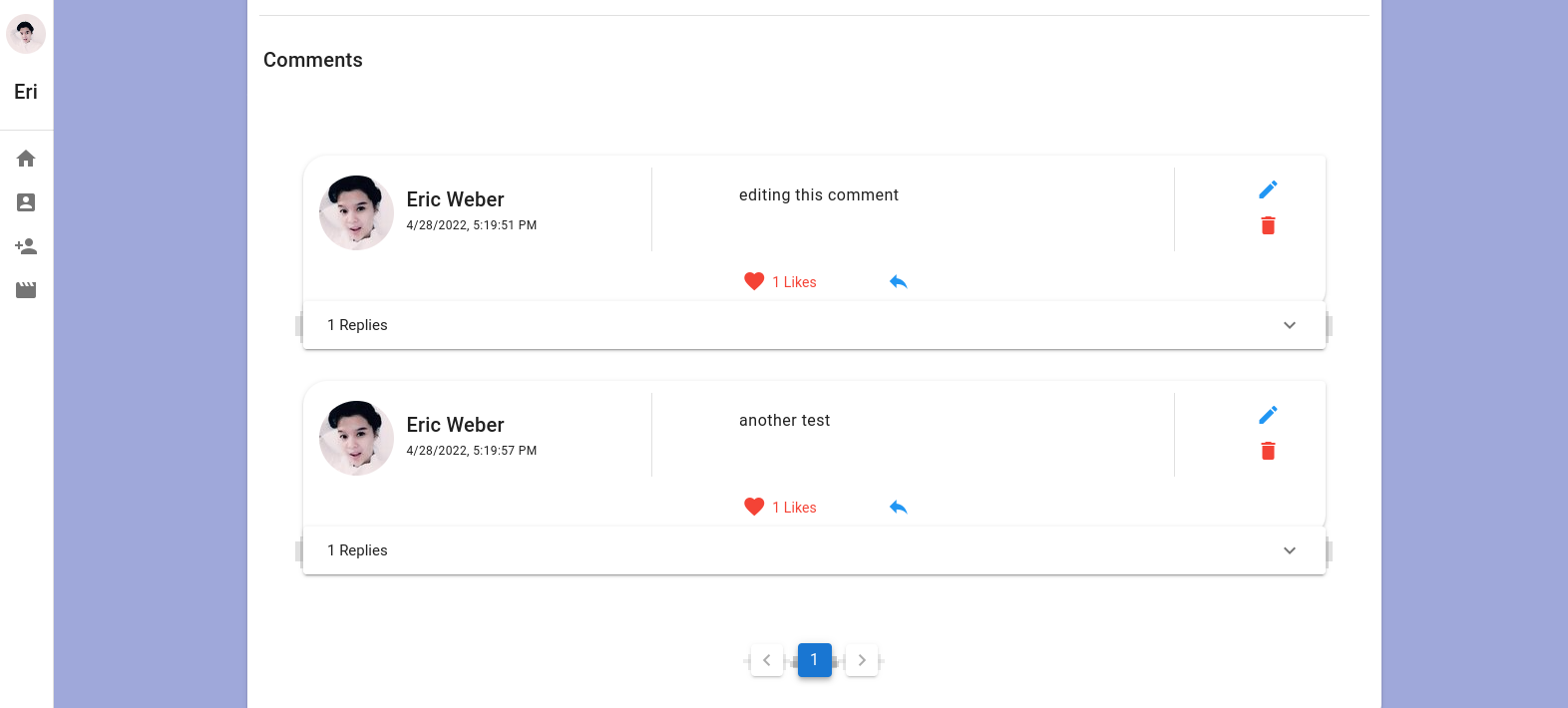
۳-۸-۲. کامنت‌ها

در بخش پایینی این پنل، بخشی برای ثبت یا مشاهده کامنت‌های موجود وجود دارد. شمایل این پنل در تصویر ۲۵-۲ نشان داده شده است.



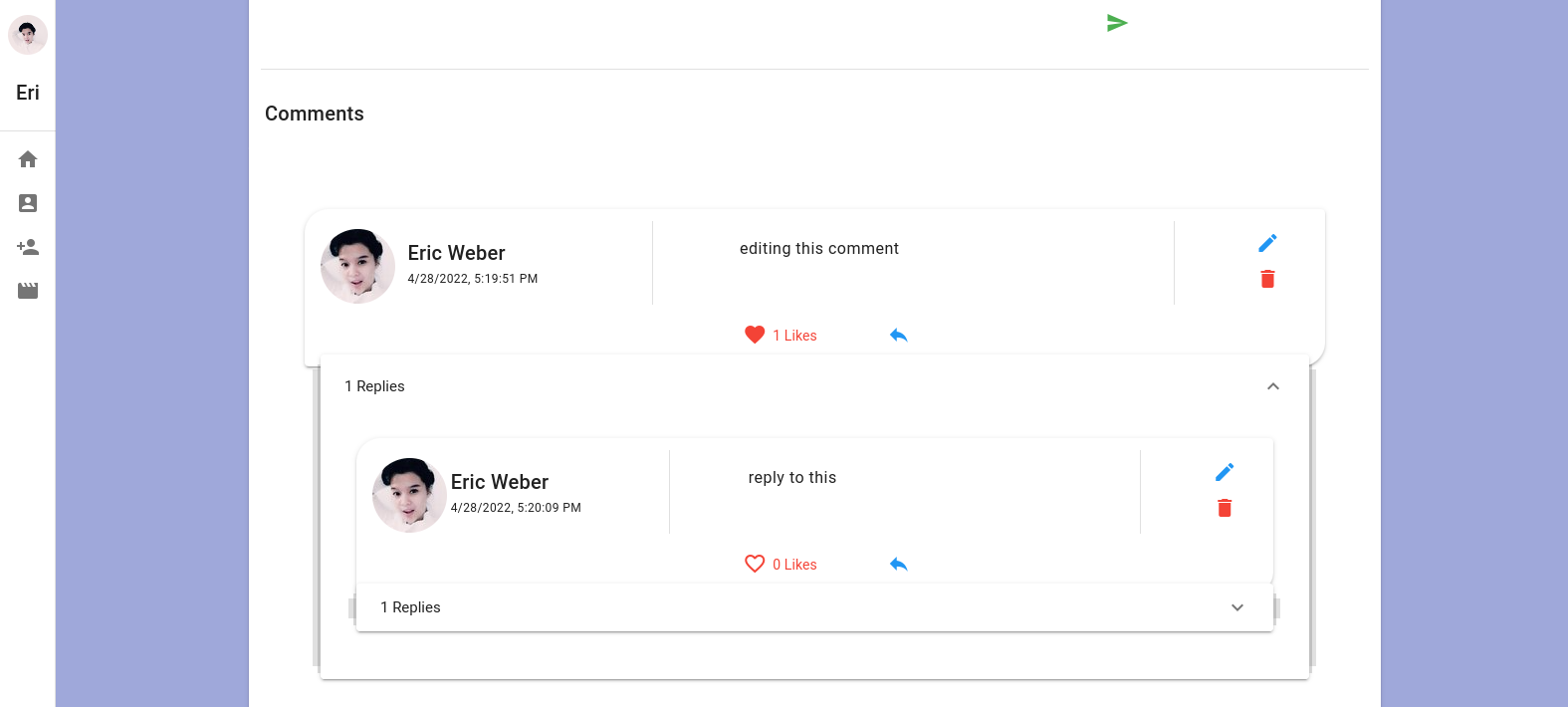
تصویر ۲۵-۲: پنل ثبت کامنت در صفحه نقد

نمونه‌ای از لیست تمامی کامنت‌های موجود در تصویر ۲۶-۲ نشان داده شده است.



تصویر ۲۶-۲: لیست کامنت‌های موجود در صفحه نقد

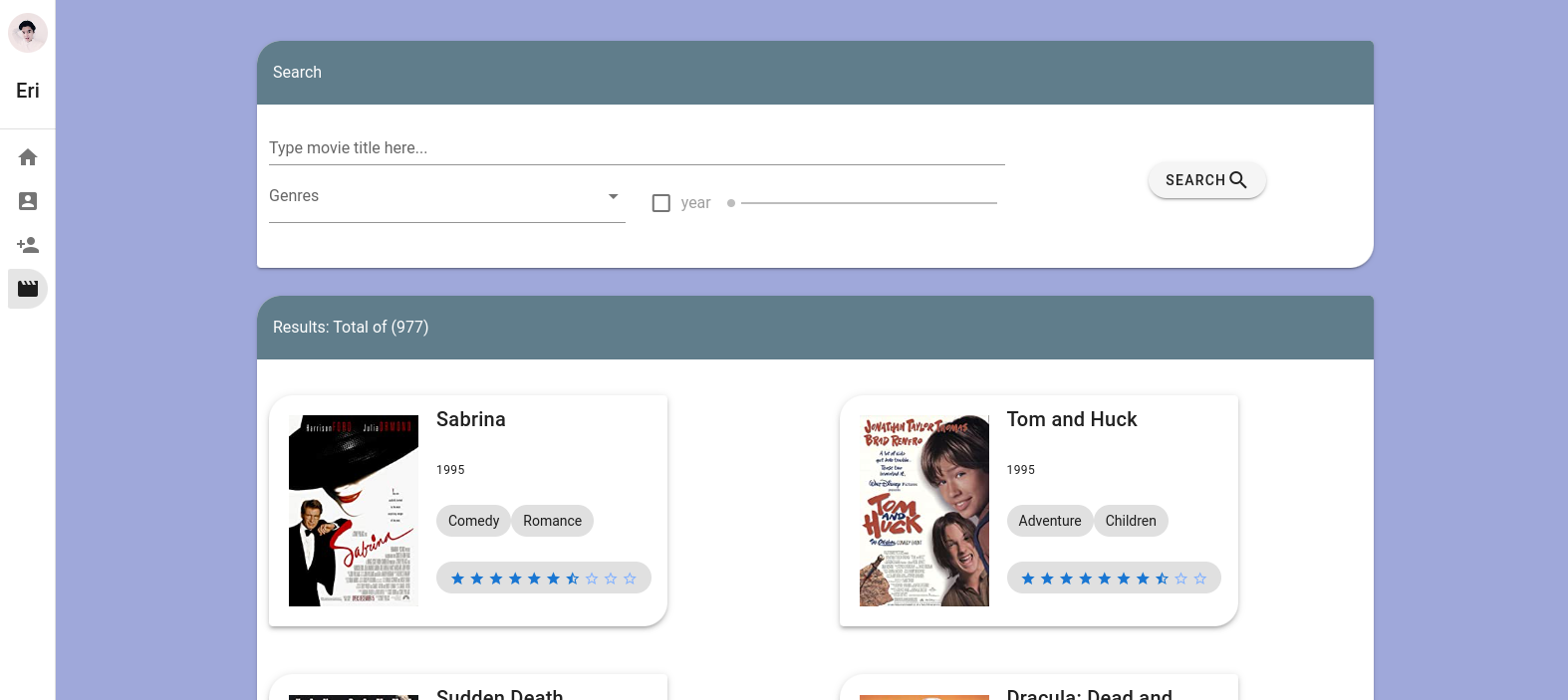
همچنین در صورت لزوم کاربر می‌تواند پاسخی برای هر کامنت ثبت کرده و پاسخ‌های موجود را مشاهده کند. نمونه‌ای از پاسخ‌های ثبت شده برای یک کامنت در تصویر ۲۷-۲ نشان داده شده است.



تصویر ۲۷-۲: نمونه‌ای از لیست پاسخ‌های یک کامنت

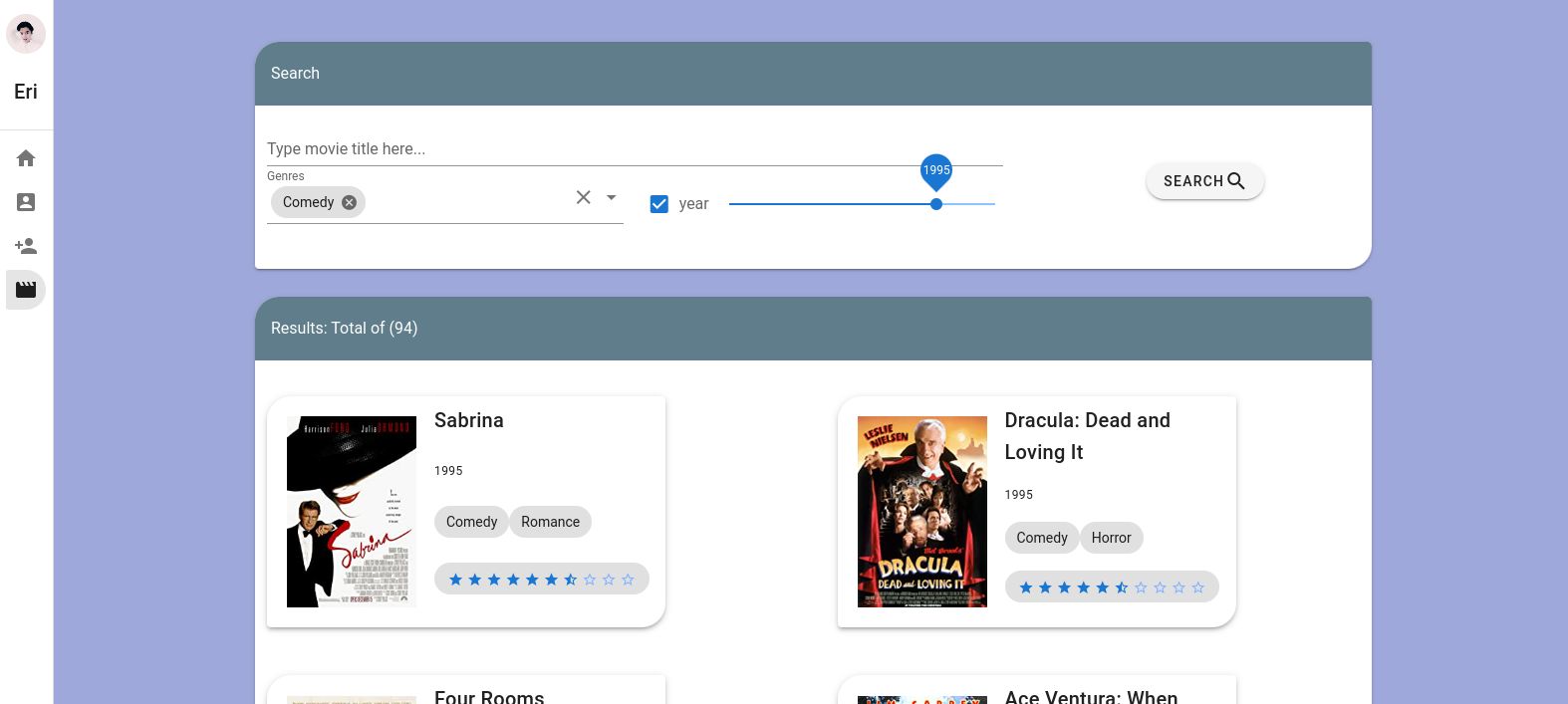
۳-۹. صفحه لیست فیلم‌ها

در این صفحه کاربران قادر خواهند بود تا لیست تمامی فیلم‌های موجود در سایت را مشاهده و از بین‌ها جستجو نمایند. شمایل این صفحه در تصویر ۲۸-۲ نمایش داده شده است.



تصویر ۲۸-۲: صفحه جستجو فیلم‌ها

کاربران می‌توانند فیلم‌ها را با توجه به عناوین آن‌ها جستجو کرده یا بر حسب ژانرهای یا سال تولید آن‌ها فیلتر نمایند. نمونه‌ای از نتایج یک جستجوی انجام شده در صفحه جستجو فیلم‌ها، در تصویر ۲۹-۲ قابل مشاهده می‌باشد.



نمونه‌ای از نتایج جستجو در صفحه جستجوی فیلم‌ها

۴. فهرست منابع

*Celery Official Documention*. (2022). Retrieved from https://docs.celeryq.dev/en/stable/index.html

*Criticker*. (2022). Retrieved from https://www.criticker.com

*Django Official Documention*. (2022). Retrieved from https://docs.djangoproject.com/en/4.0/

*Django Rest Framework Official Documention*. (2022). Retrieved from https://www.django-rest-framework.org/

*Letterboxd*. (2022). Retrieved from https://letterboxd.com

Segaran, T. (2007). *Programming Collective Intelligence.* O’Reilly Media.

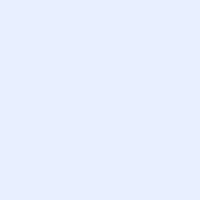
*Vue Router Official Documention*. (2022). Retrieved from https://router.vuejs.org/guide/

*VueJs Official Documention*. (2022). Retrieved from https://v2.vuejs.org/v2/guide/

*Vuetify Official Documention*. (2022). Retrieved from https://vuetifyjs.com/en/getting-started

*Vuex Official Documention*. (2022). Retrieved from https://v3.vuex.vuejs.org/guide/

برگ تأیید هیئت داوران/ صورت‌جلسۀ دفاع (به زبان انگلیسی)



**Abstract**

Nowadays, with the increasing popularity of movies among the audience, a wide variety of content and genres has been created among the produced movies. This diversity of content has made the process of finding the right film for the audience difficult. Having an intelligent system that has the ability to identify, differentiate and rank users and movies based on different character and tastes, will greatly facilitate the process of film finding. This intelligent system will enable users to receive appropriate and personalized recommendations from the system and get acquainted with users with similar character and tastes.

In this project, we design and implement an intelligent recommender system using collaborative filtering algorithms, and then use some common web design framework such as Django and VueJs to design and implement an interactive web-based environment for users to interact with the system and each other.

**Keywords:**

****

Institution/University of …

Faculty of ...

A Dissertation Submitted in Partial Fulfillment for the Degree of …..

Design and Implementation of Cinema Social Media including Recommendation System

By

**Sadegh Hamedani Poor**

Supervisor

**Dr. Maryam Chini Pardaz**

Advisor

**Dr. Maryam Chini Pardaz**

**May 2021**