



پروژه موزیک پلیر فاز دوم

دانشگاه اصفهان
برنامه نویسی پیشرفته - دکتر رضا رضانی

بهار 1403

تاریخ تحویل

1403/2/28

طراحان پروژه

ثنا سبحانی
حسین احمدی
مهرآفرین صیرفیان

زهرا معصومی
یونس جمشیدی
مهدی مظاهری

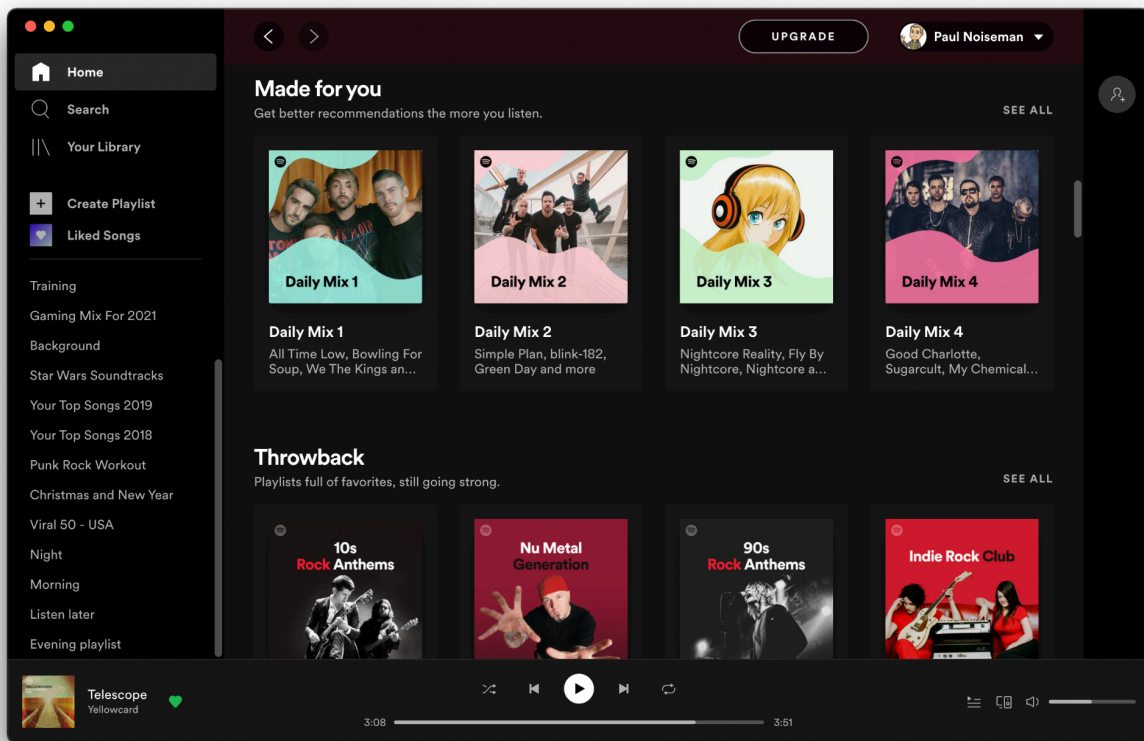


فهرست

2.....	معرفی پروژه
3.....	تاریخ تحویل و نکات مورد توجه
4.....	شرح پروژه
4.....	(A) واسط‌ها
5.....	(B) مدیریت استثناء
7.....	(C) گرافیک
10.....	نکات پیاده‌سازی
11.....	معیارهای ارزیابی

معرفی پروژه

در این پروژه، قصد داریم موزیک پلیری مشابه [Spotify](#) پیاده‌سازی کنیم.



در این برنامه عملیات زیر به صورت کلی قابل انجام است:

1. پخش فایل‌های صوتی
2. ایجاد پلی‌لیست‌ها و افزودن فایل صوتی به آن‌ها
3. نمایش اطلاعات آرتیست‌ها و دنبال کردن آن‌ها
4. انتشار موزیک یا پادکست توسط آرتیست‌ها
5. مشاهده آمار مربوط به فایل‌های صوتی و آرتیست‌ها

هدف فاز دوم این پروژه، سنجش میزان یادگیری شما در زمینه مفاهیم واسطه‌ها، مدیریت استثناء، طراحی گرافیکی در جاوا و توانایی در استفاده از این مفاهیم در عمل است. با انجام این پروژه، مفاهیمی که تا کنون از زبان جاوا آموخته‌اید را به کار خواهید گرفت و امیدواریم علاوه بر کسب

تجربه‌ای خوب از برنامه‌نویسی و انجام پروژه‌ای کاربردی، سطح دانش و توانایی کدزنی شما بیش از پیش افزایش یابد.

Note: لطفاً قبل از شروع کد زدن، مستند پروژه را تا انتها مطالعه فرمایید! (=)

تاریخ تحویل و نکات مورد توجه

تاریخ تحویل این پروژه، 28 اردیبهشت 1403 می‌باشد. موارد زیر در تحویل و ارائه پروژه مورد بررسی قرار خواهد گرفت:

1. همان‌طور که گفته شد، هدف این پروژه سنجش میزان یادگیری و درک شما از مفاهیم شی‌گرایی و برنامه‌نویسی در زبان جاوا می‌باشد؛ در نتیجه رعایت اصول شی‌گرایی (محصول‌سازی، ارث‌بری، چندریختی و انتزاع) و مدیریت استثناء (پرتاب و مدیریت صحیح) در نمره‌دهی پروژه شما بسیار اثرگذار خواهد بود.
2. این پروژه باید با بهره‌گیری از معماری MVC پیاده‌سازی شود. تشخیص مواردی که باید در هر بخش از این معماری (model-view-controller) قرار گیرند، به عهده خودتان است.
3. موارد امتیازی برای یادگیری بیشتر دانشجویان علاقمند و همچنین نمره‌ای اضافه بر نمره کل قرار داده شده‌اند. طبیعتاً پیاده‌سازی این موارد اجباری نیست اما در صورت پیاده‌سازی، تسلط کافی برای ارائه داشته باشید.
4. نحوه پیاده‌سازی مواردی که «صریحاً» در مستند پروژه ذکر نشده است، به عهده خودتان و خلاقیت خودتان خواهد بود. از آزمون و خطا نترسید و بهترین روشی که به نظرتان می‌رسد را پیاده‌سازی کنید. هرگونه راه‌حل قابل اجرا در صورت کارکرد درست و عدم تناقض با اصول شی‌گرایی و معماری MVC، نمره کامل را خواهد گرفت.
5. ما در تیم‌های امیدواریم که در انتهای این پروژه شاهد افزایش مهارت شخص شما در برنامه‌نویسی باشیم؛ در نتیجه هرگونه مشابهت «غیر معمول» بین پروژه‌های ارسالی، منجر به صفر شدن نمره تقلب‌گیرنده، و نه تنها صفر شدن، بلکه کم شدن کل نمره پروژه (100%-) از فرد تقلب‌دهنده خواهد شد.



شرح پروژه

این فاز از پروژه، در ادامه فاز یک پیاده‌سازی خواهد شد. در نتیجه نیاز است مشکلات کد خود را با توجه به بازخورد تیمی‌ها در ارائه فاز یک، اصلاح نموده و سپس به پیاده‌سازی فاز دو بپردازید. در این بخش، اجزای پروژه را شرح می‌دهیم؛ لطفاً با دقت کافی و توجه به جزئیات این بخش را مطالعه کنید.

(A) واسط‌ها

برای موجودیت‌های فاز یک، واسط‌های زیر را پیاده‌سازی کنید:

• واسط Iterable

این واسط را برای کلاس پلی‌لیست به نحوی پیاده‌سازی کنید که بتوانیم با حلقه foreach روی آهنگ‌های پلی‌لیست پیمایش انجام دهیم. سپس از این نوع حلقه در View مربوط به نمایش پلی‌لیست‌ها استفاده کنید.

• واسط Comparable

برای کلاس فایل صوتی، این واسط را به صورتی پیاده‌سازی کنید که ویژگی‌های زیر از دو فایل صوتی را به ترتیب زیر با هم مقایسه کند:

- عنوان فایل صوتی (به ترتیب حروف الفبا)
 - تعداد لایک‌ها
 - بر اساس نوع فایل صوتی (ابتدا آهنگ‌ها و بعد پادکست‌ها)
 - تعداد پلی شدن
- سپس عملیات مرتب‌سازی را با استفاده از این قابلیت، بازنویسی کنید.

(B) مدیریت استثناء

استثناءهای زیر را در کد فاز قبل خود در نظر بگیرید. هرجایی که ممکن است خطاهای زیر رخ دهد، استثناء مربوط به آن را پرتاب کرده و در View مناسب مدیریت کنید.

• خطای فرمت نامعتبر

هنگام ثبت نام در صورتی که فرمت ایمیل یا شماره تلفن درست نباشد خطای `InvalidFormatException` پرتاب شود.

• خطای ورود ناموفق

هنگام ورود به حساب کاربری در صورتی که نام کاربری وجود نداشته باشد، خطای `UserNotFoundException` پرتاب شود و در صورتی که رمز عبور صحیح نباشد، خطای `WrongPaswordException` پرتاب شود. هر دوی این کلاسها از یک کلاس پدر به نام `FailedLoginException` ارث‌بری می‌کنند.

• خطای کمبود اعتبار

هنگامی که کاربر می‌خواهد اشتراک حساب پرمیوم را تهیه یا تمدید کند، در صورتی که اعتبار حسابش برای خرید بسته مورد نظر کافی نیست این خطا پرتاب شود.

• خطای محدودیت حساب عادی

هنگام ساخت پلی‌لیست جدید، در صورتی که حساب کاربری عادی و تعداد پلی‌لیست‌های فعلی کاربر 3 تا (سقف تعداد پلی‌لیست‌ها برای کاربر عادی) باشد؛ یا هنگامی که به یک پلی‌لیست فایل صوتی اضافه می‌کنیم و آن پلی‌لیست در حال حاضر 10 فایل صوتی دارد، خطای `FreeAccountLimitException` پرتاب شود.

نکات قابل توجه:

1. برای هر نوع خطا، یک کلاس `Exception` بسازید و در سازنده آن، پیغام مناسب را به سازنده پدر بدهید.
2. همه خطاها باید در View و با توجه به نوع آن‌ها مدیریت شوند و پیغام مناسب به کاربر نمایش داده شود.

3. استثنای پیش فرض جاوا مثل `InputMismatchException` یا `ParseException` و خطاهای مشابه را به گونه‌ای مدیریت کنید که وقتی اتفاق افتاد، پیغام خطا نمایش داده شده و کاربر دوباره به همان صفحه برگردد. (مثلاً در قسمت ساخت حساب کاربری هستید و ویژگی تاریخ که فرمت مشخصی باید داشته باشد را به اشتباه وارد می‌کنید و استثنای `ParseException` (یا خطاهای مشابه در صورت استفاده از `formatter`های دیگر) پرتاب می‌شود. پس از مدیریت آن در `View` و مشاهده متن خطا توسط کاربر، در صفحه ساخت حساب کاربری بمانید تا کاربر بتواند دوباره عملیات ثبت نام را انجام دهد...)

4. در صورتی که برنامه به هر نحوی تمام شد (چه با استثنا چه به صورت عادی) پیغام "Have a good day" را به کاربر نمایش دهید.

5. مدیریت خطا و نحوه نمایش آن، به عهده خلاقیت خودتان است. می‌توانید خطا را در همان صفحه و زیر فیلد مورد نظر نمایش بدهید یا به صورت `Alert` و `JOptionPane` در یک پنجره جداگانه.

C) گرافیک

به جای View های کنسولی در فاز قبل، می‌خواهیم صفحات گرافیکی را قرار بدهیم. در ادامه صفحه‌های گرافیکی مورد نیاز برنامه ذکر شده است. به کار بردن خلاقیت و پیاده‌سازی بخش‌های اضافه در این بخش آزاد است، اما حتماً موارد ذکر شده را پیاده‌سازی کنید.

• ساختار کلی صفحات

همه صفحات برنامه، بخش‌های زیر را به صورت ثابت و مشترک دارند:

- Sidebar: منوی کناری برنامه که شامل گزینه‌های زیر است:
 - Home: با انتخاب این گزینه وارد صفحه Home می‌شویم.
 - Search: می‌توانید یک فیلد در منو برای این کار در نظر بگیرید، یا کلاً یک صفحه جدید برای آن بسازید. در نهایت این گزینه باید عملیات «جستجو» که در فاز قبل پیاده‌سازی کردید را اجرا کند و نتایج جستجو را در یک صفحه جدید نمایش دهد.
 - Library: با کلیک روی این گزینه، در صورتی که کاربر لاگین کرده باشد وارد پنل حساب خود می‌شود. در غیر این صورت پیام خطا نمایش داده شود.
 - All Artists: در صفحه‌ای جدید، لیست تمامی آرتیست‌ها را نمایش دهید.
 - All Audios: در صفحه‌ای جدید، لیست تمامی فایل‌های صوتی را نمایش دهید.
 - Main section: این بخش برای هر صفحه متفاوت است و شامل محتوای مربوط به آن صفحه می‌شود.
 - Play bar: بخش پایین صفحه که شامل برخی اطلاعات فایل صوتی، نوار پخش فایل صوتی و گزینه‌های زیر است:
 - stop/play
 - next audio
 - previous audio
 - گزینه‌های لاگین/لاگ‌اوت و ثبت‌نام در بالای همه صفحات وجود دارند.
- همه صفحات به طور مشترک این بخش‌ها را دارند. در ادامه محتوایی که در قسمت Main و اصلی هر صفحه باید نمایش داده شود را شرح می‌دهیم:

• صفحه Home

اولین صفحه‌ای که برنامه نشان می‌دهد صفحه Home است. محتوای این صفحه برای افراد لاگین کرده «لیست فایل صوتی‌های پیشنهادی»؛ و برای افراد لاگین نکرده «لیست محبوب‌ترین فایل صوتی‌ها (بر اساس تعداد لایک)» می‌باشد.

- **صفحه ورود**

اطلاعات مورد نیاز برای ورود را دریافت کرده و در صورت صحیح بودن آن‌ها، وارد پنل کاربر می‌شود. در غیر این صورت در همان صفحه باقی می‌ماند.

- **صفحه ثبت نام**

اطلاعات مورد نیاز برای ثبت نام را دریافت کرده و در صورت موفقیت آمیز بودن یا نبودن عملیات، پیام مناسب را نمایش می‌دهد. در صورت موفقیت آمیز بودن عملیات به صفحه Home برمی‌گردیم. هنگام ثبت نام، کاربر نوع حساب کاربری خود را انتخاب می‌کند: خواننده، پادکستر یا شنونده. در صورتی که نوع حساب کاربری شنونده انتخاب شود، پس از ثبت نام موفقیت آمیز، صفحه انتخاب ژانرها نمایش داده شود.

- **صفحه انتخاب ژانرهای مورد علاقه**

در این صفحه یک جدول 3x3 از ژانرها نمایش داده می‌شود. کاربر حداکثر 4 ژانر مورد نظر خود را با کلیک انتخاب کرده و ثبت نام خود را تکمیل می‌کند. ژانرهای انتخاب شده را به نحوی متفاوت (مثلاً با رنگ border متفاوت) نمایش دهید. (در صورت انتخاب بیش از 4 ژانر خطا داده شود و در صورت انتخاب مجدد ژانری که قبلاً انتخاب شده، آن ژانر از انتخاب‌ها حذف شود.)

- **پنل شنونده**

در بخش اصلی این صفحه، اطلاعات حساب کاربری، لیست پلی لیست‌ها و امکانات شنونده (ساخت پلی لیست جدید، مشاهده following‌ها و تهیه اشتراک) نمایش داده می‌شود.

- **پنل ادمین (🌟 امتیازی)**

در این صفحه، نمودار آمار، و امکانات ادمین: مشاهده لیست آرتیست‌ها، فایل‌های صوتی و گزارش‌ها نمایش داده می‌شود.

- **پنل خواننده¹**

در این بخش، اطلاعات حساب، درآمد کسب شده، لیست آثار و امکانات خواننده: مشاهده دنبال‌کننده‌ها، میزان پلی شدن آثار به صورت نمودار و انتشار آهنگ قرار دارد.

¹ نیازی به پیاده‌سازی پنل پادکستر نیست.

- **صفحه پخش موزیک**

در این صفحه، یک فایل صوتی را پخش و اطلاعات آن را مشاهده می‌کنیم. بخش اصلی این صفحه شامل: قسمت پخش فایل صوتی (اطلاعات فایل صوتی و کاور)، متن آهنگ یا کپشن پادکست، امکان لایک و افزودن به پلی لیست می‌باشد.

- **صفحه مشاهده فایل صوتی‌های یک پلی لیست**

در این بخش، فایل‌های صوتی یک پلی لیست به صورت یک لیست نمایش داده می‌شود. با انتخاب هر یک از فایل‌های صوتی، وارد صفحه پخش آن می‌شویم.

- **صفحه لیست همه آرتیست‌ها**

با انتخاب یک آرتیست از این لیست، وارد صفحه اطلاعات آن آرتیست می‌شویم.

- **صفحه لیست همه فایل صوتی‌ها**

با انتخاب یک فایل صوتی، وارد صفحه پخش آن فایل صوتی می‌شویم.

- **صفحه اطلاعات آرتیست**

در این صفحه، اطلاعات آرتیست و لیست آثار او نمایش داده می‌شود. همچنین گزینه‌های گزارش و دنبال کردن را برای کاربری که اطلاعات این آرتیست را مشاهده می‌کند، در این بخش قرار بدهید.

- **صفحه خرید یا تمدید اشتراک**

در این صفحه، بسته‌های اشتراک به کاربر نمایش داده می‌شود. با انتخاب یکی از بسته‌ها و تایید خرید، در صورت کافی بودن اعتبار حساب، عملیات با موفقیت انجام می‌شود. در غیر این صورت پیام خطا نمایش داده شود.

- **صفحه انتشار آهنگ**

گزینه انتشار آهنگ در پنل خواننده وجود دارد. با انتخاب این گزینه، وارد صفحه انتشار آهنگ می‌شویم که اطلاعات آهنگ را دریافت کرده و آن را به آثار آن خواننده اضافه می‌کند. (نیازی به ساخت آلبوم برای آهنگ نیست).

نکات قابل توجه:

1. هر صفحه باید یک گزینه Back برای برگشت به صفحه قبل از خود داشته باشد.

2. پس از انجام هر کدام از عملیات، باید نتیجه آن (موفقیت‌آمیز بودن یا خطا داشتن) نمایش داده شود.
3. در هر بخش از برنامه، در صورتی که روی یک فایل صوتی کلیک کنیم، باید به صفحه پخش آن منتقل شویم. همچنین نوار پخش در پایین صفحات، باید با اطلاعات آن فایل صوتی آپدیت شود.
4. پخش ادامه یک فایل صوتی در همه صفحات، قابلیت ✨ امتیازی است. در حالت عادی نیازی به این کار نیست و صرفاً پخش همان فایل صوتی از ابتدا کافی است.



نکات پیاده‌سازی

در پیاده‌سازی پروژه، موارد زیر رعایت شود:

1. قبل از شروع فاز دو، حتماً اشکالات فاز یک را برطرف کنید.
2. کاور آهنگ‌ها و فایل آن‌ها، می‌تواند به صورت لوکال و روی سیستم خودتان (و آدرس‌دهی نسبی) یا از طریق لینک و به صورت آنلاین نمایش داده شود یا پخش شود. انتخاب روش به عهده خودتان است، اما نمایش یک تصویر به عنوان کاور و پخش فایل صوتی اجباری است.
3. اتصال صفحات به هم باید از طریق دکمه‌ها یا لینک‌ها باشد و کار کند. (نه این که هر صفحه را جدا اجرا بگیریم!)
4. طراحی ظاهر برنامه به عهده خلاقیت خودتان است. در صورت تمایل می‌توانید از CSS برای استایل دادن و زیباتر کردن ظاهر برنامه خود استفاده کنید.
5. استفاده از SceneBuilder اجباری نیست، ولی کار شما را راحت‌تر می‌کند. اما دقت کنید که بعضی از عملکردها باید در کد پیاده‌سازی شوند.



معیارهای ارزیابی

1. رعایت اصول کدنویسی تمیز، (خصوصاً استفاده نکردن از اسکرنر و پرینتر در کلاس‌های غیر از view و نداشتن متدهای طولانی و غیرهذفمند) بسیار ضروری خواهد بود و در صورت عدم رعایت از یک سوم تا نصف نمره کسر خواهد شد.
2. توجه کنید که بستر اصلی پیاده‌سازی پروژه در گیت‌هاب است و کامیت‌های درست در بازه‌های زمانی کوتاه، بخشی از معیار نمره‌دهی می‌باشد.
3. دقت کنید رعایت اصول شی‌گرایی خصوصاً کپسوله‌سازی متغیرها و پیاده‌سازی متدهای getter و setter بخشی از معیار نمره‌دهی می‌باشد.
4. برای ارائه، فایلی شامل دستوراتی برای ایجاد ادمین، کاربرها و فایل‌های صوتی و دستوراتی برای تست همه متدها تهیه کنید تا هنگام ارائه درستی عملکرد توابع بررسی شود.
5. جداسازی منطقی فایل‌های پروژه توسط پکیج‌ها جزو معیارهای نمره‌دهی خواهد بود.
6. هنگام ارائه پروژه، از شما سوالاتی درباره نحوه پیاده‌سازی بخش‌های مختلف پروژه و مفاهیم مورد استفاده در برنامه پرسیده خواهد شد. لذا بهتر است با آمادگی قبلی و تسلط کافی روی کد، در جلسه ارائه حاضر شوید.

در پایان، امیدواریم موفق باشید و از مسیر لذت ببرید. ❤️