



پروژه موزیک پلیر فاز اول

برنامه نویسی پیشرفته
دکتر رضا رضانی

بهار 1403

تاریخ تحویل

1403/1/15

طراحان پروژه

ثنا سبحانی
حسین احمدی
مهرآفرین صیرفیان

زهرا معصومی
یونس جمشیدی
مهدی مظاهری

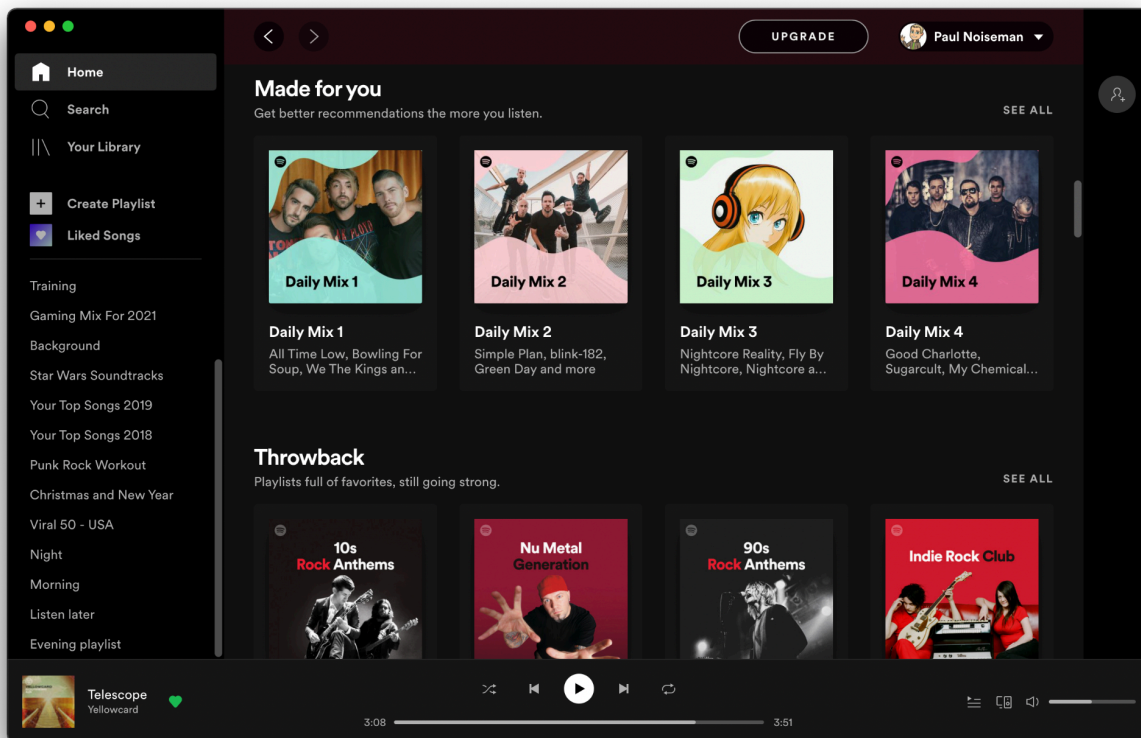


فهرست

| | |
|---------|-------------------------------------------------|
| 2..... | معرفی پروژه |
| 3..... | تاریخ تحویل و نکات مورد توجه |
| 4..... | شرح پروژه |
| 4..... | (A) شمای کلی موجودیت‌ها و سلسله مراتب ارث‌بری |
| 5..... | (B) ویژگی‌های موجودیت‌ها |
| 8..... | (C) عملکردها و عملیات تعریف شده برای موجودیت‌ها |
| 9..... | (D) توضیحات عملکردها |
| 11..... | (E) روند اجرای برنامه |
| 13..... | لیست دستورها |
| 13..... | کامندهای شنونده |
| 14..... | کامندهای ادمین |
| 14..... | کامندهای آرتیست |
| 15..... | نکات پیاده‌سازی |
| 16..... | 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 این پروژه، سنجش میزان یادگیری شما در زمینه مفاهیم شی‌گرایی 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 پروژه‌های 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، 15 فروردین 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 (محصورسازی، ارث‌بری و چندریختی) 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

در این بخش، اجزای پروژه را شرح می‌دهیم؛ لطفاً با دقت کافی و توجه به جزئیات این بخش را مطالعه کنید.

(A) شمای کلی موجودیت‌ها و سلسله مراتب ارث‌بری

در این پروژه، موجودیت‌های زیر استفاده خواهند شد. در صورت نیاز با حفظ ساختار پروژه می‌توانید موجودیت‌های خود را نیز تعریف کنید. در ادامه ویژگی‌های مورد نیاز هر موجودیت ذکر خواهد شد.

- پایگاه داده (Database): این کلاس Singleton است.
- فایل صوتی (Audio): از این کلاس نمی‌توان نمونه ساخت.
 - پادکست (Podcast)
 - آهنگ (Music)
- پلی‌لیست (Playlist)
- آلبوم (Album)
- حساب کاربری (User Account): از این کلاس نمی‌توان نمونه ساخت.
 - ادمین (Admin): این کلاس Singleton است.
 - شنونده (Listener)
 - عادی (Free)
 - پرمیوم (Premium)
 - آرتیست (Artist)
 - پادکستر (Podcaster)
 - خواننده (Singer)
- گزارش (Report)
- ژانر (Genre) با مقادیر زیر:
 - Rock
 - Pop
 - Jazz
 - HipHop
 - Country
 - True Crime

- Society
- Interview
- History

● بسته‌های اشتراک پرمیوم با مقادیر زیر:

- 30 days (\$5)
- 60 days (\$9)
- 180 days (\$14)

(B) ویژگی‌های موجودیت‌ها

در لیست زیر برخی از فیلدهای مورد نیاز موجودیت‌های برنامه ذکر شده‌است. در صورت نیاز (با حفظ نقش بقیه ویژگی‌ها و متدها) می‌توانید موارد دیگری را نیز به آنها اضافه کنید.

● کلاس Audio

- شناسه عددی یکتا
- نام/عنوان فایل صوتی
- نام آرتیست (رشته)
- تعداد پلی شدن
- تعداد لایک‌ها
- تاریخ انتشار (از نوع Date)
- ژانر (از نوع Genre)
- لینک فایل صوتی (رشته)
- کاور (رشته)

● کلاس پادکست

- کپشن

● کلاس آهنگ

- متن آهنگ

● کلاس آلبوم

- شناسه عددی یکتا
- نام

- نام خواننده (رشته)
- لیست آهنگ‌ها

● کلاس پلی لیست

- شناسه
- نام پلی لیست
- نام کاربر ایجاد کننده (رشته)
- لیست فایل صوتی‌ها

● کلاس کاربر

- نام کاربری یکتا (رشته)
- رمز عبور (رشته)
- نام و نام خانوادگی
- ایمیل
- شماره تلفن (رشته)
- تاریخ تولد (از نوع Date)

● کلاس ادمین (Singleton)

این کلاس ویژگی خاصی ندارد و ویژگی‌های مورد نیازش را از کلاس پدر به ارث می‌برد.

● کلاس آرتیست

- درآمد (عدد اعشاری)
- لیست دنبال‌کننده‌ها (از نوع کلاس حساب کاربری)
- بیوگرافی

● کلاس پادکستر

- لیست پادکست‌ها

● کلاس خواننده

- لیست آلبوم‌ها

● کلاس شنونده

- اعتبار حساب
- لیست پلی لیست‌ها
- تعداد پلی کردن فایل‌های صوتی توسط کاربر (از نوع Map)

- تاریخ اتمام اشتراک (از نوع کلاس Date - در صورتی که حساب شنونده عادی است، این فیلد null خواهد بود.)
- ژانرهای مورد علاقه (از نوع Genre)

● کلاس شنونده عادی

- محدودیت افزودن آهنگ به یک پلی لیست (عدد غیر قابل تغییر و مشترک بین همه نمونه‌ها)
- محدودیت تعداد پلی لیست‌های ساخته شده (عدد غیر قابل تغییر و مشترک بین همه نمونه‌ها)

● کلاس شنونده پرمیوم

- تعداد روزهای باقی مانده اشتراک

● کلاس گزارش

- کاربر گزارش دهنده
- آرتیست گزارش شده
- توضیحات

● کلاس دیتابیس Database

- این کلاس تمام اطلاعات مربوط به برنامه را نگهداری می‌کند و از الگوی Singleton پیروی می‌کند.
- لیست همه کاربرها
- لیست همه فایل صوتی‌ها
- لیست گزارش‌ها

C) عملکردها و عملیات تعریف شده برای موجودیت‌ها

شنونده (عملیات مشترک بین هر دو نوع شنونده):

- ثبت نام و ورود به پنل کاربری
- ایجاد پلی لیست (متفاوت برای عادی و پرمیوم)
- افزودن فایل صوتی به پلی لیست موجود (متفاوت برای عادی و پرمیوم)
- پخش فایل صوتی
- لایک فایل صوتی
- جستجو، مرتب‌سازی و فیلتر فایل‌های صوتی (🌟 امتیازی: استفاده از Stream)
- مشاهده following‌ها
- گزارش آرتیست
- مشاهده لیست آرتیست‌ها و اطلاعات و آثار یک آرتیست
- دنبال کردن آرتیست
- مشاهده پلی لیست‌های ساخته شده و انتخاب یک پلی لیست و مشاهده محتویات آن
- مشاهده فایل صوتی‌های پیشنهاد شده به کاربر با توجه به ژانر و آرتیست مورد علاقه او
- مشاهده اطلاعات حساب کاربری
- خرید یا تمدید اشتراک (متفاوت برای عادی و پرمیوم) و افزایش اعتبار حساب کاربری

ادمین:

- ورود به پنل کاربری
- مشاهده آمار محبوب‌ترین فایل صوتی‌ها بر اساس تعداد لایک
- مشاهده اطلاعات آرتیست‌ها و فایل صوتی‌ها
- مشاهده گزارش‌ها
- مشاهده اطلاعات حساب کاربری

آرتیست:

- ثبت نام و ورود به پنل کاربری
- مشاهده دنبال‌کننده‌ها
- مشاهده میزان پلی شدن آثار
- محاسبه درآمد توجه به نوع کلاس (پادکستر یا خواننده)
- مشاهده اطلاعات حساب کاربری

پادکستر:

- انتشار پادکست

خواننده:

- ایجاد آلبوم جدید
- انتشار آهنگ

(D) توضیحات عملکردها

در این بخش برخی از متدهای ذکر شده در قسمت قبل را شرح می‌دهیم. پیاده‌سازی این متدها باید لزوماً مطابق این مستند باشد، ولی پیاده‌سازی مواردی که در این قسمت ذکر نمی‌شوند آزاد و به عهدهٔ خلاقیت خودتان گذاشته شده است¹.

- **ثبت نام کاربر جدید:** پس از دریافت اطلاعات لازم، در صورتی که حساب کاربری شنونده بود، ژانرهای موجود به او نمایش داده شود و کاربر حداکثر 4 ژانر مورد علاقهٔ خود را انتخاب کند. همچنین در صورت تکراری بودن نام کاربری، پیغام خطا نمایش داده شود. همه حساب‌های کاربری شنونده در ابتدا شنونده عادی هستند و بعداً می‌توانند اشتراک پرمیوم خریداری کنند. همچنین این برنامه در هنگام ثبت نام هر شنونده، به او \$50 اعتبار هدیه می‌دهد.
- ایمیل و شماره تلفن وارد شده باید فرمت درستی داشته باشند. با استفاده از Regex این موارد را اعتبارسنجی کنید و در صورت معتبر نبودن، پیغام خطا نمایش دهید.
- ✨ امتیازی: تشخیص قوی یا ضعیف بودن رمز عبور با Regex
- **ورود به حساب کاربری:** با دریافت نام کاربری و رمز عبور، در صورتی که کاربری با این نام کاربری و رمز عبور وجود داشت، وارد پنل کاربری مربوط به خود شود. در غیر این صورت پیام خطا (عدم وجود نام کاربری یا رمز عبور اشتباه) نمایش داده شود.
- **جستجو در فایل‌های صوتی:** این عملیات با توجه به اسم فایل‌های صوتی یا نام آرتیست انجام می‌شود. برنامهٔ شما باید این قابلیت را داشته باشد که عبارت وارد شده توسط کاربر را هم در نام آرتیست و هم در نام فایل صوتی جستجو کند و همهٔ نتایج را نمایش دهد.
- **مرتب‌سازی فایل‌های صوتی:** این عملیات بر اساس موارد زیر انجام می‌شود:
 - محبوب‌ترین‌ها (بر اساس تعداد لایک)
 - بیشترین تعداد پلی شدن
- **فیلتر فایل‌های صوتی:** این عملیات بر اساس موارد زیر می‌تواند انجام شود:
 - آرتیست
 - ژانر

¹ دقت کنید که دقیقاً عملکرد خواسته شده را پیاده‌سازی کنید، خلاقیت در نحوهٔ پیاده‌سازی آزاد است نه تغییر کلی یک عملیات!

- تاریخ انتشار (🌟 امتیازی: فیلتر فایل‌های صوتی منتشر شده بین دو تاریخ)
- **پیشنهاد آهنگ به کاربر:** این متد با توجه به ژانر فایل‌های صوتی‌ای که تا کنون بیشتر از بقیه توسط این کاربر پلی شده‌اند، فایل‌های صوتی‌ای که لایک کرده است و آرتیست‌هایی که دنبال کرده است، 10 فایل صوتی که احتمالاً بیشترین تطابق با سلیقه کاربر را دارند، به او پیشنهاد می‌دهد. نحوه پیاده‌سازی این متد آزاد است و با خلاقیت خودتان انجام شود. همچنین در پیاده‌سازی این متد باید از ژانرهای مورد علاقه کاربر که هنگام ثبت نام انتخاب کرده است، استفاده کنید.
- **ایجاد پلی‌لیست:** نام پلی‌لیست دریافت و پلی‌لیستی خالی ساخته می‌شود. (در صورت نیاز به افزودن فایل صوتی به پلی‌لیست، از گزینه مربوطه در منوی شنونده استفاده شود)
این عملیات برای کاربر عادی و پرمیوم متفاوت است. کاربر پرمیوم محدودیتی در تعداد ساخت پلی‌لیست‌ها ندارد ولی کاربر عادی تنها 3 پلی‌لیست می‌تواند ایجاد کند. در غیر این صورت پیغام خطا نمایش داده می‌شود. (این عملیات باید با استفاده از **چندریختی** پیاده‌سازی شود).
- **افزودن فایل صوتی به پلی‌لیست:** با توجه به نوع حساب شنونده، این عملیات متفاوت خواهد بود. شنونده عادی به هر پلی‌لیست تنها 10 فایل صوتی می‌تواند اضافه کند، ولی کاربر پرمیوم محدودیتی ندارد. (این عملیات باید با استفاده از **چندریختی** پیاده‌سازی شود).
- **خرید یا تمدید اشتراک:** به کاربر بسته‌های اشتراک پرمیوم نمایش داده می‌شود: 30 روزه، 60 روزه و 180 روزه. کاربر با انتخاب یکی از این موارد، در صورت کافی بودن اعتبار حساب اشتراک را تهیه می‌کند. پس از پرداخت، اگر کاربر شنونده عادی باشد به کاربر پرمیوم تبدیل می‌شود (حذف و ایجاد کلاس جدید با اطلاعات قبلی ولی از نوع شنونده پرمیوم و تنظیم فیلدهای آن) و در صورتی که کاربر پرمیوم باشد، تعداد روزهای باقی‌مانده اشتراک و تاریخ آن آپدیت می‌شود. همچنین پس از هر بار مشاهده اشتراک، این دو فیلد به‌روزرسانی شده و یک روز از اشتراک کم می‌شود! (==))
- (🌟 امتیازی: کم شدن از روزهای اشتراک با استفاده از **Task Scheduling**)
- **افزایش اعتبار حساب کاربری:** کاربر مقداری که می‌خواهد به اعتبار حسابش افزوده شود را وارد کرده و پس از پرداخت هزینه (نیازی به پیاده‌سازی این بخش نیست)، مقدار مورد نظر به اعتبار حسابش افزوده می‌شود.
- **انتشار فایل صوتی:** آرتیست اطلاعات فایل صوتی را وارد کرده و آن را منتشر می‌کند. دقت کنید که هر آهنگی که توسط یک خواننده منتشر می‌شود، به یک آلبوم تعلق دارد ولی پادکست‌ها نیازی به این کار ندارند.
- **محاسبه و به‌روزرسانی درآمد آرتیست:** اگر کاربر گزینه نمایش اطلاعات حساب کاربری را انتخاب کند، باید قبل از نمایش اطلاعات، درآمد او به‌روزرسانی شود. به این صورت که با توجه به آثار آن آرتیست و تعداد پلی شدن آنها، درآمد محاسبه شود: (این عملیات باید با استفاده از **چندریختی** پیاده‌سازی شود)
- به ازای هر بار پلی شدن پادکست، به درآمد پادکستر 0.5 دلار اضافه می‌شود.

- به ازای هر بار پلی شدن آهنگ، به درآمد خواننده 0.4 دلار اضافه می‌شود.

(E) روند اجرای برنامه

1. در شروع برنامه، ادمین به صورت یکتا در برنامه ساخته می‌شود.
2. اولین عملیات برنامه یکی از گزینه های ورود یا ثبت نام است.
3. عملیات ثبت نام با توجه به نوع کاربری که می خواهد ثبت نام کند انجام می‌شود.
4. عملیات ورود با دریافت نام کاربری و رمز عبور و بررسی آنها انجام می‌شود.

عملیاتی که برای موجودیت‌ها ذکر شد، به صورت دستوراتی در کنسول وارد می‌شوند. (مانند وارد کردن دستورات در محیط CMD یا Terminal) مثلاً برای ثبت نام از دستوری با فرمت زیر استفاده می‌کنیم:

```
Signup -[email] -[name] -[password] -[username] ...
```

در این دستور، NewListener مشخص می‌کند چه کاری می‌خواهیم انجام دهیم و در ادامه خط، بقیه مشخصات مورد نیاز برای ساخت حساب وارد می‌شود.
نمونه استفاده از این دستور برای ساخت حساب شنونده:

```
Signup -rezaR -12345678 -RezaRamezani -reza@gmail.com ...
```

پس از وارد کردن این دستور، برنامه شما با توجه به نوع دستور عملیات مورد نظر را انجام داده و در صورتی که خروجی داشته باشد، آن را در صفحه نمایش می‌دهد. همچنین در صورت وجود خطا، متن آن نمایش داده شود.

همچنین دقت کنید که عملیات هر نوع کاربر، باید تنها زمانی انجام شود که آن کاربر در سیستم Login کرده باشد و عملیاتی که انجام می‌دهد جزو دستورات مجاز آن نوع کاربر باشد، در غیر این صورت پیام خطا نمایش داده شود. (مثلاً کاربر شنونده نباید بتواند به اجرای دستورات ادمین دسترسی داشته باشد و غیره). در هر مرحله از برنامه، تعدادی عملیات مجاز می‌توان انجام داد، در صورتی که دستوری غیر از آنها وارد شود، پیام خطا چاپ کنید.

در ادامه عملیات موجود در هر نوع حساب کاربری (پس از ورود به حساب) را در یک نگاه آورده‌ایم:

پنل شنونده²:

شامل 4 بخش زیر است: (با انتخاب هر گزینه منوی مربوط به آن نمایش داده می‌شود)

1. مشاهده فایل‌های صوتی پیشنهادی (10 فایل صوتی پیشنهاد شده توسط برنامه، نمایش داده شوند)

² عملیات برای هر دو نوع حساب کاربری یکسان است، تفاوت‌ها در عملیات با چندریختی پیاده‌سازی می‌شوند.

2. مشاهده آرتیست‌ها
 - a. در صورت انتخاب یک آرتیست، اطلاعات و لیست آثار او نمایش داده شود.
 - b. امکان دنبال کردن آرتیست
3. جستجو، مرتب‌سازی و فیلتر فایل‌های صوتی
 - a. افزودن فایل صوتی به پلی‌لیست
 - b. انتخاب فایل صوتی برای پخش (نمایش اطلاعات آهنگ)
 - i. لایک فایل صوتی
 - ii. مشاهده متن آهنگ
4. بخش Library:
 - a. مشاهده پلی‌لیست‌ها
 - i. انتخاب پلی‌لیست و مشاهده فایل‌های صوتی داخل آن
 1. انتخاب فایل صوتی برای پخش (نمایش اطلاعات آهنگ)
 - a. لایک فایل صوتی
 - b. مشاهده متن آهنگ
 - b. ایجاد پلی‌لیست (متفاوت برای کاربر عادی و پرمیوم)
 - c. افزودن فایل صوتی به پلی‌لیست موجود
 - d. مشاهده following‌ها
 - i. گزارش آرتیست
5. مشاهده اطلاعات حساب کاربری
 - a. افزایش اعتبار حساب کاربری
 - b. خرید یا تمدید اشتراک

پنل ادمین:

1. مشاهده آمار محبوب‌ترین فایل صوتی‌ها (تعداد لایک)
2. مشاهده لیست فایل صوتی‌ها و اطلاعات آنها در صورت انتخاب
3. مشاهده لیست آرتیست‌ها و اطلاعات آنها در صورت انتخاب
4. مشاهده گزارش‌ها

پنل پادکستر و خواننده³:

1. مشاهده دنبال‌کننده‌ها
2. مشاهده میزان پلی شدن آثار

³ موارد ذکر شده بین هر دو پنل پادکستر و خواننده یکسان است. تنها تفاوت این دو در انتشار موزیک یا پادکست و ایجاد آلبوم برای خواننده می‌باشد.

3. محاسبه درآمد (متفاوت برای پادکستر و خواننده)

4. مشاهده اطلاعات حساب کاربری

5. ایجاد آلبوم (خواننده)

6. انتشار پادکست یا آهنگ

دستور ثبت نام (شنونده و آرتیست)، ورود و مشاهده اطلاعات حساب نیز برای تمام حساب کاربری‌ها قابل اجرا است.

لیست دستورها⁴:

1. Signup -L|S|P -[username] -[password] -[name] -[email] -[phone number] -[birth date] -[bio⁵]
 - new user: **L**istener or **S**inger or **P**odcaster
2. FavouriteGenres -[favourite genres separated with comma(,)]
 - only for listeners, after signup
3. Login -[username] -[password]
4. Logout
5. AccountInfo

کامندهای شنونده

6. GetSuggestions

✨ امتیازی: دستور بالا را به صورت GetSuggestion +n که n تعداد فایل صوتی‌های پیشنهادی مورد نظر را نشان می‌دهد و توسط کاربر وارد می‌شود، پیاده‌سازی کنید. پس از اجرای این دستور، n فایل صوتی‌ای که بیشترین تطابق با سلیق کاربر را دارند، نمایش داده می‌شوند.

7. Artists
8. Artist -[username]
9. Follow -[username]
10. Search -[artist name OR audio's title]
11. Sort -L|P
 - (based on **L**ikes or **P**lays)
12. Filter -A|G|D -[filter by]
 - (Filter by **A**rtist, **G**enre or **D**ate)

⁴ بخش‌های هر دستور با علامت - از هم جدا می‌شوند و مواردی که در [] ذکر شده‌اند، ورودی‌های آن دستور هستند و علامت | به معنای OR (تنها یکی از موارد ذکر شده استفاده شود) می‌باشد.

⁵ برای آرتیست‌ها فقط.

13. Add -[playlist's name] -[audio's ID]

کامندهای بخش Library

14. ShowPlaylists

15. SelectPlaylist -[playlist's name]

16. Play -[audio's ID]

17. Like -[audio's ID]

18. Lyric -[audio's ID]

19. NewPlaylist -[playlist's name]

20. Followings

21. Report -[artist's username] -[explanation]

22. IncreaseCredit -[value]

23. GetPremium -[package]

کامندهای آدمین

24. Statistics

25. Audios

❖ امتیازی: امکان مشاهده فایل‌های صوتی به صورت صفحه‌بندی شده - در صورت نیاز به دستور بالا آرگومان‌های دیگری اضافه کنید.)

26. Audio -[audio's ID]

27. Artists

28. Artist -[username]

29. Reports

کامندهای آرتیست

30. Followers

31. ViewsStatistics

32. CalculateEarnings

33. NewAlbum -[name]

34. Publish -M|P -[title] -[genre] -[lyric|caption] -[link] -[cover] -[album ID⁶]

- Publish **M**usic or **P**odcast

⁶ for musics



نکات پیاده‌سازی

در پیاده‌سازی پروژه، موارد زیر رعایت شود:

1. قبل از شروع به کد زدن پروژه، نمودار UML کلاس‌ها را طراحی کنید. (🌟 امتیازی)
2. برای همه کلاس‌های مدل (بجز Database و Enumها) متد toString نوشته شود. این متد در کلاس فرزند، باید از متد کلاس پدرش استفاده کند.
3. ایجاد شناسه‌های یکتا باید در برنامه انجام شود. شناسه‌ها در ورودی دریافت نمی‌شوند.
4. در محاسبه تاریخ‌ها و انجام عملیات روی آنها، باید از ویژگی‌های کلاس Date⁷ استفاده کنید.
5. در بخش‌هایی از پروژه که از لفظ «فایل صوتی» استفاده شده است، منظور هر دو نوع موزیک و پادکست است.
6. همهٔ دسترسی‌ها به فایل‌های صوتی، کاربرها، گزارش‌ها و ... باید از طریق کلاس Database انجام شود. همچنین تغییرات مورد نیاز نیز روی همین نمونه‌ها باید انجام شود.
7. تشخیص درست نوع متغیرها و فیلدهایی که در این مستند ذکر نشده‌اند، بر عهدهٔ شماست.

⁷ استفاده از کلاس‌های کمکی جاوا که به همین منظور پیاده‌سازی شده‌اند مجاز است.



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. ❤️