

## نوشتن API برای یک اسکرپر موزیک

### سناریو:

شما عاشق موسیقی هستید و همیشه آخرین آهنگ را می‌خواهید. پس یک اسکرپر برای برای گرفتن جدیدترین موزیک از سایت موردعلاقه خودتون نوشتید. یه بار که یکی از دوستانتون اسکرپر رو میبیند، ازتون میخواد که یه ورژن مشابه برای اونم انجام بدید. شما که میدونید دوستتون زیاد اهل برنامه نویسی نیست و احتمالاً اگه کد رو هم بهش بدید نمیتونه ازش سر دربیاره راه ساده‌تر رو پیش میگیرید: اسکرپر خودتون رو به یک API تبدیل می‌کنید که هر کس دوست داشت بتونه جدیدترین موزیک رو دریافت کنه.

### اهداف فاز ۱:

- استخراج داده‌ها از طریق وب اسکرپینگ
- طراحی ویوهای جنگو و ایجاد یک REST API پایه
- حافظه cache برای بهبود عملکرد

### وظایف فاز ۱:

#### اسکرپ کردن وب سایت موسیقی

- یک اسکرپر بسازید که جدیدترین آهنگ‌ها رو از یک وب سایت موسیقی مشخص دریافت کند. (هر نفر یک وب سایت انتخاب کنه)
- داده‌های اسکرپ شده رو بر اساس دسته‌بندی‌هایی مانند ژانر، هنرمند یا تاریخ انتشار سازماندهی کنید.
- خطاهای مهم، مثل مشکلات شبکه یا فیلدهای خالی داده رو مدیریت کنید.

#### ایجاد ویوهای جنگو

- یک پروژه جنگو راه‌اندازی کنید و ویوهای طراحی کنید که داده‌های اسکرپ شده (مانند آهنگ، دسته) سرو کنه.
- توی ویوها و فیلدهاتون انعطاف‌پذیری حداقلی داشته باشید. یعنی مثلاً برای موزیک سه یا چهارتا فیلد اساسی (کاور، لینک دانلود، اسم، مدت زمان و...) کافیه

#### ساخت API با Django Rest Framework (DRF)

- اندپوینت‌های API ایجاد کنید تا آخرین آهنگ‌ها رو بر اساس دسته‌بندی فهرست کند.
- اندپوینت دیگری ایجاد کنید که اطلاعات دقیق یک آهنگ خاص رو بازبایی می‌کند.

## اهداف فاز ۲:

- اتصال Redis به عنوان کش بک اند
- ایمن سازی API با احراز هویت
- اجرای throttle برای جلوگیری از سوء استفاده

## وظایف فاز ۲:

### ذخیره سازی با Redis

- Redis caching را برای ذخیره آخرین داده های موسیقی به مدت یک ساعت برای کاهش بار روی سرور پیاده سازی کنید.
- ما از شما می خواهیم اطمینان حاصل کنید که داده ها به طور خودکار در حافظه کش زمانی که منقضی می شوند یا به روزرسانی های جدید وجود دارد، به روزرسانی می شوند.

### پیاده سازی احراز هویت پایه

- API را با استفاده از Basic Authentication ایمن کنید.
- مطمئن شوید که کاربران غیرمجاز نمی تونن به API دسترسی داشته باشند.

## تروتلینگ

- تعداد دفعاتی را که کاربر می تواند در یک بازه زمانی معین به API ریکوئست ارسال کنید، تنظیم کنید.
- قوانین throttling را برای جلوگیری از اضافه بار سرور و محافظت در برابر استفاده سوء استفاده از API طراحی کنید.

**نکته ۱:** توجه کنید که فازبندی ها برای راحتی کار شما انجام شده. در صورت صلاح دید خودتون میتونید اونها رو تغییر بدید (نمره بابت تحویل تسک ها در فازهای نامربوط به خودشون کم نمیشه)

**نکته ۲:** در صورت مشاهده کدهای خیلی مشابه به هم نمره پروژه انفرادی شما صفر خواهد بود

**نکته ۳:** لطفا فقط یک سایت رو برای اسکرپ انتخاب کنید. در انتخاب تکنولوژی آزادیید اما توصیه میشه از BS (beautiful soup) برای اسکرپ استفاده کنید.