

Ciência de Dados - PL

Professor: Jorge Louçã

UC: Interfaces Web para a Gestão de Dados

3º ano | ano letivo: 2021 - 2022

26/10/2021

Relatório sobre a Aplicação Web:

MCDb – Music Collection Database

Trabalho realizado por:

Catarina Castanheira, 92478

João Martins, 93259

Joel Paula, 93392

Objetivo e Visão Geral da Aplicação Web:

O objetivo geral da aplicação é disponibilizar aos seus utilizadores procurar de forma simples e intuitiva por artistas e respetivos álbuns e adicionar estes últimos a coleções que eles próprios tenham criado. Além desta funcionalidade, que no fundo permite a gestão de bibliotecas de álbuns musicais, os utilizadores poderão facilmente escrever *reviews*, fazer ratings nos seus álbuns e também consultar médias de classificação de *reviews* de outros utilizadores. Aquilo que torna esta aplicação uma aplicação simples e ágil é o facto de ir buscar informação a uma base de dados externa disponível publicamente – Discogs (<https://www.discogs.com>) – não existindo assim a necessidade de inserir dados sobre os álbuns manualmente.

Pontos fortes da aplicação: uso de uma API para integrar informação externa sobre álbuns e artistas; um sistema de login que, usando somente o *decorator @login_required* com a devida construção de *forms* permite que a gestão de acessos de utilizadores seja feita “autonomamente”, sem necessidade de que nós próprios façamos esta configuração dentro de cada *view*; recurso a classes do Bootstrap para configurar o *layout* e grafismo das diversas páginas.

Modelos para a Base de Dados (consultar diagrama em Anexo):

- User
- Collection
- Artist
- Record
- Collection_Record
- Rating
- Like_Artist
- Staff_Picks

Funcionalidades:

1. Pesquisar Artistas/Álbuns (Funcionalidade Pública):

URL: */search*

Pesquisa de texto livre e devolve os cinco artistas e os dez álbuns mais relacionados.

2. Consultar Página de Artistas/Álbuns (Funcionalidade Pública):

URL: */artist/<int:artist_id>*

/record/<int:record_id>

Consulta de informação sobre um artista ou um álbum previamente importados do Discogs.

3. Coleção de Álbuns (Consulta pública e edição restrita a Utilizadores Autenticados):

URL: */mycollections/<str:username>*

Criar uma coleção onde podemos adicionar álbuns.

Adicionar/Remover Álbuns de dada Coleção.

4. Criar/editar Ratings/Reviews de Álbuns (Restrita a Utilizadores Autenticados):

URL: `/review/<int:review_id>`

Criar e editar *ratings/reviews* (de um a cinco estrelas) para álbuns. Os ratings/reviews serão públicos, mas a sua edição é restrita ao utilizador que a criou.

5. “Gostar” de Artista e Consultar My Artists (Restrita a Utilizadores Autenticados):

URL: `/artist/<int:artist_id>/like`

`/artist/<int:artist_id>/unlike`

O utilizador consegue fazer “like” na página de um artista específico. Na página My Artists, o utilizador poderá consultar a lista dos artistas aos quais já fez “like”.

6. Comunidade

Partilha e Consulta de Coleções (Funcionalidade Pública):

URL: `/mycollections/<str:username>`

`/mycollections/<str:username>/<int:collection_id>`

Uma vez que o URL de dada coleção é inteligível (tem o *username* declarado e uma *id* específica da coleção em causa), e uma vez que as coleções de qualquer utilizador não têm restrições na consulta, um utilizador pode facilmente copiar o URL e partilhá-lo externamente.

Partilha e Consulta de Reviews/Ratings (Funcionalidade Pública):

URL: `/review/<int:review_id>`

`/reviews`

De forma similar ao caso da partilha de coleções, a página que contém *reviews* e *ratings* sobre dado álbum também é pública para consulta. Desta forma é também possível a partilha através do URL.

Staff Picks (Definição é Restrita a Superusers Autenticados; Consulta é Funcionalidade Pública):

URL: `/`

A página de início da aplicação web tem visível um campo com aqueles álbuns que foram selecionados pelos *superusers* – os ditos *staff picks*. Qualquer utilizador não autenticado consegue ver este campo. Contudo a escolha dos álbuns que o integram é só permitida a *superusers*.

Top Artists (os artistas mais bem cotados pela comunidade de utilizadores da MCDb) (Funcionalidade Pública):

URL: `/`

Consiste num campo na página inicial, visível a qualquer utilizador, com uma listagem dos artistas com melhor média dos ratings dos álbuns criados pelos utilizadores registados.

Top Records (os álbuns mais bem cotados pela comunidade de utilizadores da MCDb) (Funcionalidade Pública):

URL: /

Funcionalidade bastante semelhante à anterior, mas lista os álbuns com as médias de *ratings* mais elevadas de todos aqueles avaliados por utilizadores registados.

Latest Reviews (as reviews mais recentes de álbuns criadas pela comunidade de utilizadores da MCDb) (Funcionalidade Pública):

URL: /

Este campo contém, também acessível a qualquer tipo de utilizador, uma listagem das últimas *reviews* criadas por utilizadores registados.

7. Login / Logout / Registo de Utilizador

URL: /register

Sistema simples de gestão de utilizadores. Permite registar utilizadores novos, fazer *login* e *logout*.

8. Youtube Videos (Funcionalidade Pública):

A partir de uma página de um álbum, é disponibilizada uma lista de links que redirecionam para páginas externas do Youtube, contendo vídeos relacionados com o álbum.

Livrarias Externas Utilizadas, Recursos não programados pelo grupo:

1 – **Levenshtein**: para ordenar resultados da pesquisa (<https://github.com/ztane/python-Levenshtein>);

2 – **requests**: para fazer pedidos a API REST do Discogs (<https://docs.python-requests.org/en/latest/>);

3 – **django-widget-tweaks**: para adicionar classes de *bootstrap* aos *forms* do Django (*login*, etc.) (<https://github.com/jazzband/django-widget-tweaks>);

4 – **Django authentication**: para criar o sistema de registo, autenticação e recuperação de *passwords*. Requereu somente criar os *templates* dos *forms* (em `music_website\home\templates\registration`) (<https://docs.djangoproject.com/en/3.1/topics/auth/customizing/>);

5 – **Django pagination**: para controlar o número de registos a serem extraídos (máximo de 25 de cada vez) da base de dados (para o funcionamento do campo de Staff Picks na página raiz) (<https://docs.djangoproject.com/en/3.2/topics/pagination/>);

6 – **Bootstrap**: estilos e validações para *forms* e outros controlos, HTML *cards*, etc. (<https://getbootstrap.com/>);

7 – **Discogs API**: permite-nos extrair informação sobre artistas e álbuns da base de dados externa, a Discogs (<https://www.discogs.com/developers>).

Anexos

- Estrutura da Base de Dados

