Film web-app

ADVANCED WEB TECHNOLOGIES

Sergio Fernández Rodríguez
40220468 | EDINBURGH NAPIER UNIVERSITY

Introduction

The web-app developed is a catalogue of films and actors. On the home page you can see the main menu and choose whether you want to access films or people.

By accessing movies, today's most popular films are shown, films which will be launched soon and the most valued by users. In each, you can access its trailer or information on its website, if these resources are available. You can also consult the cast of each film. By visiting the cast, all actors with a profile picture appear, with a short description and a link to Google to search for that actor.

When you are in film page you can find the film you want writing the title or the appropriate keywords in the search box located on the top right. Thus, films you want will appear, of which you can obtain the same information that you get from the most popular films, for example.

Moreover, you can access people section, in which the cast of one of the most popular films randomly appears. Just as films, you can also search the actors you want using the people search box, available in this section.

To build the application has been used tools such as Python, Flask, Jinja2, HTML, CSS or JavaScript.

Design

I have divided the architecture of the web-app into two sections: 'films' and 'people'.

The 'films' section is dedicated to exposing the most popular films, films which will be launched soon and the most valued by users. It uses the path '/films', and you can search any film you want using the search box. When looking for a film, the hierarchy looks like this: '/films/search/<searched>'.

Three features of all films are shown: trailer, official website and cast. Trailer and official website buttons redirect off the web-app. Upon entering the cast, all actors in the film with an introduction of his biography appear. The route in this case is as follows: '/films/<film>/cast'.

The case of 'people' section is similar, using the path '/people'. By accessing this section you will find the cast of one of the most popular films, randomly chosen among the top 6. We can also search, in this case to find actors and actresses. When you search, we will use the path '/films/search/<searched>', and all people which partially match will appear.

A short description and a link to Google to look for more information appear in the search results of actor, as in the case of the cast of a film.

To make the web-app, we have used a base template ('base.html') that is repeated on every page of the website. All other templates extend from this base template.

Finally, it has also been created the template for the 404 error, and can be tested with the path '/force404'.

Enhancement

The first thing that I would improve of the web application would be the main page. I would add global search feature, including films, actors and cinematographic studios.

Second, I would create a specific section for film studios. Thus, it would be possible to look for only studios.

Third, give support to television series, creating a template that will be adapted to them. You could also find only television series and a short description of each show will appear. In addition, when accessing one of them, it will show the number of seasons it has and chapters of every season. Of course, just as films, you could also consult its cast.

Fourth, create a section to sort films by genre. When you access it, all categories will appear and could select the one you want to search for films of this kind.

Finally, it is important to note that all software can always be improved, so that we could make many other improvements to this web-app. For example, you could build a login system using a database (SQLite) so that users could have their own account and bookmark their favourite films.

Critical evaluation

A critical evaluation of the web application can be very subjective, but I would take into account the following aspects in particular.

The web application is not simply a catalogue in which you search and then resulting images and titles appear, but the user has a more dynamic control of the results. You can get information directly and access their official trailer on YouTube. Also, you can access the cast of each film which, in turn, contains direct links to information of each actor.

Thus, the user is not a mere spectator but can interact with the page.

It is important to say that we start practicing with very simple aspects of Python and HTML, so that the web is significantly more advanced than studied. Obviously, you can improve it in many aspects like all software, and this has been detailed in the 'improvements' section. All these improvements would enrich the functionality of the web making it a fairly complete and enough to launch a first version at web market. Thus, we could test how the user gets on with the web-app before developing all the functions fully and getting direct feedback.

Personal evaluation

This work has helped me to learn and know a lot of aspects of Python and templates mainly.

A useful aspect has been everything related to redirections in Python and the correct use of the hierarchy. Thus, it has been achieved that all routes are properly organized.

The template for the graphic design of the site was very different from how it finally left, and I had to learn a lot of HTML tags which I did not know before. So, I managed to remove those parts of the web that seemed unnecessary. By this several problems arose since the web is out of position with some overlapped images. I got it solved looking at Google, and I found a way to design static pages as 'films'.

Moreover, I have also learned CSS, because I had to change some lines to change colours of texts and not merge with the background.

Summary

TMDB API
https://www.themoviedb.org

Tmdb3 resource
https://github.com/wagnerrp/pytmdb3

Advanced Web Technologies - Notes & Workbook 2015-2016

Website template
https://w3layouts.com/theater-entertainment-mobile-website-template/

Flask
http://flask.pocoo.org/

Python Documentation
https://www.python.org/doc/