PocketMovies Application

Group 6 TPW

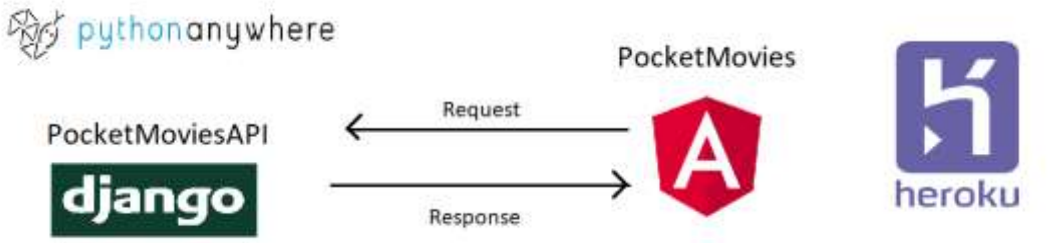
Óscar Sánchez(101631)

Marta Ferreira(88830)

Pedro Santos(93221)

GitHub URL: https://github.com/oscarfersan/PocketMovies_tpwAngular/

1.-Architecture



The app developed is formed by 2 main structures: an Angular app, where the user uses the frontend and manage the data through views, and a Django app, which uses Django Rest Framework. The interaction between them are very simple. The user make a request(POST,PUT,GET,DELETE), and in the RestAPI Django manage that request to modify the database. The RestAPI is deployed on pythonanywhere and the angular app on heroku.

API URL: http://pedrobear.pythonanywhere.com/

ANGULAR: https://pocketmovies.herokuapp.com/

2.-Actors of the App

The app consider 2 type of users:

-Clien: Regular user who wants to see the Information of a movie, director, actor or producer and classify the movies as Want to watch, watched and Favourite.

-Admin: Member of the staff who can make all the actions provided to a regular user, but this type of user also has permission to modify the database(e.g:Delete a Producer)

User Credentials:

- Regular Users:

    1. Username: oscar | Password: oscar

- Admin User:

    1. Username: admin | Password: admin

3.-Main functionalities

All users:

1. Register an Account;
    a. In the Angular App, the user fills correctly the form and the API returns the auth token to Access the API.
2. Login to/ Logout off an existing account;
    a. In Angular, the user fill the login form and the API allows him to see all the information of the API
    b. Exists a Logout button who allows the user to end the session

3. Reset user password;

4. Access Movies/ Actors/ Directors/ Producers pages

a. In the sidebar provided, a user can navigate and geta ll the movies, actors,directors and producers, fetching al lof them from the API passing the auth token and distinct parameters(e.g:/movies/all ,/movies/favourite)

5. Access individual information pages for Movies/ Actors/ Directors/ Producers;

a. Following a procedure like in the lists, the user can access the data of a certain object clicking on it and the API Will fetch it by its id(e.g/movies/info/1).

6. Add Movies to Favorites/ Want to Watch/ Have Watched lists;

a.In the ListMovies Page, the card shows different options of adding the movie to some of your lists. The API gets the request and modify that lists.

7. Access each one of those lists through the sidebar (My Movies);

Users with admin privileges:

10. Add new Movies/ Actors/ Directors/ Producers;

a.If the user is logged with staff permissions, will appear a button on the page that allows the user to add a new object. Filled the form, the API modify the database and creates the new object. When it´s done, the Angular app will notify you if the add was successfull.

11. Edit Movies/ Actors/ Directors/ Producers;

a.In the info page and in each one of the cards showed on the ListMovies page(only for movies), the user can access to the edit form, which will send the information to the API and this one will modify the object. When it´s done, the Angular app will notify you if the edit was successfull.

12. Delete Movies/ Actors/ Directors/ Producers.

a.As the edit method, the option will appear in the ListMovies page and in the info page of any object. Before deleting, the page will advice you if you are sure about deleting that object. In case of yes,when it´s done, the Angular app will notify you if the delete was successfull.