BigLab1: FilmLibrary in React

During the four weeks of the first BigLab, you will develop the front-end of the FilmLibrary application using React. To create your repository, you must login to GitHub Classroom and join (or create) your group. For more details, please have a look at the *GitHub Classroom instructions*. In the following lines you can find the links for the BigLab repositories on GitHub Classroom. Please, choose the link of **your** course.

- Applicazioni Web I: https://classroom.github.com/a/ieHJvLLd
- Web Applications I [A-J]: https://classroom.github.com/a/-qyvjfaH
- Web Applications I [K-Z]: https://classroom.github.com/a/1FGUxLDA

To better keep track of your progress, we *suggest* you work incrementally "week-by-week", e.g., by creating, inside your repository, a branch for each week of the BigLab.

Graphical User Interface components

Your webpage must contain all the elements already mentioned in laboratory 3 targeting a desktop browser. Specifically, a **navigation bar** (with a logo, a search box, and a user icon), a **sidebar** with the filters, the **main** content (i.e., the list of films), and a **"+" button** for adding new films.

Each film should be displayed in the following manner:

- The **title** of the Film. The title of the favorite films should be displayed in red.
- A **checkbox** to show and set whether the Film is a favorite (☑) or not (☐).
- The Film's **watch date** in the format "Month D, Yr" (e.g., "March 24, 2022"), if any.
- The **rating** of the Film is expressed between one and five **stars**. If the score is not assigned, five empty stars are visualized. If the score is three out of five, your GUI should display three filled stars and two empty stars.

Roadmap of the following weeks

a) During the first week, you will start **setting up the front-end** for the FilmLibrary. To do so, you will use the React-enabled version of Bootstrap. The page has to be static; you do not have to implement any user action. For the visual appearance of the web application, you can get inspiration from the screenshot in the next page.

b) In the second week, you will **re-structure** your web-based FilmLibrary **to exploit the component-based** approach of the React framework. Specifically, you will divide your application into different components, and you will identify the state and props required to store and visualize the films, starting from the films data structure. You will also implement the **filters** (i.e., Favorite, Best Rated, Seen Last Month, Unseen).

c) During the third week, you will add to your application the various actions to manipulate the FilmLibrary. For instance, you will include the possibility to **add** new **films** and to implement some **basic editing** functionalities to the FilmLibrary.

d) In the last week, you will learn how to restructure your application to use **routes** for supporting multiple "pages" by developing a different route for each filter. Furthermore, you will allow users to complete **edit** previously inserted **films**.

Evaluation Criteria & Deadlines

The point received for your work is added to the final exam score of each member of the team.
We will follow these evaluation criteria:

- All team members will receive 1 point if the submitted React application is *complete*. In other words, it successfully implements *all the functionalities of the 4 weeks*, i.e., points a), b), c), and d).
- All team members will receive 0.5 points if the submitted React application is *partially complete.* It is necessary to implement **at least:**
    - o all the functionalities of the first 2 weeks, i.e., points a) and b)
    - o one point of the other two weeks (e.g., *c1. Add new Film*)
- Team members will receive 0 points otherwise.

The assignment must be submitted in the main branch of the assigned group repository before **Sunday, May 8, 2022, at 23:59 CEST** (see the the GitHub Classroom instructions for details).

Example Screenshot

Note: the actual appearance of your application may be different from the screenshot (e.g., you can choose whenever color, icon, layout you want). However, you must include all the components previously cited and all required functionalities.