01UDFOV/01TXYOV – Web Applications I

React: State and Life Cycle

In this lab, you will continue to re-structure your web-based task manager to exploit the React framework and its component-based approach.

Exercise

Starting from the web application you developed in the previous lab, complete its functionality by properly using the functions offered by React, mainly in terms of *state* and *form* management.

In particular, you should add all the features pertaining to the user actions, namely:

- filter the task list according to the *filters* and *projects* available in the sidebar;
- create a new task (through the "+" button available in the bottom right corner and the relative form);
- edit an existing task, by re-using the form for creating a new task.

Plan how and where to set up the state. Create all the suitable callbacks and form event handlers. Consider the full life cycle of the application. We are still working without a REST server, so the task list should (still) be in a JavaScript array.

Beware: the form for creating/editing a task that you developed so far uses the Bootstrap's Modal. Bootstrap uses jQuery, under the hood, for handling its elements, as for the modal. Unfortunately, jQuery and React do not play well together. Thus, you can either:

- get rid of the modal, and adopt a solution similar to the one developed during the lectures;
- replace Bootstrap with its React-enabled version: https://react-bootstrap.github.io (same CSS, no jQuery, all the elements available as components).

Hints:

1. *You can use the specifications reported in Labs 3 and 4, if needed.*
2. *You can use the project developed during the lectures as an example or to get inspiration:* https://github.com/polito-WA1-2020/react-scores
3. *If needed, you may start from the previous lab's solution, available at:* https://github.com/polito-WA1-2020/lab7-react