01UDFOV/01TXYOV – WEB APPLICATIONS I

JAVASCRIPT IN THE BROWSER

During this third lab, you will add some "dynamic" behaviors, through JavaScript and the DOM manipulation, to the user interface realized in the previous lab.

EXERCISE 1 - POPULATE THE TASK LIST

Starting from the outcome of the previous lab, replace the "static" list of *all* tasks present in the "main content" (right area of the page) with a list generated from a JavaScript array. By changing the content of the JavaScript array, the displayed list of tasks will *automatically* update.

The JavaScript array will contain the tasks with all their properties (i.e., description, important, deadline, project name, private or shared). For now, consider the "private/shared" properties as a *flag*.
In addition, each task must have associated a *unique identifier* (i.e., an "id" property).

Similarly, the list in the webpage will maintain the same appearance and functionality as before (e.g., project names are in a colored label, important tasks have a red checkbox, etc.). For the purpose of this lab, all the labels for projects can share the same color.

The array must be contained in a *separate* .js file, properly included in the HTML file, thus not directly written in the HTML markup.

Hints:

1. *Continue to use Bootstrap 4.4 classes and components to style the webpage, if needed:* *https://getbootstrap.com*
2. *You can use the solution available for Lab 2 as a starting point, if you prefer:* *http://github.com/polito-WA1-2020/lab2-html-css*

EXERCISE 2 - FILTERS!

Make the "filters" work, by extending the previous exercise. In particular, you should enable the actions of the following *filters*, present in the left sidebar of the HTML page:

- **All**, to display *all* the tasks (as in Exercise 1);
- **Important**, to display tasks marked as *important*, only;
- **Today**, to show tasks whose deadline is today (e.g., *April 3, 2020*)
- **Next 7 Days**, to show tasks whose deadline is between tomorrow and the next 7 days (e.g., *Saturday 4th of April to Friday 10th of April*, inclusive)
- **Private**, to display tasks marked as *private*, only;
- **Shared with…**, to show tasks *shared* with somebody else, only.

All these actions will update the task list in the same “main content” area populated in the first exercise, i.e., without creating a new page or a new area. In other words, this will create different *views* of the same task list without modifying the content of the JavaScript array.

Add some suitable tasks to the JavaScript array created in Exercise 1 to test all the filters.

EXERCISE 3 – HIGHLIGHT EXPIRED TASKS

Starting from the outcome of the previous exercise, color in red the *deadline* text of any task in the “All” view in the moment they expire. Keep the red color across page reloads, once a task expired.

Hints: You can use setTimeout() in your .js file to manipulate the color of the text present in the deadline field of the “All” view.