Technical University of Moldova

Faculty of Computers, Informatics and Microelectronics

Department of Software Engineering and Automation

Web Programming

Laboratory work #3

LEARN JAVASCRIPT BASICS

Author:
Daniela Afteni
std. gr. FAF-203

Supervisor:
Alexei Șerșun

Chișinău 2023

1 Task

The task for this lab is:

1. Copy index.html, style.css and [app.js](app.js) to your repo;
2. Modify them to build an application for to-do list;
3. The app has to cover basic needs: - to add to the list; - to remove from the list; - to mark as done; - see ”done” and ”to-do” lists separately.
4. The app has to look attractive.

2 Results

In the laboratory work nr. 3 have been realised: the basic app, attractive app, the list is preserved after page refresh, the app implements a notification mechanism, 5 useful features the app implements besides the ones listed above.

To implement this laboratory work, I have done firstly the HTML file, then the CSS [1] for styling and the last JavaScript [2] file for function.

In the Figure 1 is represented how the app looks like when it has been run for the first time, which consists of an empty list name, input section for the new tasks, 3 options on how the tasks may be characterised, the add button, the list (which is in Figure **??** still empty, because there are no introduced tasks), and 2 main buttons for the representation of separate done tasks and to-do tasks.

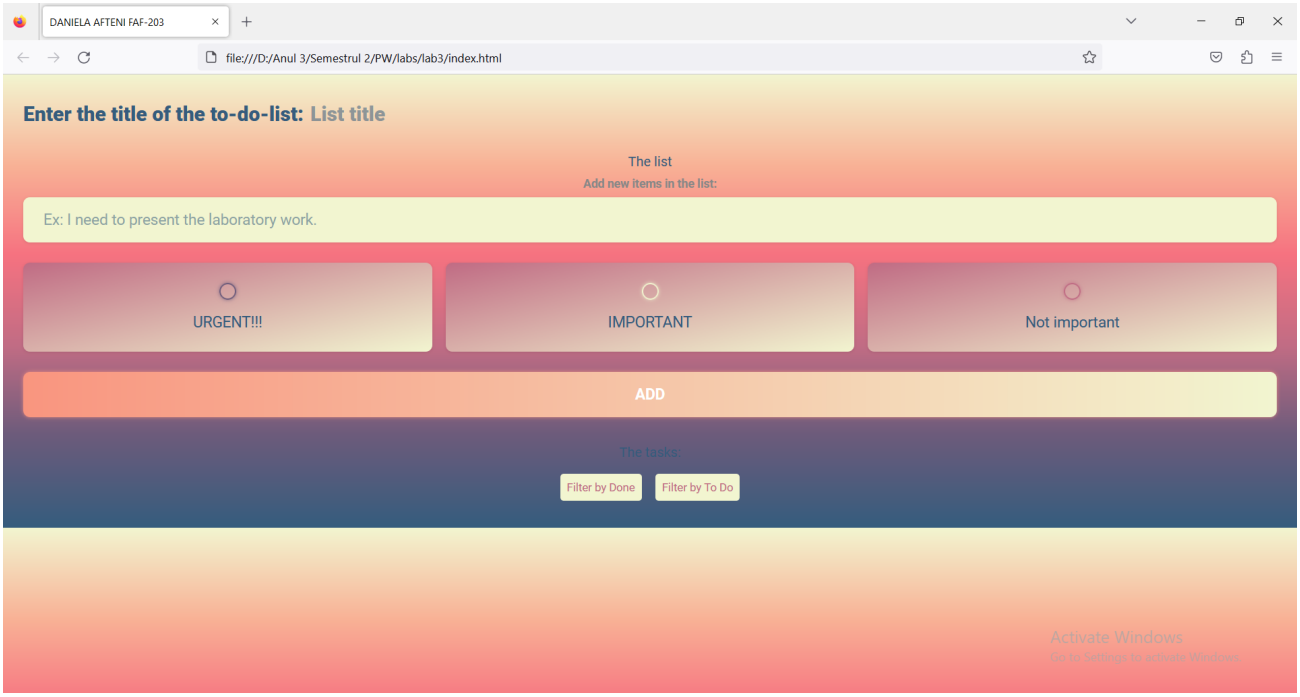

Figure 1: The initial list

The next is Figure 2, which determines the introduction of new tasks. As it can be seen, there was written the list name at the top of the page, as well as 3 introduced tasks, that are different regarding how urgent they are. Setting the importance of the task can be done before pushing the adding button. This is going to be set as a main characteristic of each task.

As well there can be seen another feature of the app, which is the representation of tasks. All the tasks are put in the top of the list (not at the bottom) depending on the date of the creation of the task. Thus, the task that was created the first, will be at the end of the list, but the most recently created is going to stay in the top of the list

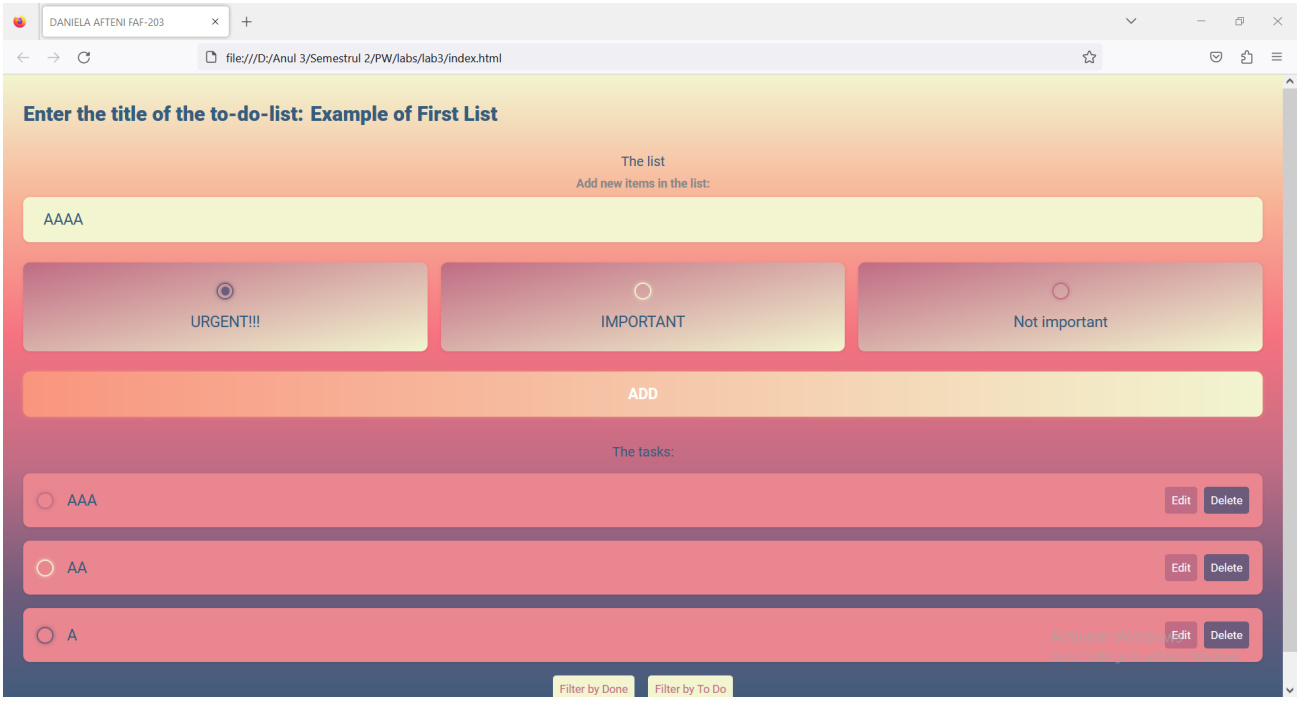

Figure 2: Adding new tasks

Regarding the setting of a task as a done task, which is shown in the Figure 3, it is generated by pushing the label tag, and represents that the task will be crossed and will have a done characteristic.

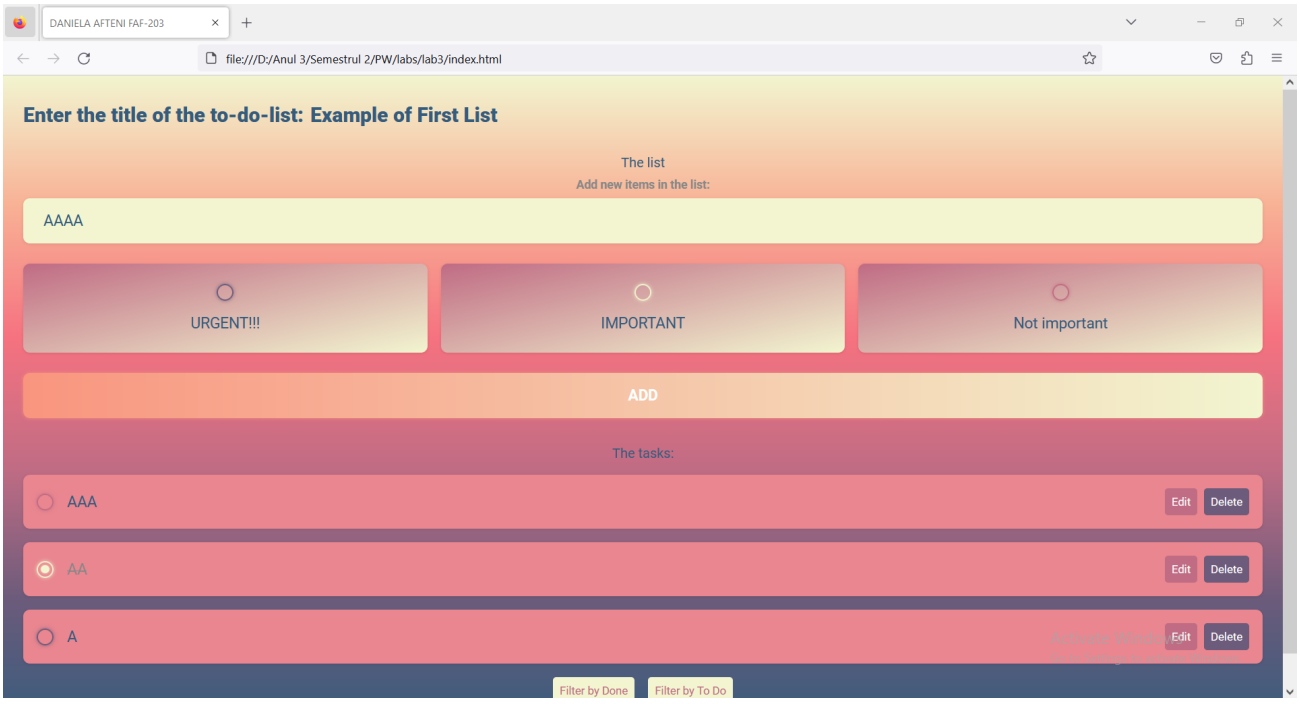

Figure 3: Setting the task as done

Another introduced feature is editing the task, after pushing the edit button on the corresponding task (Figure 4).

Deleting the task is represented in the Figure 5, which is done in the same way as the editing (by pushing the button), but the logic is another one.

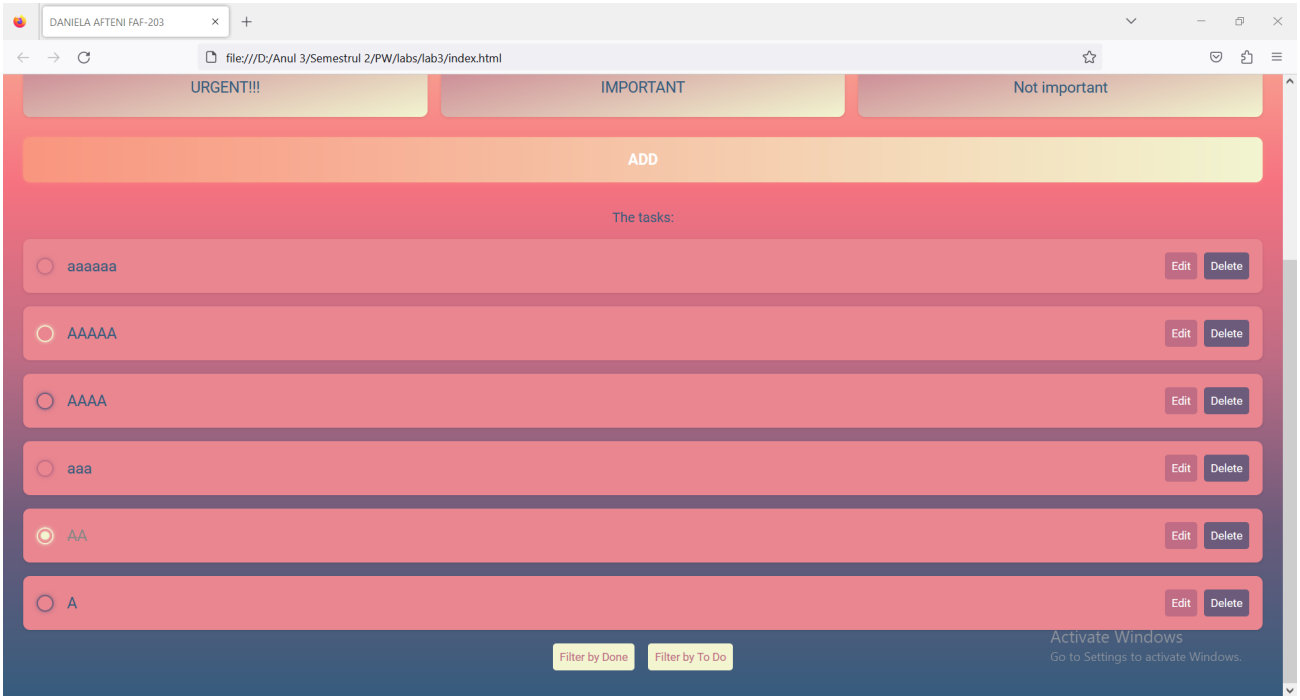

Figure 4: Editing the tasks

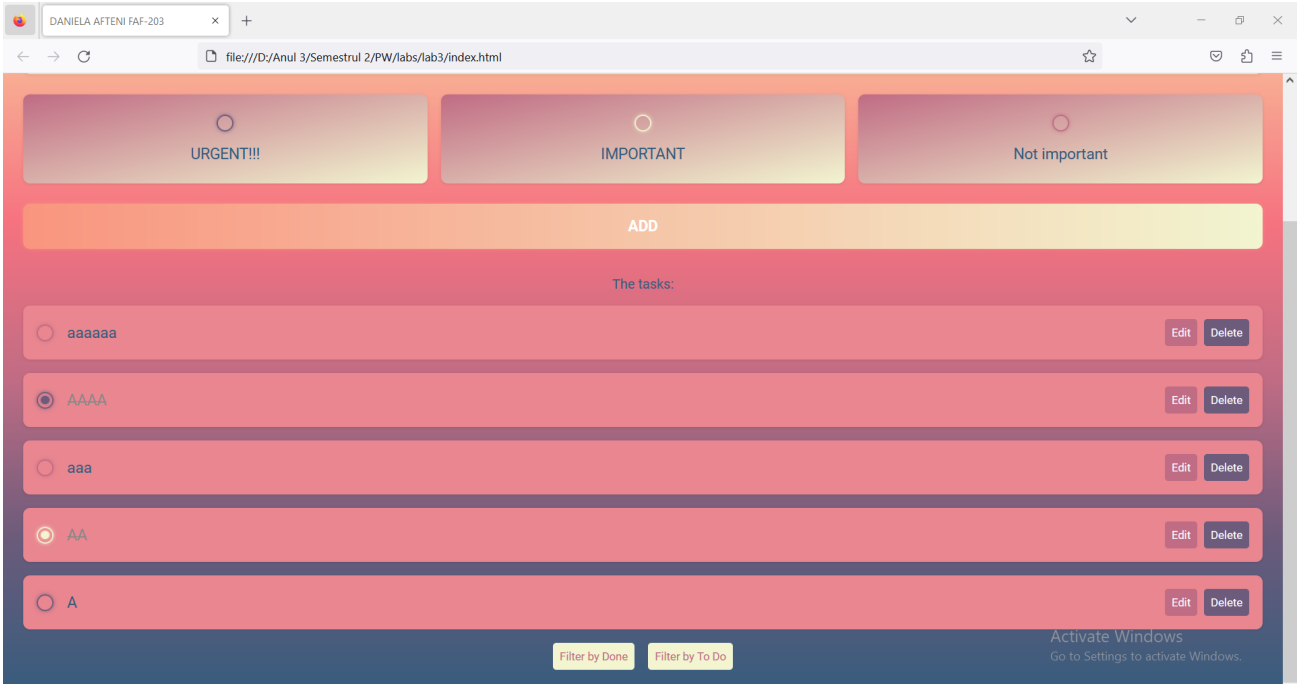

Figure 5: Deleting the task

In the Figure 6 and Figure 7 are shown the mechanism of the separation of done and to-do tasks.

Figure 6: Separation of done tasks

Figure 7: Separation of to-do tasks

In case somebody tries to add an empty task, then the mechanism of notification is going to start, by alerting the user that he needs to complete this section before submitting the new task in the final list (Figure 8).

In the last figure, Figure 9, is represented a part of the code.

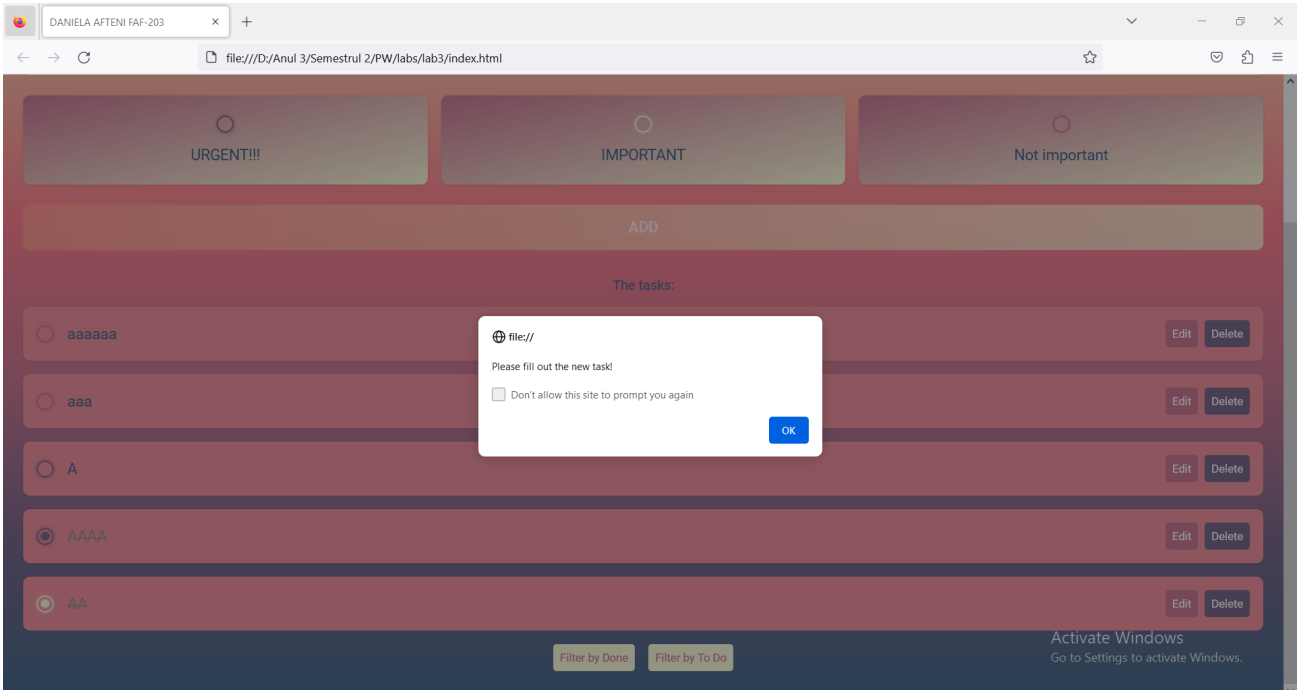

Figure 8: Part of the code

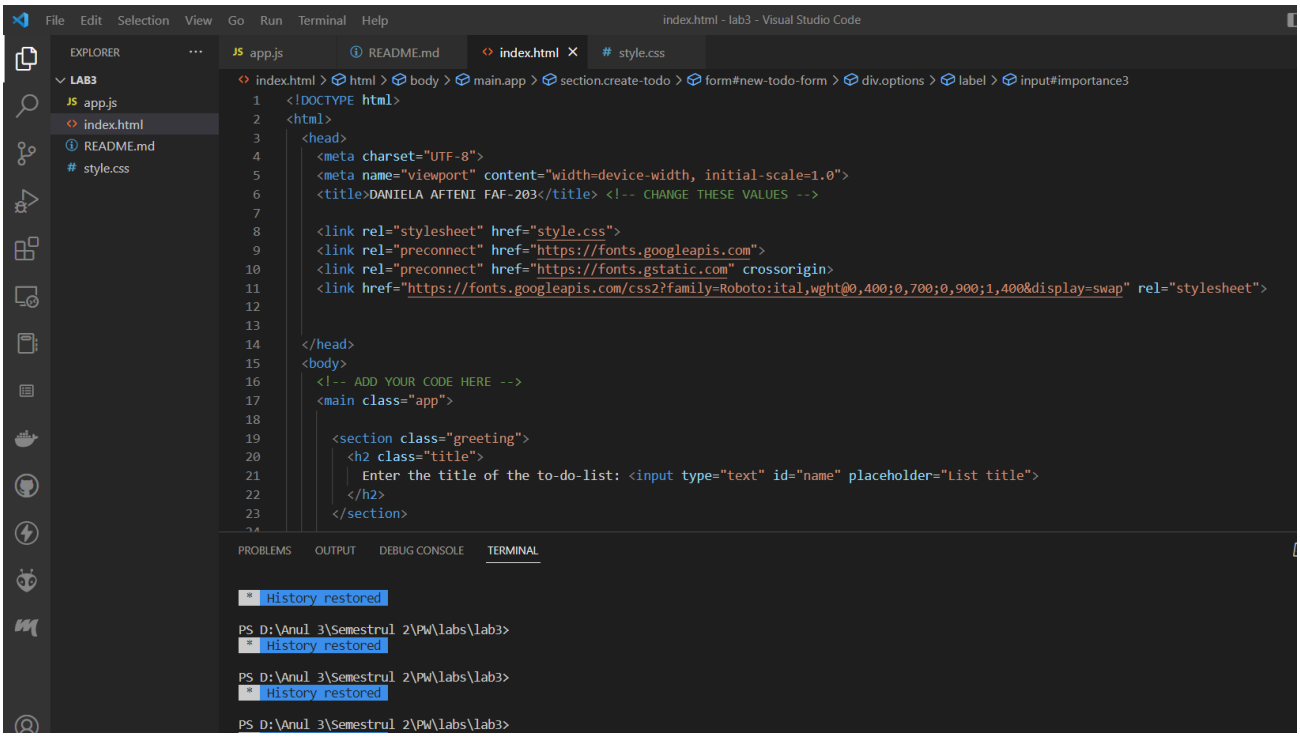

Figure 9: Part of the code

The realized main points:
1. to add to the list;
2. to remove from the list;
3. to mark as done;
4. see ”done” and ”to-do” lists separately.

The realized bonus points:
1. the app has to look attractive;
2. the list is preserved after page refresh.

The realized useful feature points:

1. naming the list by the user;
2. editing the tasks by clicking on the edit button on the corresponding task;
3. characterizing the tasks by their importance;
4. filtering the list by done or to-do tasks;
5. showing the tasks depending on the creation date of the them (the task that was created the first, will be at the end of the list, but the most recently created is going to stay in the top of the list).

3 Conclusion

Due to this laboratory work, I practiced working with HTML and CSS, that I learned as well in the laboratory work nr. 1, and learned JavaScript which was the functionality of the app. Depending on the skeleton that was build in the HTML, designed in the CSS file, there were added corresponding creation of new items, their filtering, sorting, setting, etc. I was also able to implement all the requirements regarding the list implementation. Thus, I learned about how to correctly structure the list, and its available functions.

References

[1] CSS Gradients, *https://www.w3schools.com/css/css3_gradients.asp*
Accessed on March 5, 2023.

[2] JavaScript, *https://developer.mozilla.org/en-US/docs/Web/JavaScript*
Accessed on March 1, 2023.