I started this project by creating a div with id, list, and hardcoding in a few to-do elements. At this time, I only had the span, and edit and input buttons. Then I decided to start working on the JavaScript code before attempting to add bootstrap. At this point, I decided to start with the things I learned doing JavaScript in the previous assignment so I added an event handler for a click on a delete button. Unfortunately, because I trying to work through the problem using JavaScript knowledge, I decided to use the click function instead of the on function. This would become a major problem that plagued me throughout this project.

At first the delete handler function worked and I decided to move on to the add. While I understand that starting with delete first may be the wrong order, I had already hardcoded a few elements so this seemed like a good idea. The add was relatively simple at first because I was getting the value from the text box and placing it inside a div with the append function. Unfortunately, I would have to come back to this while trying to create the edit handler.

At first with edit, I thought I would just have a second global text box that could edit any of the items. However, I did not know how to reference which element was being edited, so in the end, I decided to make each element have its own text box and save button. I then realized I would need to add these objects when creating a new element, so I went to my add handler and tried to add the new objects to the now very long string in the append function call.

It was at this point, that everything broke. I had been regularly testing things I added, but after working for a little while on editing, the remove button stopped working for elements that I didn’t hardcode. Then the edit button stopped working as well. In an attempt to fix this by looking over the classes in the add handler, I also broke the add function. Thus, after many frustrating hours, I went and googled examples of to-do lists. I discovered a new way of organizing the add function which helped make it more readable which fixed my problems with that. I also found the “on” function for jquery which actually managed to fix most of my problems. So now instead of referencing a class specifically, I was referencing my list item and finding the class inside it.

During this time of frustration, much of the functionality of the project changed. I had been experimenting with checkboxes, but eventually just went with clicking the class and making it slash through. It is interesting to consider what would have been different if I didn’t change so much trying to fix a lot of problems. After spending so much time on other problems, I decided to go with the simplest, if inefficient, method for local storage by just storing the entire div.