

PUI Assignment 3

Implementing Interactive Prototype with HTML/CSS/JAVASCRIPT

Hannah Rosen

Challenges and Bugs

The first big challenge I faced was how to bring users to a second screen for adding tasks, as I had laid out in my previous iterations. I really wanted to carry over this functionality from my prototype, because I wanted to continue allowing users to select the family member responsible for a given chore, and doing that and adding the text of task seemed like way too much content and functionality for a pop-up or modal. The problem was that I wouldn't be able to use javascript to save inputs between multiple html documents, so after exploring some options, I realized I could write both screens into the same html document, and use the jQuery functions `hide()` and `show()` to switch between the two. I also used the `$(document).ready` to hide the second screen on page load.

A difficult challenge turned out to be how to dynamically display the image of the person assigned to a new task in the list view. I was able to use `console.log()` to ensure that I was capturing the person's name in the variable "familymember" that I had defined to hold it, but I was really struggling to figure out how to pass that through to the new task. I thought about trying to use a loop to check which person it was and assign the image that way, but that seemed overly complicated and I was hoping to find a simpler solution. I started by changing some of the formats of the images so they were all jpgs and I made sure they all followed the same naming convention of name.jpg. Next I needed to use the variable familymember to insert the person's name into the source link in the image tag that I would be adding to the DOM using the `.append()` function. This process ended up being quite buggy, and took a while to get right. The key issues was figuring out the right the locations and types of quotations to use. I had to plan where to use single quotations and double quotations so that I wouldn't be accidentally closing strings when I didn't intend to. I also had to always double check that Brackets, the text editor I was using, wasn't inserting extra quotations where they weren't needed. It tended to do that. This part of the code was especially difficult to debug, because it was mostly in quotations so it was not appearing with the usual colour coded markers that Brackets uses to help read and understand the code. See the relevant code snippet below, with the variable in red.

```
$("#family_list").append('<li class="list-group-item">' + '<span class="label"></span>' + taskentered + '<span  
class="glyphicon glyphicon-remove float_right"></span><span class="glyphicon  
glyphicon-ok float_right"></span></li>');
```

Another challenge I faced, was that newly added tasks were not reacting to my complete a task and delete a task functions (activated by clicking the checkmark and "x" icons.). Only the pre-populated tasks were behaving as I expected. I knew that some of my classmates had experienced the same issue, so I reached out to them to talk it through. What I learned from those discussion was that since that javascript document loads all at once, new elements created after the initial loading would not be aware of the functions that had

already loaded. I would need to repeat those functions in the append function I was using to create the newly added tasks so they would be aware of them and respond to them later when clicked. In order to do this easily, I put both the complete and delete functions into a single function to easily call those functions again without repeating the whole thing.

External tools and resources

Bootstrap

I used bootstrap to help make the site mobile, and to use some of their css to decrease the amount of css I had to write myself. When planning out my html elements, I used the bootstrap documentation to find examples that matched what I was intending to create, like the list classes I used to create the body of the list.

Stack Overflow and W3 School

I used both stack overflow and the w3 school resources when I was looking for help achieving a certain function or trying to find a particular function or syntax I needed. I would google the problem, and go to these resources until I had found what I was looking for or solved the problem. I found the w3 school particularly helpful for looking up specific html tags or jQuery functions and brief descriptions of how to use them, and stack overflow particularly useful when I was searching for a functionality and needed some ideas for implementing it.

Some examples of the types of search terms I was using:

- Hide in jQuery
- Add a class in jQuery
- "this" in jQuery
- Types of html inputs
- Get a text box value in jQuery

Extra Credit Features

- Tasks assigned to family members, and ability to assign new tasks to family members by selecting their image
- Tasks fadeout when deleted
- Hover states on checkmark and x icons for completing and deleting tasks.

Link to github repository: <https://github.com/bananarosen/PUI2016>