

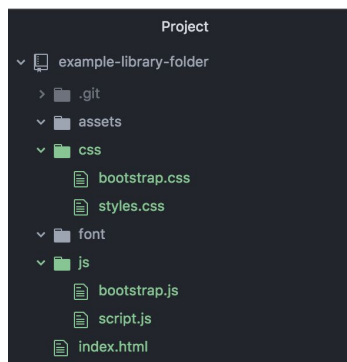
Capstone Project - Phase 2 of 6 (UI)

Getting Started

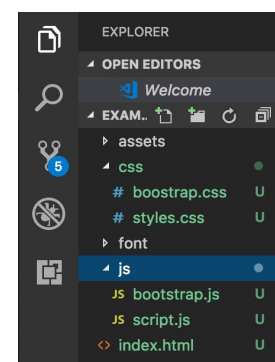
Recall that you already have both a local and a remote repository for your Capstone Library project. In Phase 1 of the Capstone Project, you built the engine that will drive your application using JavaScript as a feature off of your develop branch and within a feature folder. **Before you begin Phase 2, please merge your final copy of Phase 1 into your develop branch if you have not already done so.** As we enter Phase 2, we will create a new feature branch within our feature folder, titled **phase-two**, and step back into the worlds of HTML, CSS, and Bootstrap to create and bring definition to the user-side interface for your application. At this point, you have seen an example of what this could look like, but it is also available below for further reference.

Set Up

Your file structure at this point should look like this: index.html at the root, a css folder with bootstrap.css and styles.css, a js folder with bootstrap.js and script.js, an assets folder for later images, and a font folder with icons/fonts for later, etc.



or

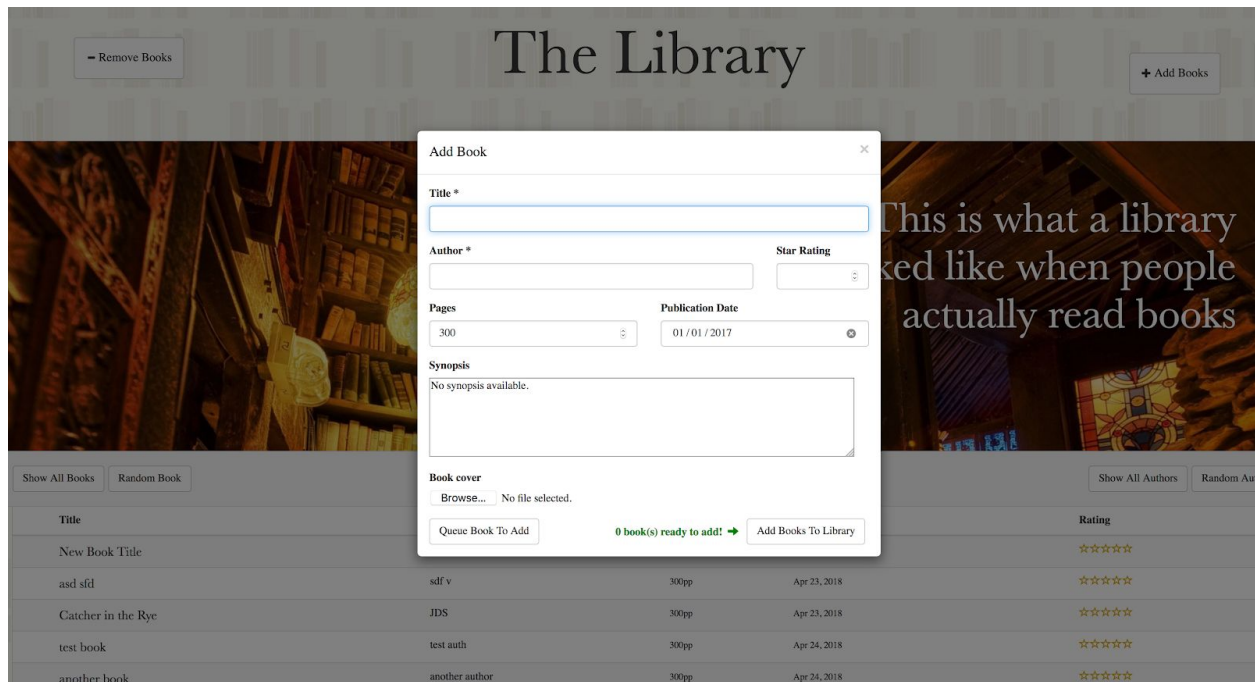


- Must use: bootstrap.css and bootstrap.js
- Cannot use: jQuery

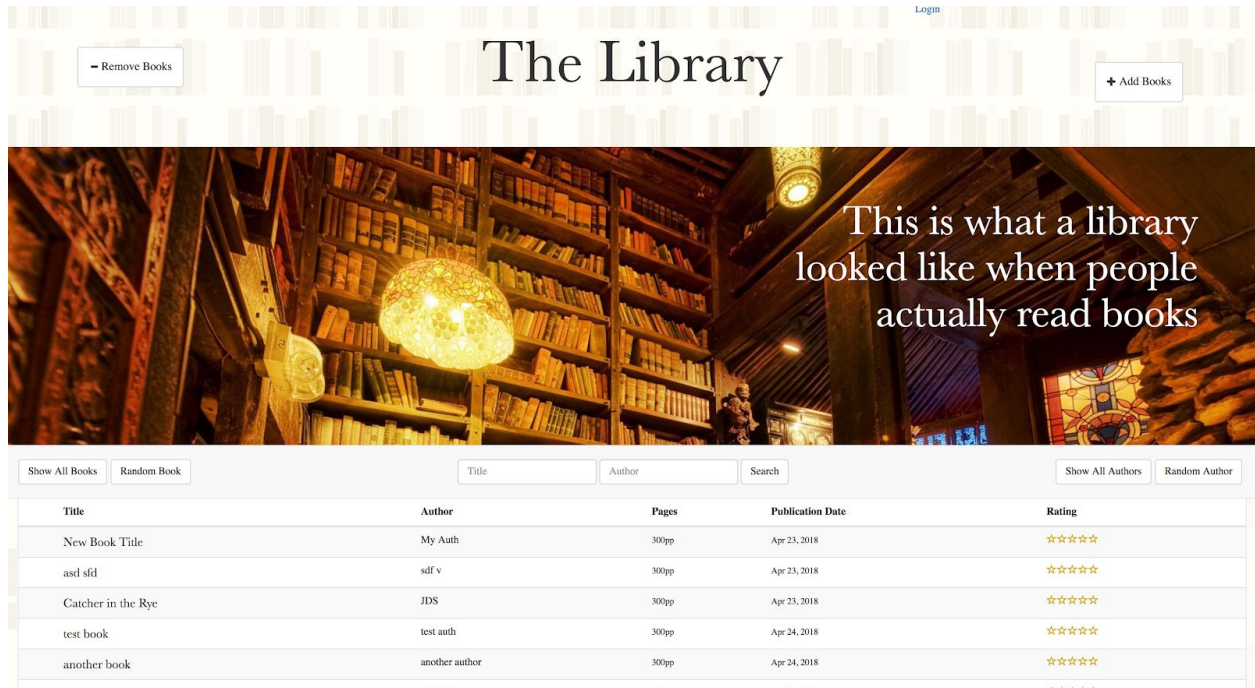
Requirements

- Use a Bootstrap grid layout to design the page
 - <https://getbootstrap.com/docs/3.3/examples/grid/>
- Include a module that will allow users to add book objects
 - Have at least 5 input fields that correspond to the book object properties (cover as a base64-encoded string, title, author, number of pages, publish date, delete)

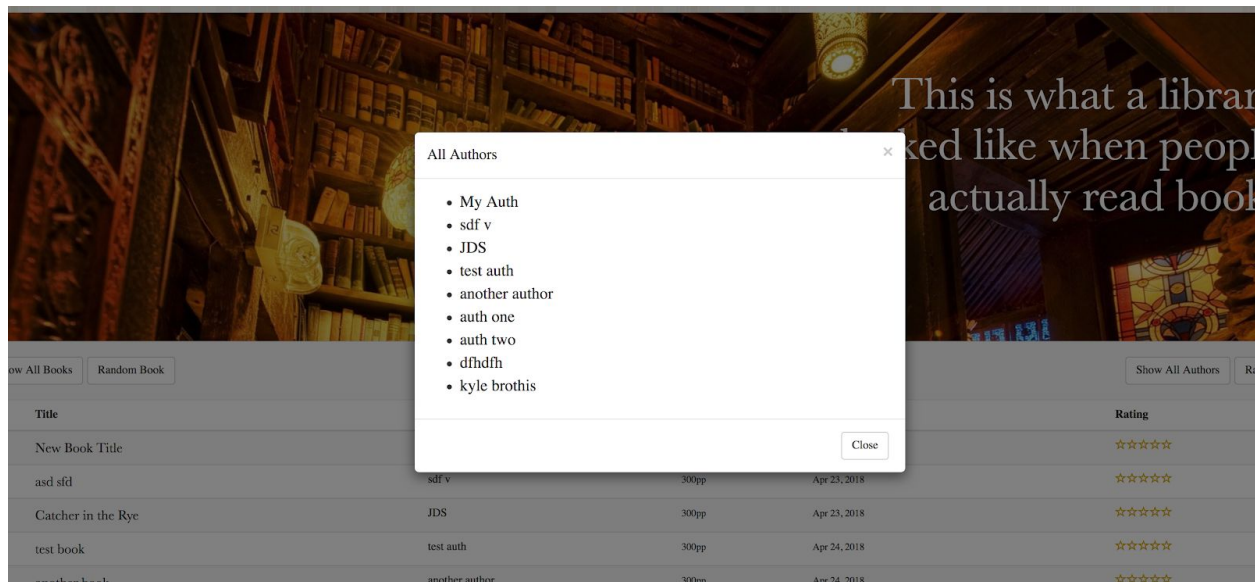
- For further information on base64 encoding:
 - <https://blogs.oracle.com/rammenon/base64-explained>
 - <https://www.base64-image.de/>
- Have a display area for a new form, or form-type element, to add as many new book objects as the user would like at once. This will end up being dynamic, meaning that it should show the option to add more than one book at a time, by displaying a new form or form-type element.



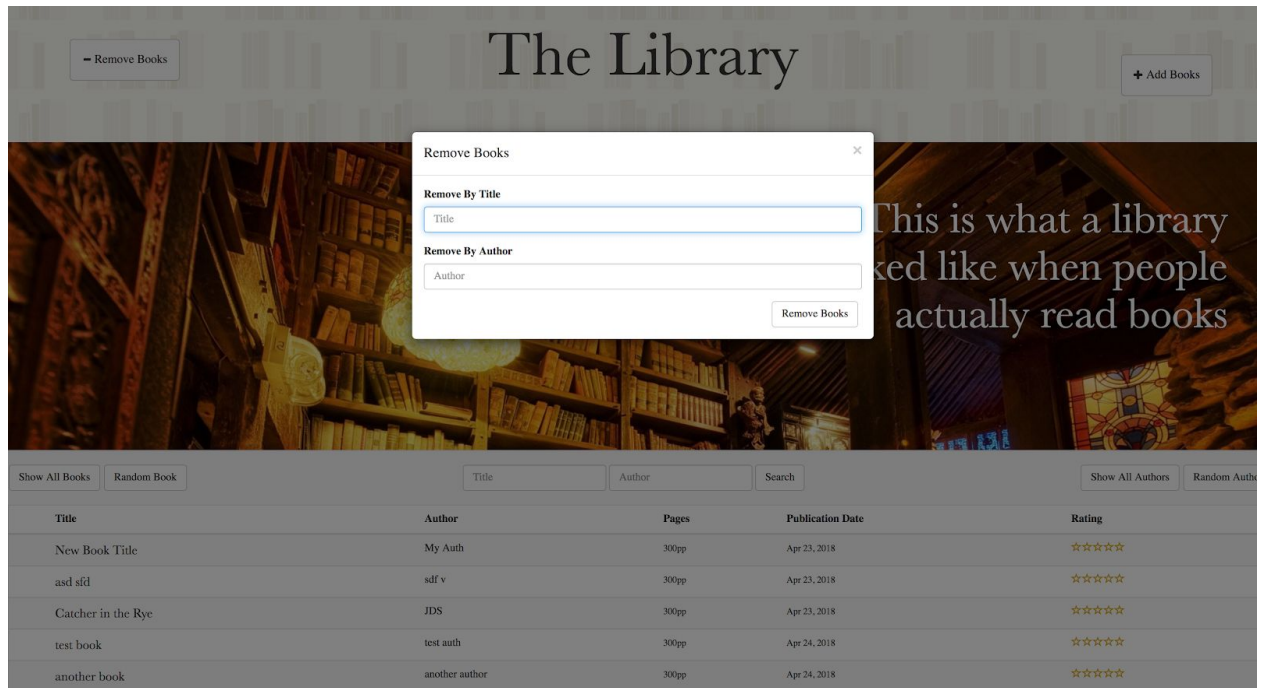
- Include a table that displays each book's information
 - Include a header with labels for each column
 - Each row will have information for a single book object
 - A single column will be present for each of your book object's properties (cover, title, author, number of pages, publish date, and delete)
 - A delete (x) button in the rightmost column of the table
 - **Bonus items:**
 - The ability to edit a book's title
 - The ability to edit a book's author
 - You may need an edit button to accomplish this



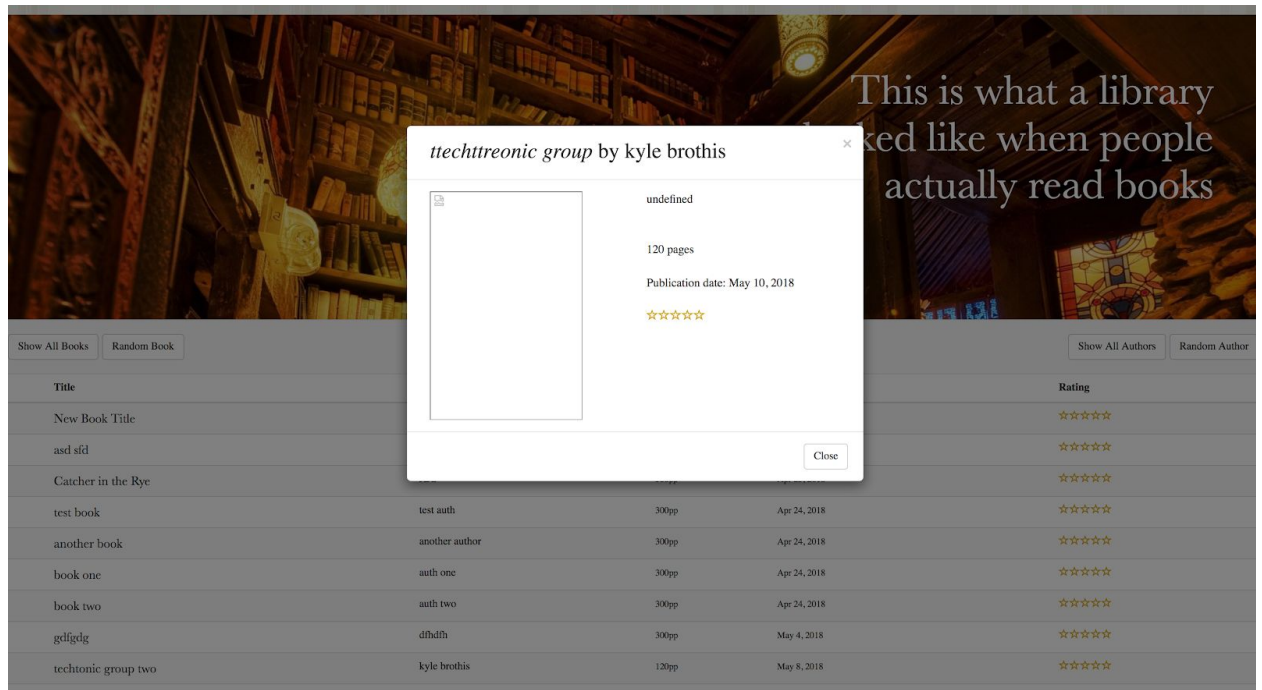
- Include a display area to show all of the authors available within the library



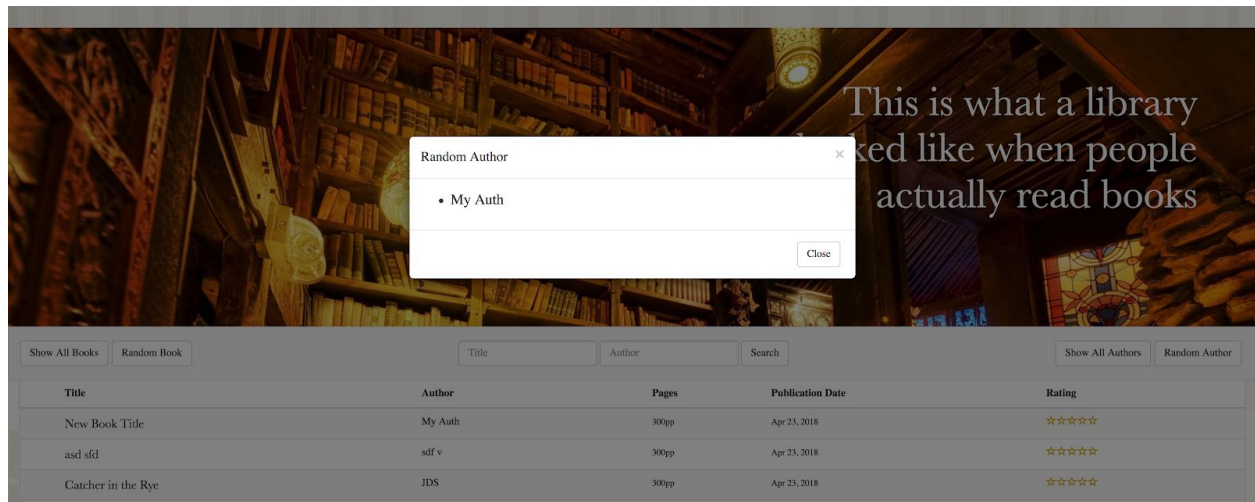
- Include a module with an input field/button that will allow you to remove books by a single, unique author



- Include buttons to:
 - Return a random book object. Think of this as a book suggestion.



- Return a random author from your library with a display area for the result



- Include at least one input field and button for searching through your library object's contents for a specific title or author match
*See table example above
- Should be mobile-responsive and have the appropriate metadata information included in the head
- Should have scripts that run just before the </body> tag to match industry standards

What to Do

1. Review the requirements
2. Devise a list of clarifying questions you might ask a client (maybe about how they want something to work, what they want it to do, etc.) and ask them
3. Plan your UI layout/design
4. Share your plan with someone else and get feedback
5. Start writing your code
6. Commit and push every time a new module is completed and functional

Helpful Materials

- <https://developer.mozilla.org/en-US/docs/Web/Reference>
- <https://www.w3schools.com/bootstrap/default.asp>
- <https://uxplanet.org/ui-ux-design-glossary-navigation-elements-b552130711c8>
- <https://uxdesign.cc/9-essential-ui-design-trends-you-should-stick-to-in-2018-26f33d1fe980>
- <https://designshack.net/articles/layouts/minimalist-design-is-taking-over-heres-why/>

Testing

- Compare your code to the rubric and clean it up for code review

Submitting

- Add Kyle and Shambre as collaborators (kbrothis and ShambreSW) on GitHub
- Project due at **5:00 pm on Thursday, July 5th, 2018**
 - If you do not think you can finish by this time, Kyle and I must receive a Slack message from you saying so by **1:00 pm** on Thursday, July 5th, 2018 (it's okay to tell us you won't finish and then discover that you are able to, but we want to know if this pace is too strenuous and adjust if necessary)