

Nolan Meeks and John Hunt

<https://swe432-textbook-app.herokuapp.com>

1. User friendly notifications

Some of our current notifications to the user use terminology that a novice user may not understand or care about. Such as information about the server and data being posted. This violates heuristic number 2: Match between system and the real world. We will fix this by adjusting the messages so that only important and relevant information is presented to the user.



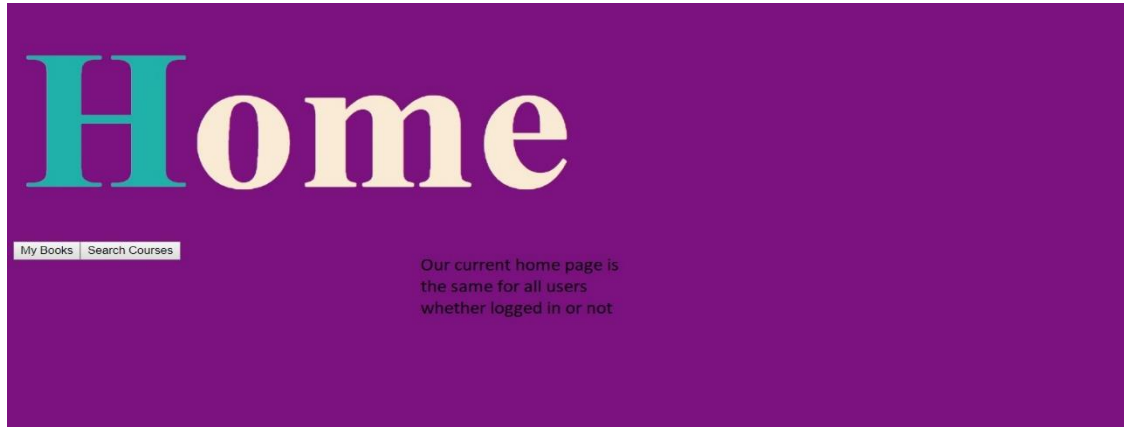
2. Shortcuts

Currently our app has no shortcuts for frequently used actions. This is a violation of heuristic number 7: Flexibility and Efficiency of Use. By adding shortcut for some common actions, we can improve the user experience by increasing the speed of their actions while on our site. Shortcuts such as clearing form data could be useful to clear all fields at once instead of one at a time.



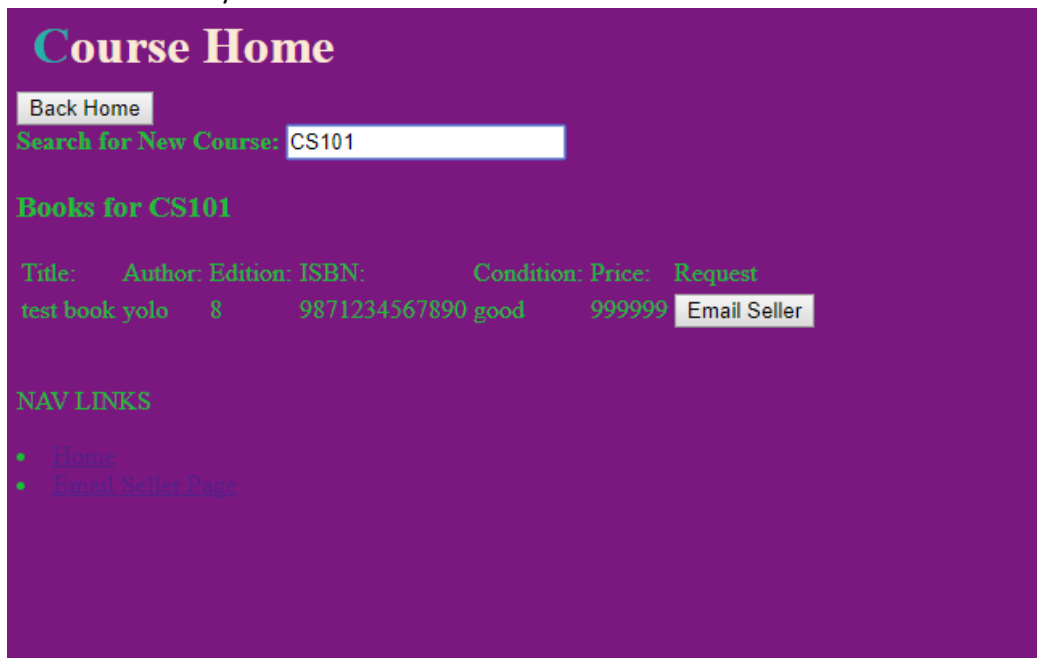
3. Adjusted layout based on user

Currently our app displays the same information no matter whether the user is logged in or not. This would violate heuristic number 3, User Control and Freedom. We could improve the user experience by providing shortcuts on the home menu based on the user's profile. This would allow them to do some common function without having to navigate to the page for that action.



4. Searching Status Displayed

Currently when you search for a course to see if there are any book posts for that course, once you hit enter, there is nothing telling you what the application is doing, or if it is even doing anything. This violates heuristic number 1, visibility of system status. To fix this, when a user begins a search for a course, we can add a loading alert box to let the user know a search has started and when they should see their results by.



5. Standardize App Layout

Right now the titles on each page vary in size and each page's buttons are relatively scattered in different places on the screen. This violates heuristic number 4, consistency and standards. To fix this, we can rework the styling on our app and re-arrange components to keep a more fluid design throughout the user's experience. It is apparent this issue exists by comparing these screenshots to each other and noticing all the differences in how things are layed out.



6. Disallow bad types of input

In our app, currently when trying to add a new book post, you can type whatever you want inside the fields, leaving room for error if users were to enter words for a price, or an improper course title. This violates heuristic number 5, error prevention. To get around this, we can make a regular expression for course titles to check to make sure it follows the proper format, as well as setting types on the input fields to they are correctly filled out. You'll see below most fields allow any input, which we need to limit.

Add Book

Know the ISBN? (Dont use this right now lol) ex. '1234567890' Search

New Book:

*Isbn: Words Can Fit Here

*Title: Anything

*Author: Anything

*Edition: Anything

Publisher: Anything

Copyright: 999999999999

Anything

Comments:

*Course: Anything

*Teacher: Anything

*Condition: Anything

*Price: \$999999999999999

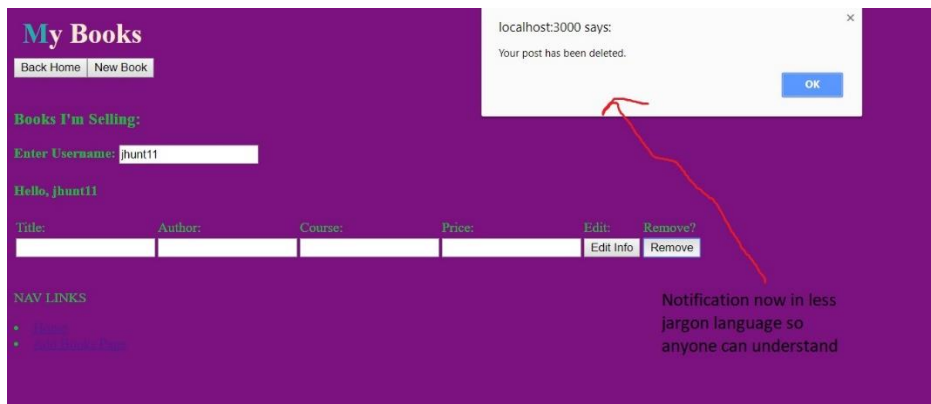
Post Book

NAV LINKS

- [Home](#)
- [My Books Page](#)

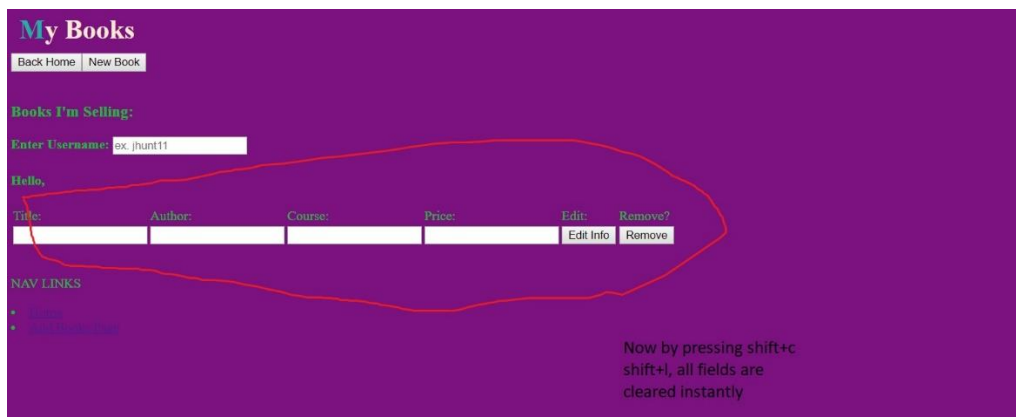
1. User friendly notifications

All notification text was revised and modified so that anything too technical was replaced. The goal was to provide accurate information in as simple a way as possible so that all users will be aware of the status of their actions. This fixes heuristic number 2 because now our app is using real world terms to define the system actions going on behind the scenes.



2. Shortcuts

I added a new library called Mousetrap to create keyboard shortcuts for our web app. I felt the ability to quickly clear data would be useful in case the user inputs the wrong data and wants to start over. This helps fix heuristic number 7 by allowing the user greater flexibility in how to edit data in the fields. This also gives us a chance to create more shortcuts that we find will be useful to users.



3. Searching Status Displayed

Added a new spinner component to help display a searching/loading message when the user is attempting to search for a course. This helps because it will allow the user to know that their search has successfully started, and if they aren't seeing data still then something is wrong.

Course Home

[Back Home](#)

Search for New Course: [Search](#)

Searching...

Books for CS101

Title:	Author:	Edition:	ISBN:	Condition:	Price:	Request
test book yolo		8	9871234567890	good	999999	Email Seller

NAV LINKS

- [Home](#)
- [Email Seller Page](#)

4. Disallow bad types of Input

I went through the add book page and in addition to the required attribute already given to many of the fields, I created several regular expressions for the fields so they would have a proper pattern to look for to help reduce the possibility of a user typing in incorrect data. This resolves the error prevention issue by not even giving users the chance to input the wrong data.

Add Book

Know the ISBN? (Dont use this right now lol) [Search](#)

New Book:

*Isbn:

*Title:

*Author:

*Price:

!

Please match the requested format.

Post Book

Comments:

*Course:

*Teacher:

*Condition:

*Price: \$

Post Book

NAV LINKS

- [Home](#)
- [My Books Page](#)