CS-571 Building User Interfaces, Fall 2021, Professor Yuhang Zhao

React 2 *β* (3 Bonus Points, Optional Extra Credit Assignment)

Improving Usability Using Heuristic Evaluation

In this assignment, you will put the ten usability heuristics we learned in class into practice toward improving the usability of your *React 2* ***α*** deliverable. You will focus on specific components of your design, identify potential violations of the heuristics, make design recommendations to address these violations, and implement recommendations that are feasible to create a new deliverable. Use this opportunity to make concrete design decisions about your project, to improve your design using the heuristics, and to build a keen eye for identifying usability issues as a UX developer.

Step 1—Identify A Focus. (0.2 Points) Review your *React 2* ***α*** deliverable with a critical eye to identify 3–5 "components" that you think are most consequential for user experience.

Step 2—Review the Heuristics. Review the ten usability heuristics we discussed in class from the slides, what principle each heuristic represents, and examples of the violations of the heuristics.

Step 3—Identify Potential Violations. (1.0 Points) Focusing on your components, inspect your design, considering each usability heuristic, for any violations of the heuristics.

Step 4—Develop Design Recommendations. (0.4 Points) For each violation you identified in the previous step, provide a design recommendation for addressing it, assessing its feasibility.

Step 5—Implement Your Recommendations. (1.4 Points) Implement the design recommendations that you identified as "feasible" in the previous step in your prototype, updating your design.

Submission Details

GitHub Classroom Starter Code

React 2 β will build on your implementation of React 2 α. You should copy your code from your React 2 α project to the React 2 β repository linked above, as that will be your starter code. When you commit and push, ensure that you are committing and pushing to the react2-beta repository, not react2-alpha.

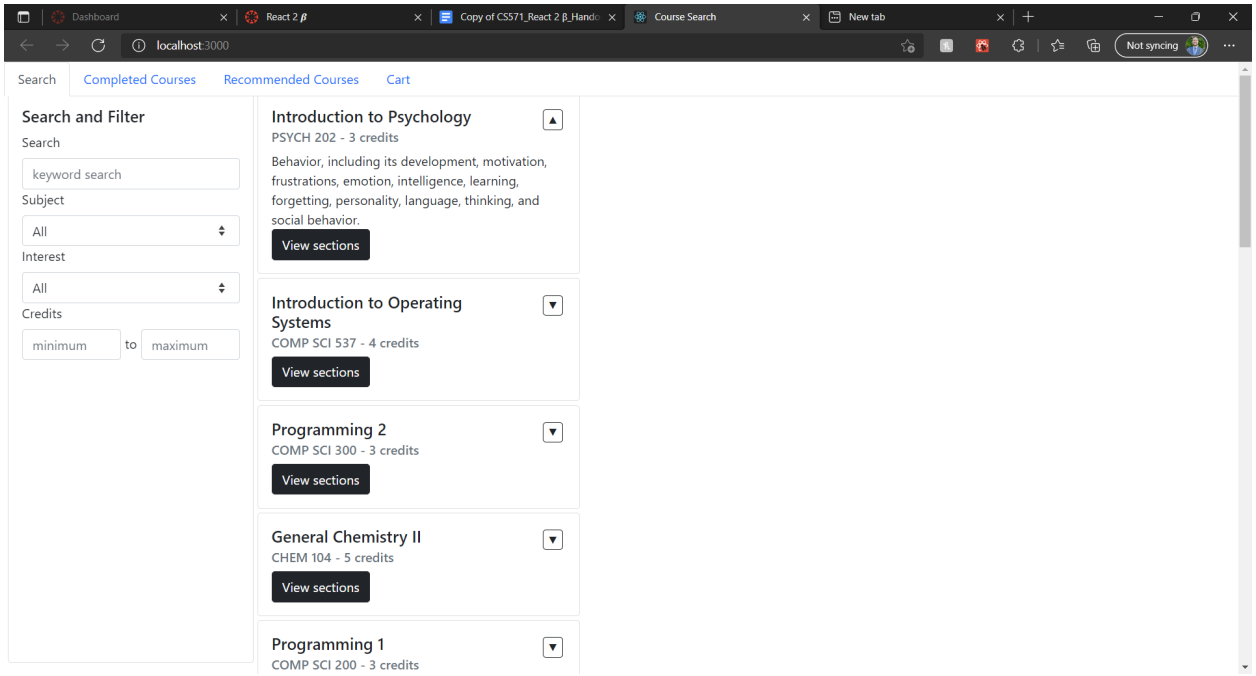
NOTE: *If you were unable to complete one or more problems in React 2* ***α*** *Assignment and your code is not fully functional, you can still complete this assignment and get full credit. Remember, in* ***β*** *assignments, you will be judged on the design aspects of your application. For example, if you were unable to develop a recommender algorithm in React 2* ***α****, you can hard-code in a few dummy courses in the Completed Courses tab and apply the design to them.*

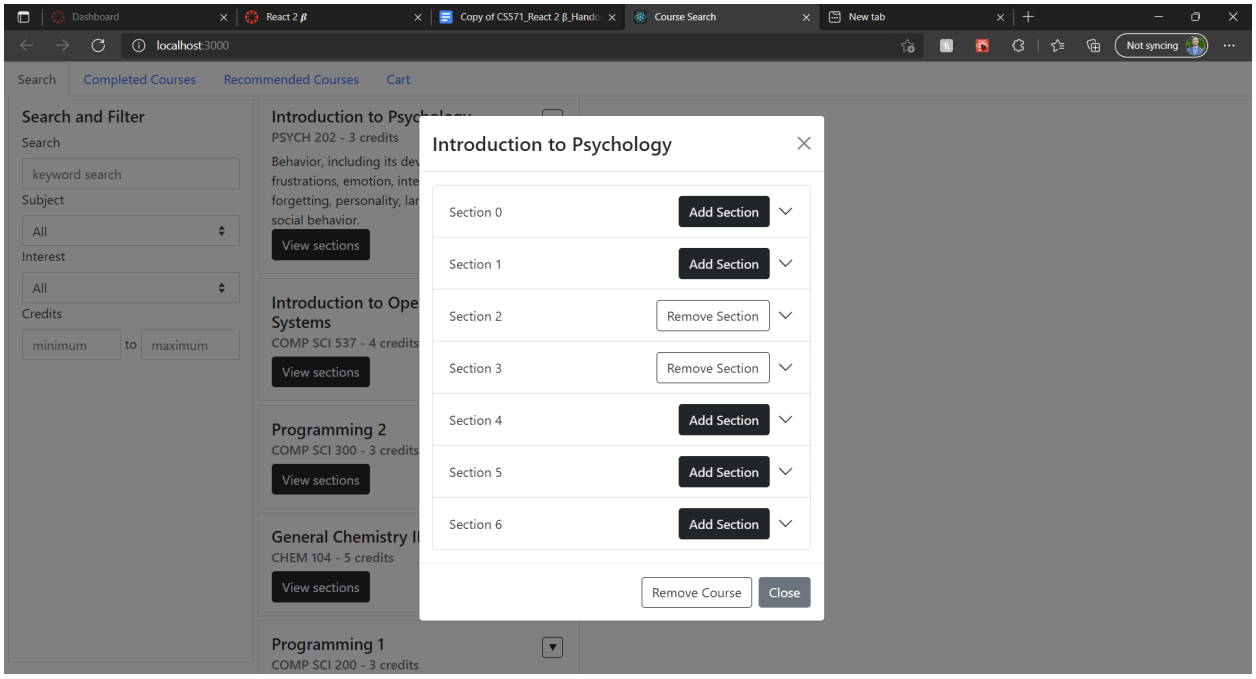
To complete the assignment, you will need to submit the following on Canvas:

1. A completed version of this document as PDF as an attachment.
2. Your repository name and latest commit hash from GitHub Classroom, e.g. `react2-beta-osori, 91ddcb6`, as a comment.

Step 1. Identify A Focus. (0.2 Points)

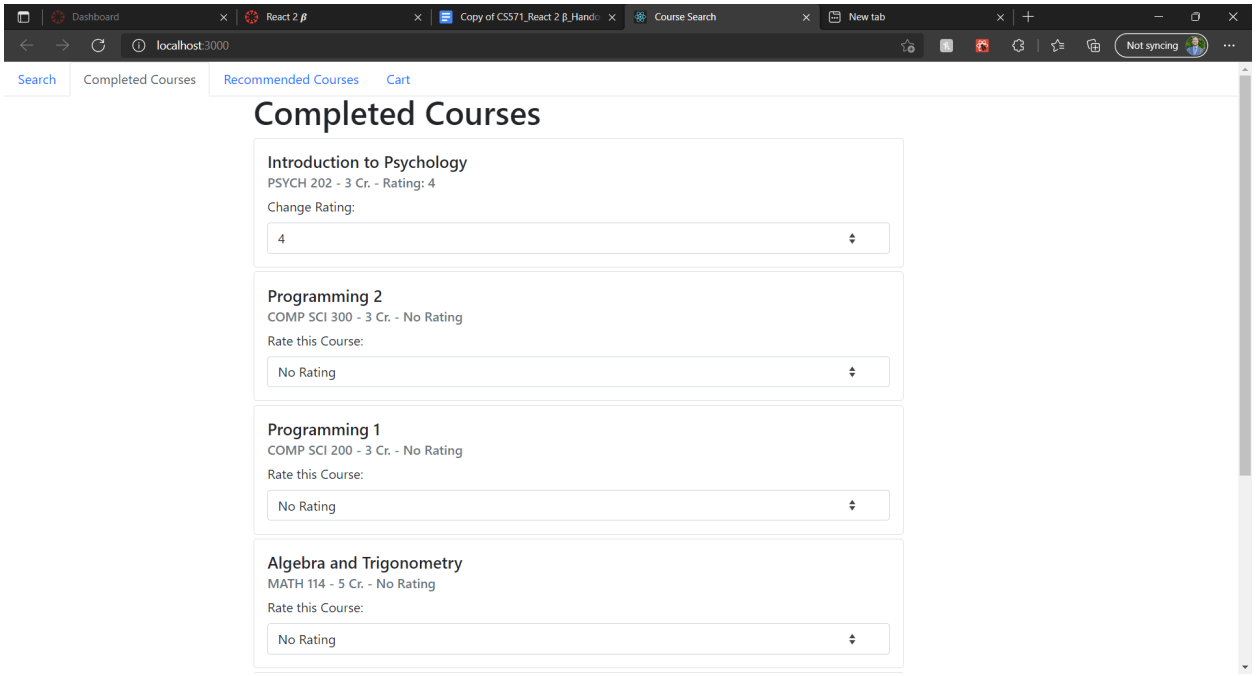
In this step, you will review your *React 2* ***α*** deliverable with a critical eye to identify 3–5 "components" that you think are most consequential for user experience and that you will put under the microscope of heuristic evaluation in the next step. In real life, your application might have hundreds of components, screens, or pages, and you will have to focus your efforts on a limited set that will make the most difference in terms of effectiveness and user experience. Similarly, you will review your design and identify 3–5 components to focus on. Here, a "component" can be the entire page/view (e.g., recommended courses) or a reusable component (e.g., the course component, the rating component), but not something as small as a button or label. Provide screenshots of each component below and provide a brief justification (1–2 sentences) of why you think each one is a critical component.



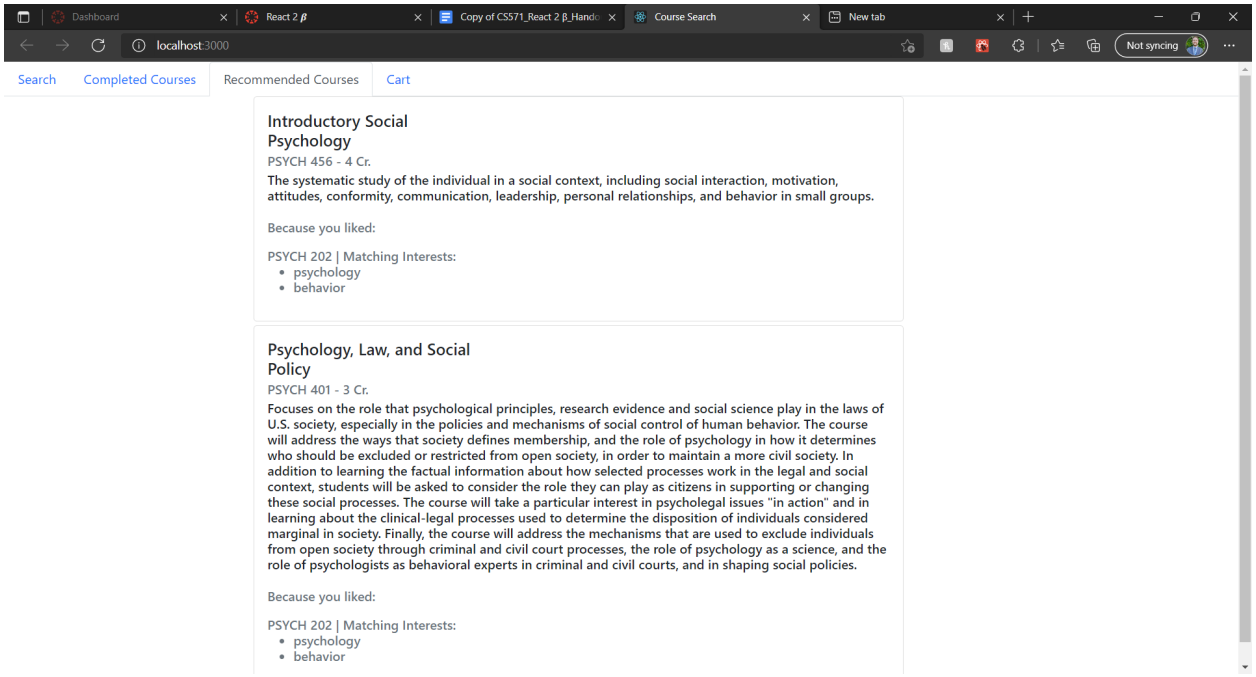


Search View / Cart View / <Course>:

These pages which utilize the <Course> component offer the primary functionality of our app. It's worth analyzing them against our heuristics.



Completed Courses View / <CompletedCourse>:
Given that Completed Courses is one of four views in our app, and that it's a necessary step in generating recommended courses, this view is worth analyzing.



Recommended Courses View / <RecommendedCourse>
Recommended Courses acts as the secondary functionality of the app, where the primary functionality is adding/removing courses. As such a big part of the overall application, the Recommended Courses view is worth analyzing.

Step 2. Review the Heuristics.

Carefully review the ten usability heuristics we discussed in class from the slides, what principle each heuristic represents, and examples of the designs that violate and support the heuristics. Below is a cheat sheet for Nielsen's ten heuristics that you can use in the next step. (This step does not have any deliverables.)

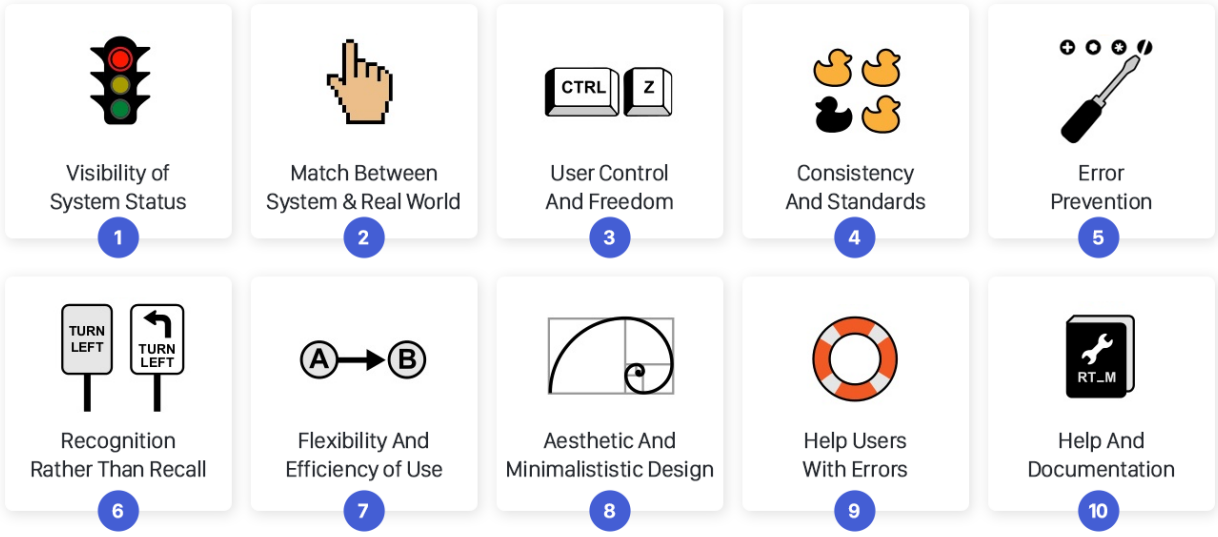
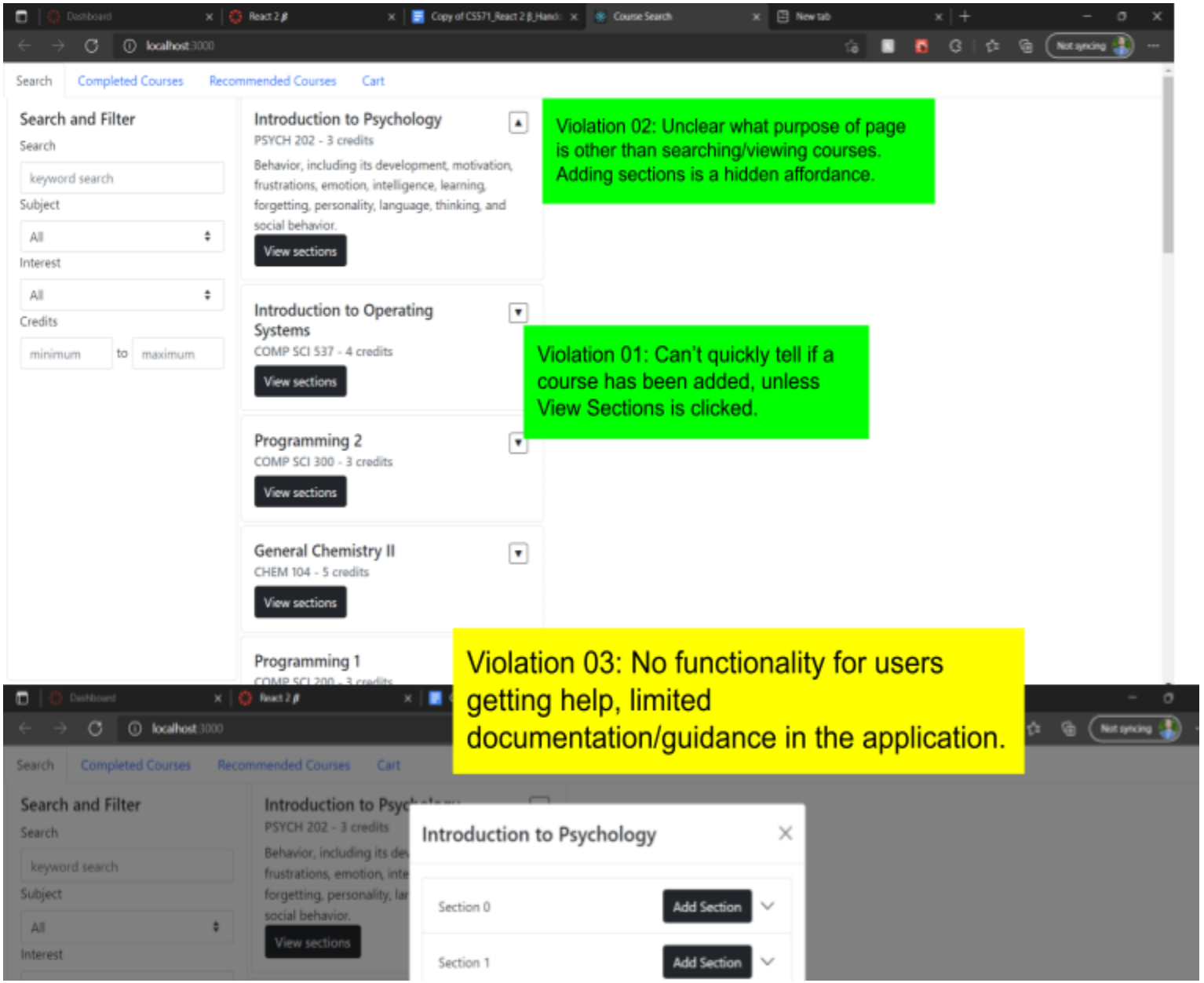
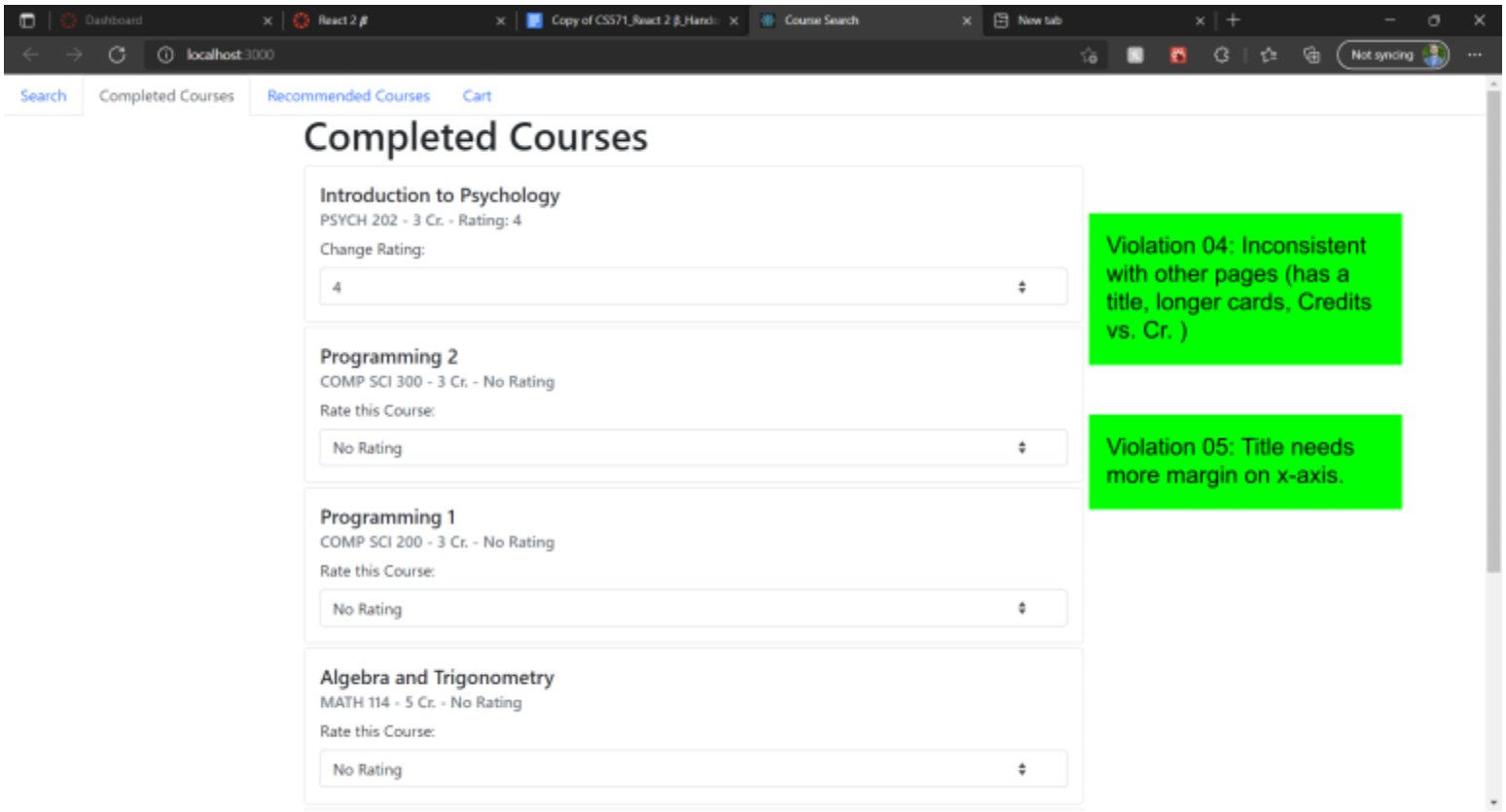


Image source: UX Collective

Step 3. Identify Potential Violations. (1.0 Points)

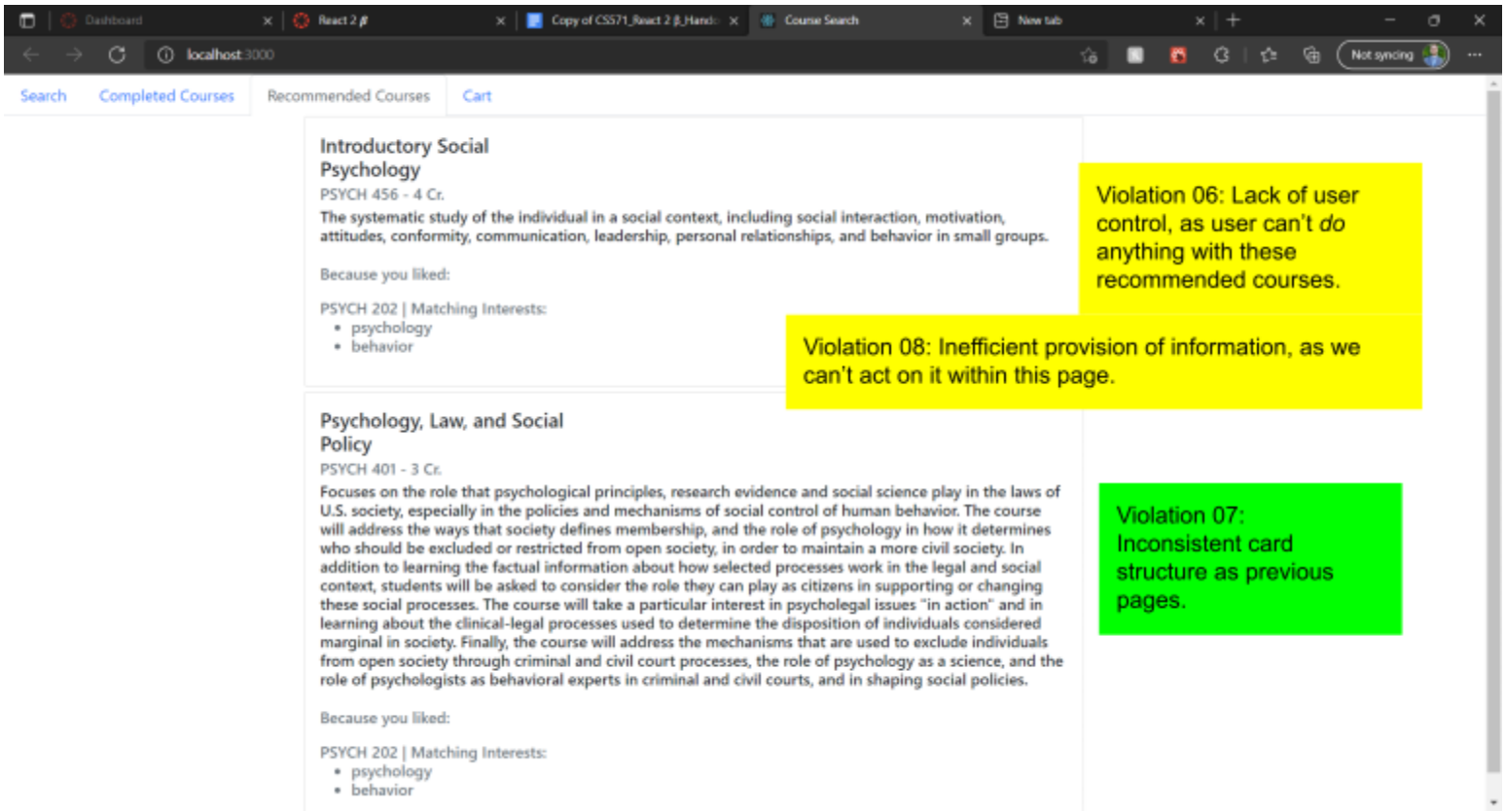
Focusing on your components, inspect your design, considering each usability heuristic, for any violations of the heuristics. For each violation, use the following table to briefly describe the violation and give it a unique number (specified in the # column). Make copies of your screenshots from Step 1, focusing on the design elements you are considering in this step, and mark them with the unique numbers so that the reader of your report can find the location of the violation in the screenshots and read your description in the table below. In addition, color-code the violations for severity, highlighting with red, orange, yellow, green, and gray for the severity-rating scale we covered in class (with red being most severe to gray being a non-issue). For each component, you will likely note violations of some of the heuristics but not others. Only highlight violations in the table below and in the screenshots, and heuristics that are not violated can be left blank.


Search
Completed Courses
Recommended Courses
Cart
Search and Filter
Search
keyword search
Subject
All
Interest
All
Credits
minimum
to
maximum
Introduction to Psychology
PSYCH 202 - 3 credits
Behavior, including its development, motivation, frustrations, emotion, intelligence, learning, forgetting, personality, language, thinking, and social behavior.
View sections
Introduction to Operating Systems
COMP SCI 537 - 4 credits
View sections
Programming 2
COMP SCI 300 - 3 credits
View sections
General Chemistry II
CHEM 104 - 5 credits
View sections
Programming 1
Violation 02: Unclear what purpose of page is other than searching/viewing courses. Adding sections is a hidden affordance.
Violation 01: Can't quickly tell if a course has been added, unless View Sections is clicked.
Violation 03: No functionality for users getting help, limited documentation/guidance in the application.
Introduction to Psychology
Section 0
Add Section
Section 1
Add Section



Violation 04: Inconsistent with other pages (has a title, longer cards, Credits vs. Cr.)

Violation 05: Title needs more margin on x-axis.


Dashboard
React 2 β
Copy of CS571_React 2 β_Hands
Course Search
New tab
localhost:3000
Not syncing
Search
Completed Courses
Recommended Courses
Cart
Introductory Social Psychology
PSYCH 456 - 4 Cr.
The systematic study of the individual in a social context, including social interaction, motivation, attitudes, conformity, communication, leadership, personal relationships, and behavior in small groups.
Because you liked:
PSYCH 202 | Matching Interests:
psychology
behavior
Psychology, Law, and Social Policy
PSYCH 401 - 3 Cr.
Focuses on the role that psychological principles, research evidence and social science play in the laws of U.S. society, especially in the policies and mechanisms of social control of human behavior. The course will address the ways that society defines membership, and the role of psychology in how it determines who should be excluded or restricted from open society, in order to maintain a more civil society. In addition to learning the factual information about how selected processes work in the legal and social context, students will be asked to consider the role they can play as citizens in supporting or changing these social processes. The course will take a particular interest in psycholegal issues "in action" and in learning about the clinical-legal processes used to determine the disposition of individuals considered marginal in society. Finally, the course will address the mechanisms that are used to exclude individuals from open society through criminal and civil court processes, the role of psychology as a science, and the role of psychologists as behavioral experts in criminal and civil courts, and in shaping social policies.
Because you liked:
PSYCH 202 | Matching Interests:
psychology
behavior
Violation 06: Lack of user control, as user can't do anything with these recommended courses.
Violation 08: Inefficient provision of information, as we can't act on it within this page.
Violation 07: Inconsistent card structure as previous pages.

Heuristic	#	Component 1	#	Component 2	#	Component 3
Visibility of system status		Violation: 01				
Match between real world & system						
User control & freedom		Extra Violation				Violation: 06
Consistency & standards				Violation: 04		Violation: 07
Error prevention						
Recognition rather than recall		Violation: 02				
Flexibility & efficiency of use						Violation: 08
Aesthetic & minimalist design				Violation: 05		
Help users with errors						
Help & documentation		Violation: 03				

Step 4. Develop Design Recommendations. (0.4 Points)

For each violation you identified in the previous step, provide a design recommendation for addressing it along with an indication of whether or not it is feasible to implement the recommendation as an extension of your *React 2* ***α*** deliverable. (Only recommendations that are beyond the capabilities we learned in class or beyond the scope of the project should be marked as not being feasible.) Order the table of recommendations based on the severity of the usability problem from most severe to least severe. Use the table below to describe your recommendations, adding additional rows as needed, and follow the same color-coding from the previous step for severity ratings.

#	Recommendation	Feasibility (Yes/No)
	Violation: 03 - Provide an FAQ or site documentation page to offer guidance on app's services.	Yes
	Violation: 03 - Provide a Contact section in order to give the user access to some live staff that can help them with their issue.	No
	Violation: 06 and Violation: 08 - Implement the View Section, Add to Section, Remove from Section functionality on the RecommendedCourse component. (Extra Violation: Inability to undo a "remove course" when in the Cart view. Implement this functionality.)	Yes (Yes)
	Violation: 01 - Provide a visual indication of whether a course is added or not (without having to	Yes

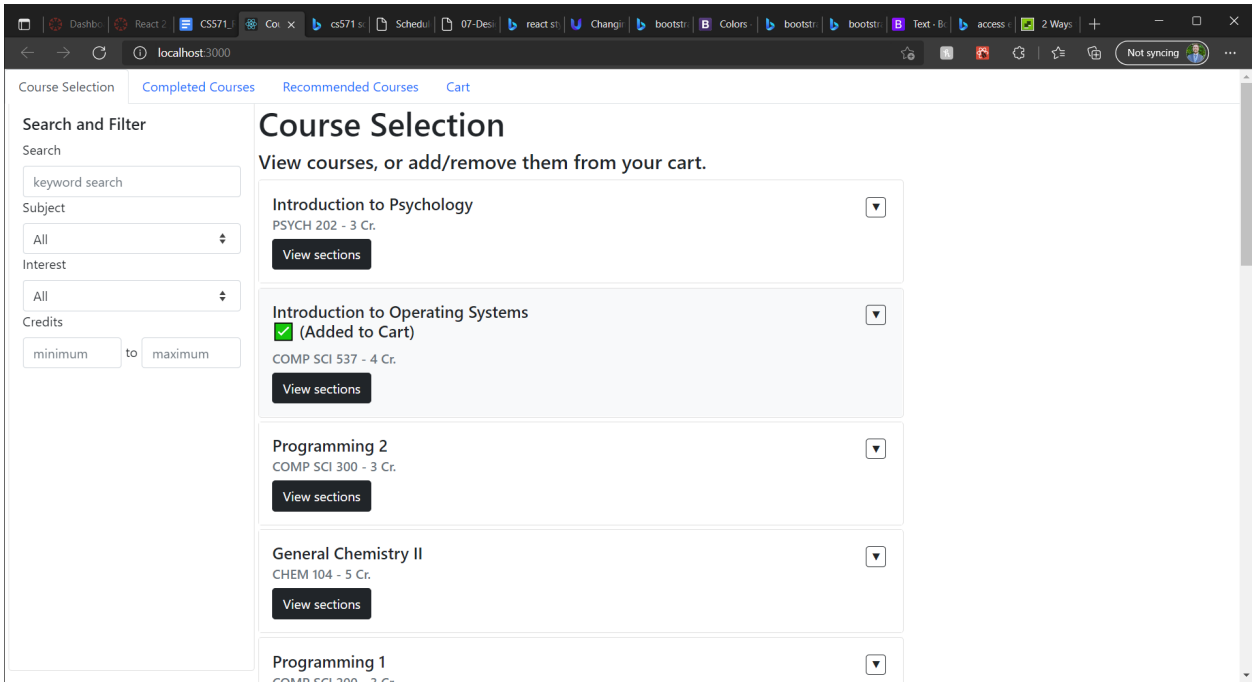
click view sections.	
Violation: 02 - Provide a visual indication that courses can be added (without having to click view sections).	Yes
Violation: 04 and Violation: 07 - Customize the cards in <Course> <RecommendedCourse> and <CompletedCourse> to be more uniform in their representation of data.	Yes
Violation: 05 - Style and/or add headers to each page describing the page's functionality.	Yes

Step 5. Implement Your Recommendations. (1.4 Points)

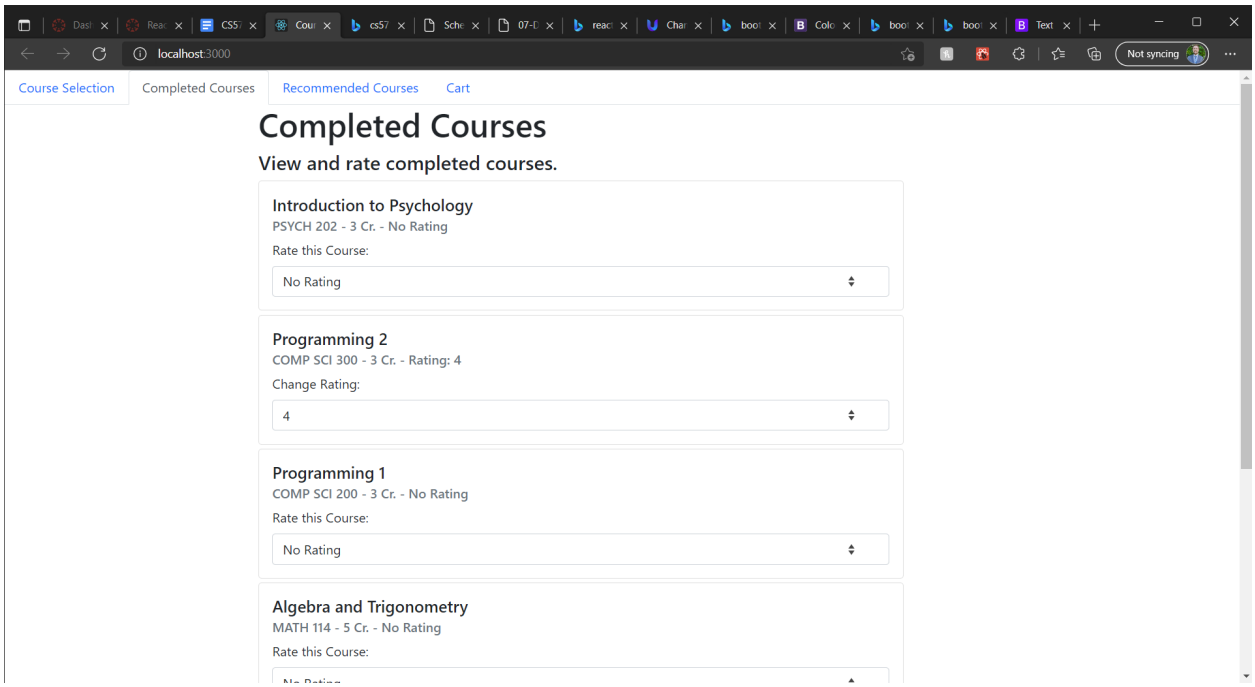
In this step, you will implement the design recommendations that you identified as "feasible" in the previous step in your prototype, updating your design. To receive full points, you will implement at least three design recommendations that can improve one or more of the components you focused on. Submit your improved React project based on instructions below and provide a paragraph that summarizes the outcome of the heuristic evaluation. In this paragraph, reflect on how your design improved, what you learned about usability in the process of applying the heuristics, and whether you gained any unexpected insights about your design.

Your deliverable will be a completed version of this document, attached to the canvas assignment as a PDF, and the GitHub Classroom repository name and latest commit hash.

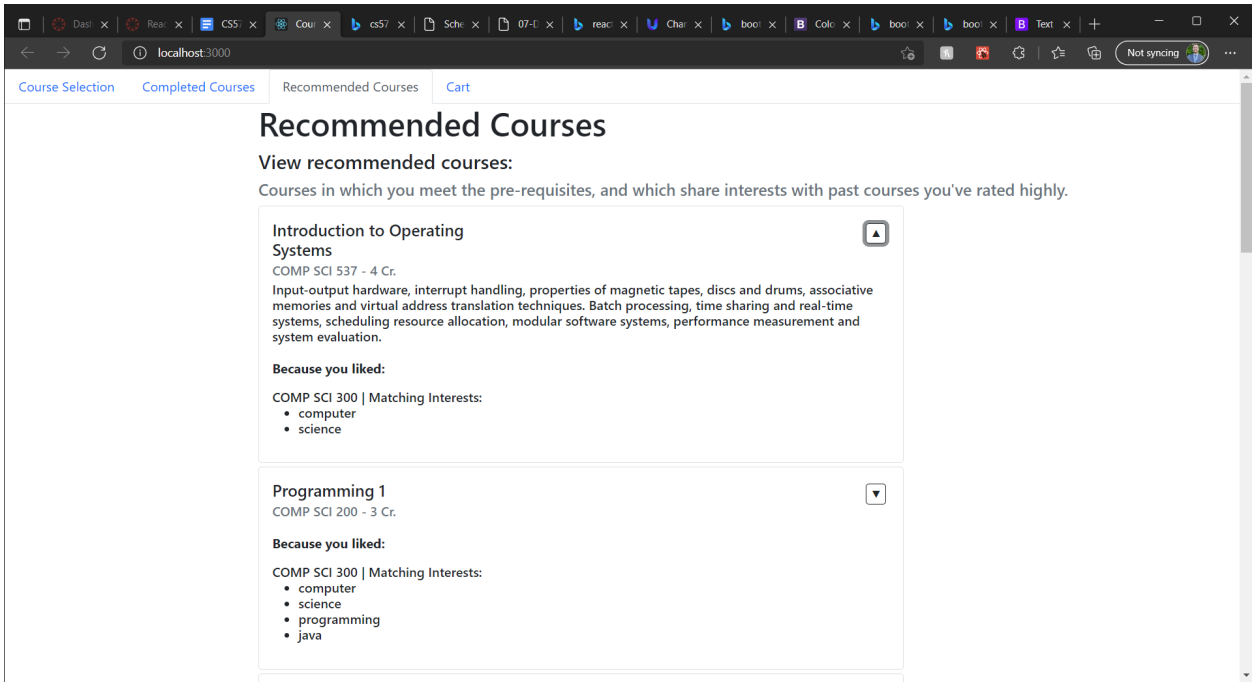
My design largely improved in terms of recognition vs. recall (the headers keep the user from needing to just try things in order to figure out functionality.) I also improved upon task status visibility, updating each course that is added with a green checkmark to set it apart from the un-added courses. Finally, I strove to make the various components more consistent with one another w/ respect to styling. In terms of usability, I am reminded that we stand on the shoulders of our predecessors, and we don't need to reinvent functional idioms that have widespread appeal. Simple things like headers, status indicators, undo buttons, etc. are quality of life features that we take for granted. These easy updates only took me around two hours to implement, which is very reasonable in the grand scheme of a large project.



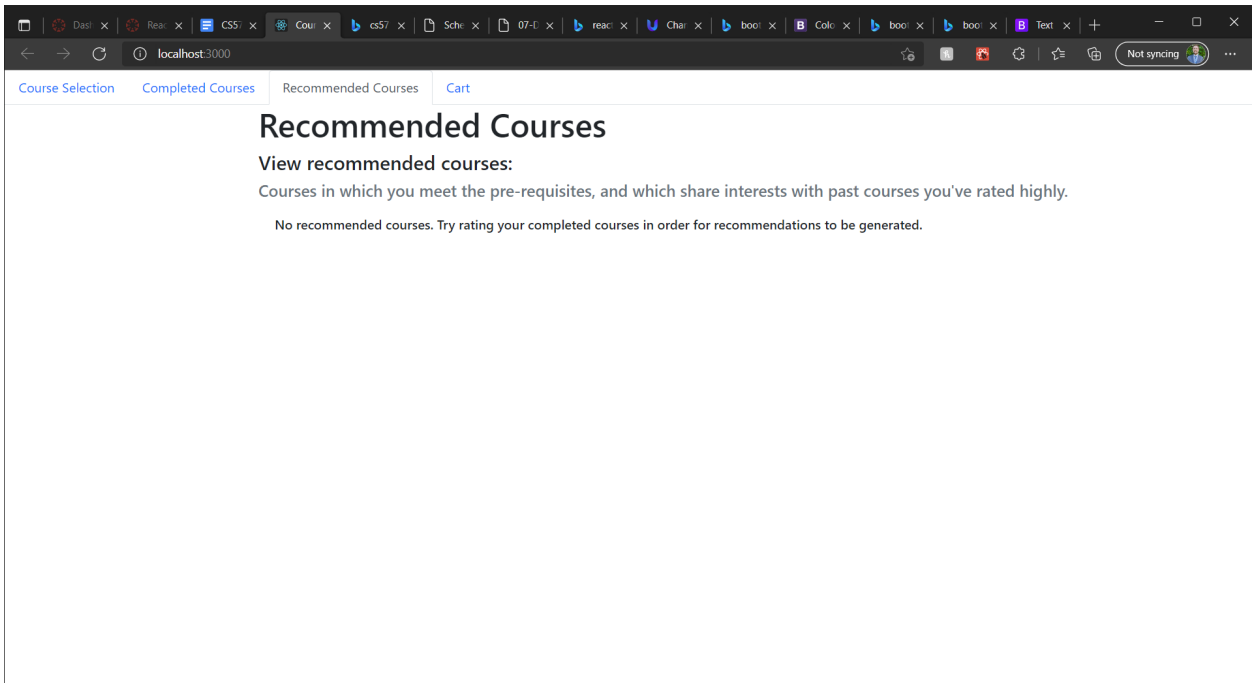
(Course Selection View: Header added, Added to Cart indicator added, course component visually uniform with rest of application.)



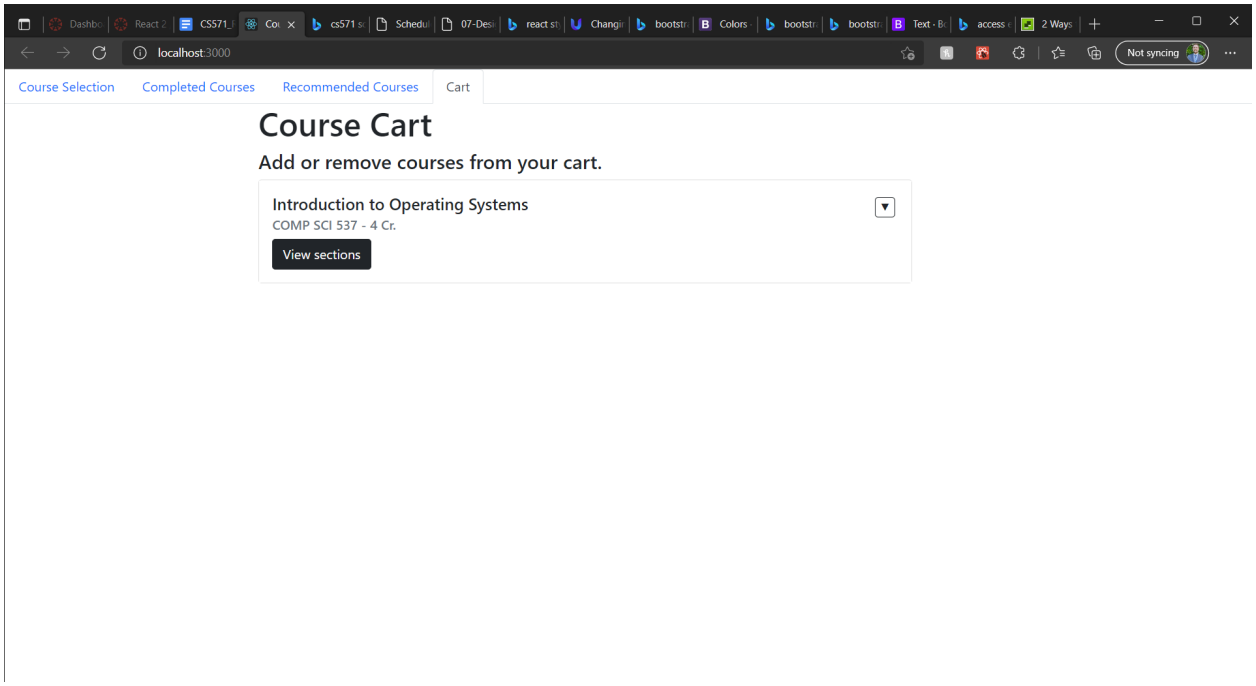
(Completed Course View: Header Added)



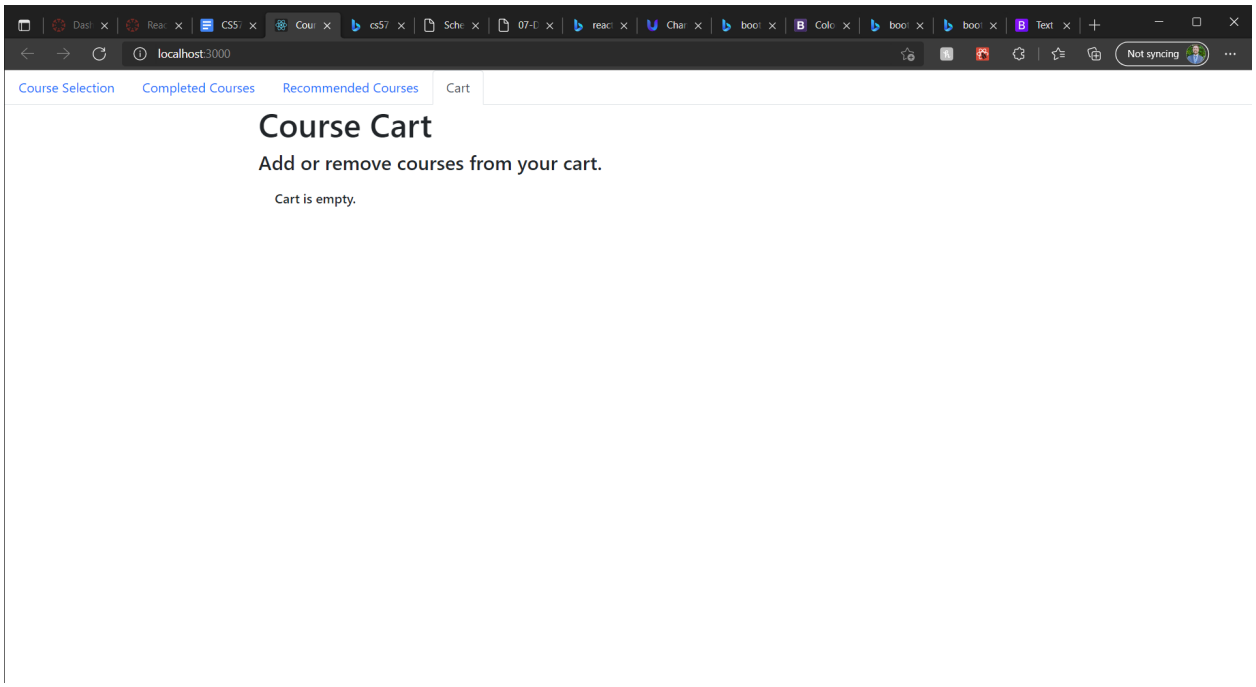
(Recommended Course View: Header added, dropdown added to give consistency with Selection/Cart views.)



(Recommended Course View: Helper Text when recommended course list is empty.)



(Course Cart View: Header added, course card doesn't show the green checkmark like it does in the course selection view.)



(Course Cart View: Helper text added notifying the user that their cart is empty. In hindsight, I should've added "To add courses to your cart, go to the Course Selection tab." I'd like to avoid going in and updating the code/doing another screenshot, though.)