React 2 β (3 Points)

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.

To complete the assignment, you will need to submit a completed version of this document as PDF to Canvas. In addition, you will submit your repository name and latest commit hash from GitHub Classroom, e.g. react2-beta-ctnelson1997, 2b0ef83.

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.

Component 1: Search and filter component

The search and filter component is critical because it is needed to narrow down the courses from a vast list of courses.

Component 2: Course and course information components

The list of courses and course information section on the right helps the user click on courses, see associated information and add courses. Hence, it is important.

Component 3: Recommended courses component

This component suggests a list of courses the user could enroll in based on the courses they have previously taken and hence is also important.

Component 4: Cart component
This component tells the user which courses have been added to the cart. Hence, it is important.

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.)
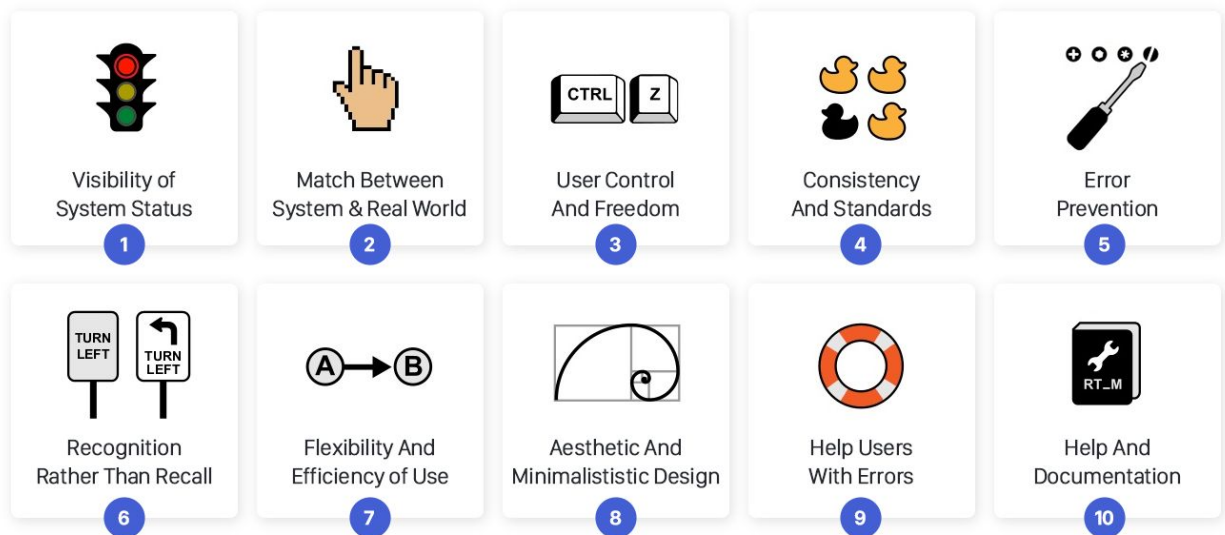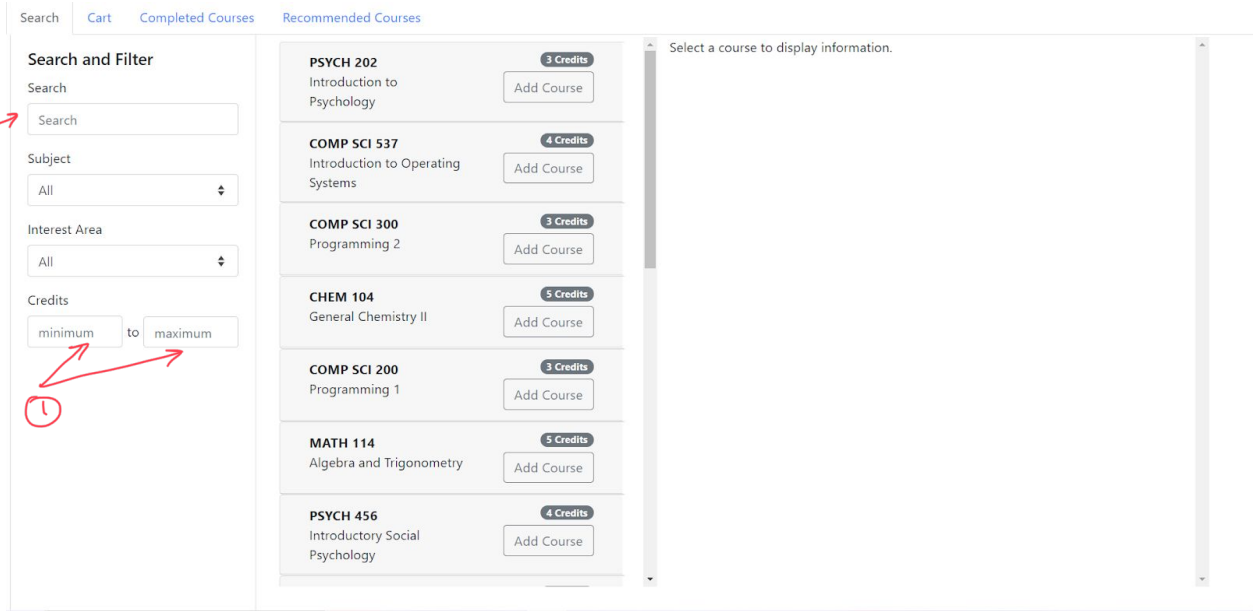

Image source: UX Collective

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.

Heuristic	#	Component 1	#	Component 2	#	Component 3
Visibility of system status						
Match between real world & system						
User control & freedom						
Consistency & standards						
Error prevention						
Recognition rather than recall					**4**	The recommended courses section is initially empty and does not provide the user with clear guidelines on how to get the list of recommended courses.
Flexibility & efficiency of use						
Aesthetic & minimalist design			**3**	The error messages saying that the user has already added the course to cart or does not satisfy the prerequisites is currently using alert boxes that are not user-friendly.		
Help users with errors	**1**	The minimum and maximum number of credits do not have any error checking mechanism to tell if the range provided is valid. No error				

	#	
		<mark>messages are displayed either.</mark>
Help & documentation	**2**	<mark>The search bar does not say what the user should input in the input field.</mark>

Heuristic	#	Component 4	#	Component 5	#	Component 6
Visibility of system status	**5**	<mark>The cart tab does not say how many courses have been added to the cart. This information is helpful to let the user know that a course has been added to the cart when the user clicks on one of the add course/section/subsection buttons.</mark>				
Match between real world & system						
User control & freedom						
Consistency & standards						
Error prevention						
Recognition rather than recall						
Flexibility & efficiency of use						
Aesthetic & minimalist design						
Help users with errors						
Help & documentation						

Component 1: Search and filter component

Component 2: Course and course information component

Component 3: Cart

Search | Cart | Completed Courses | Recommended Courses

Select a course to display information.

⑤
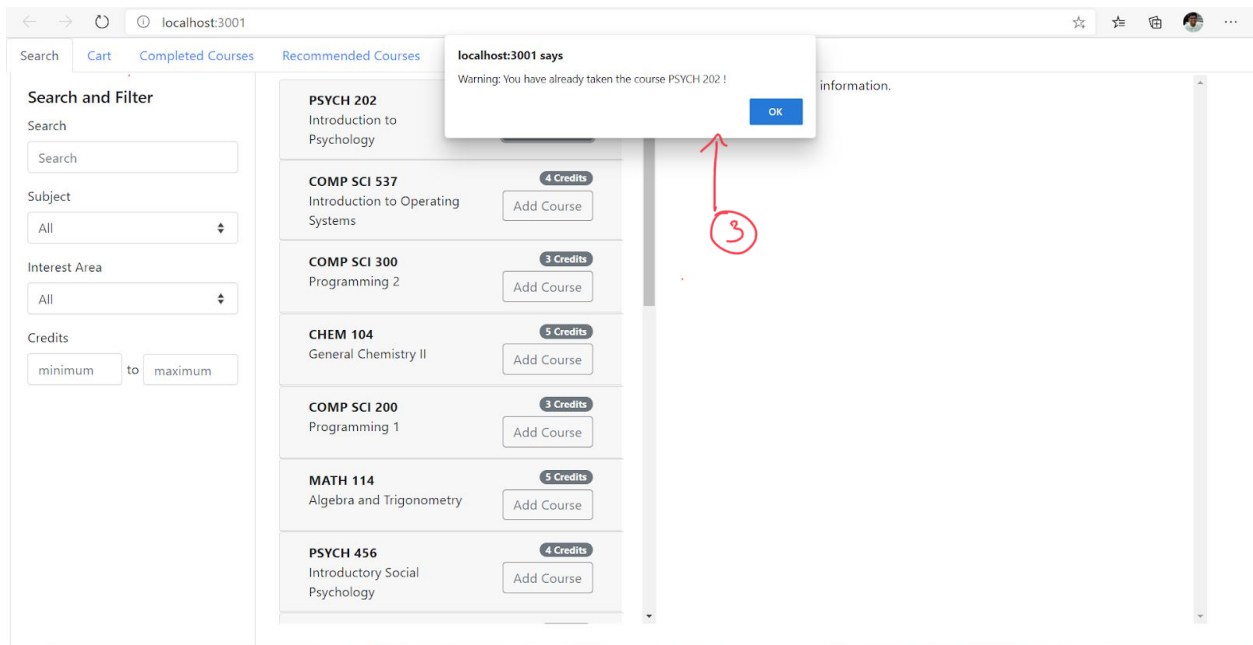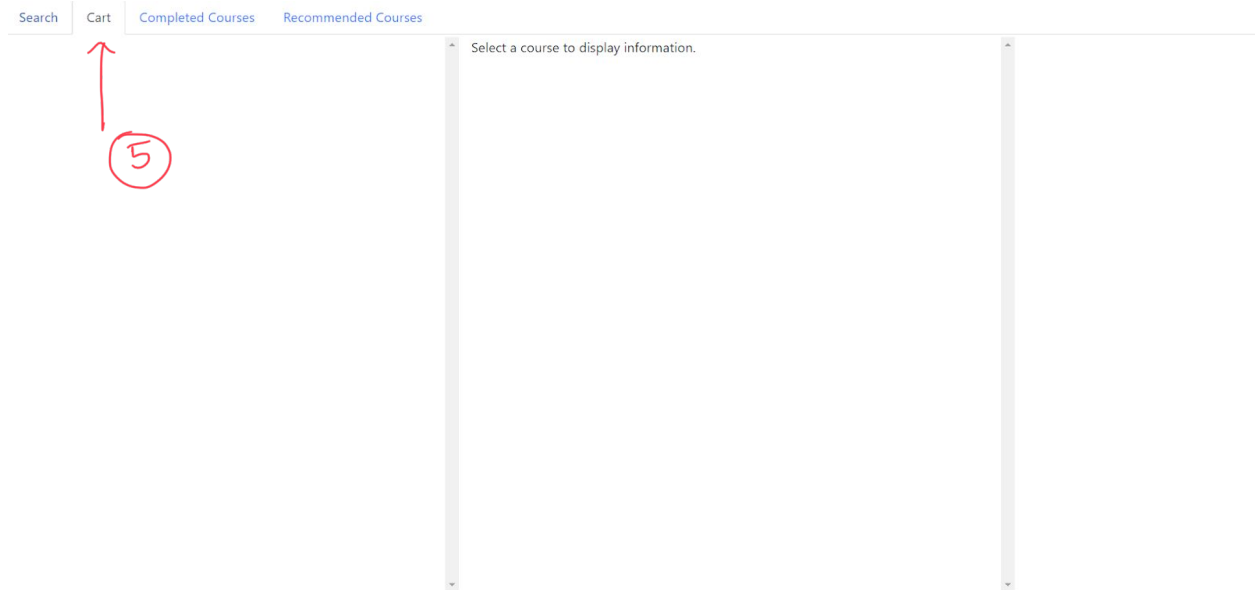
Component 4: Recommended courses
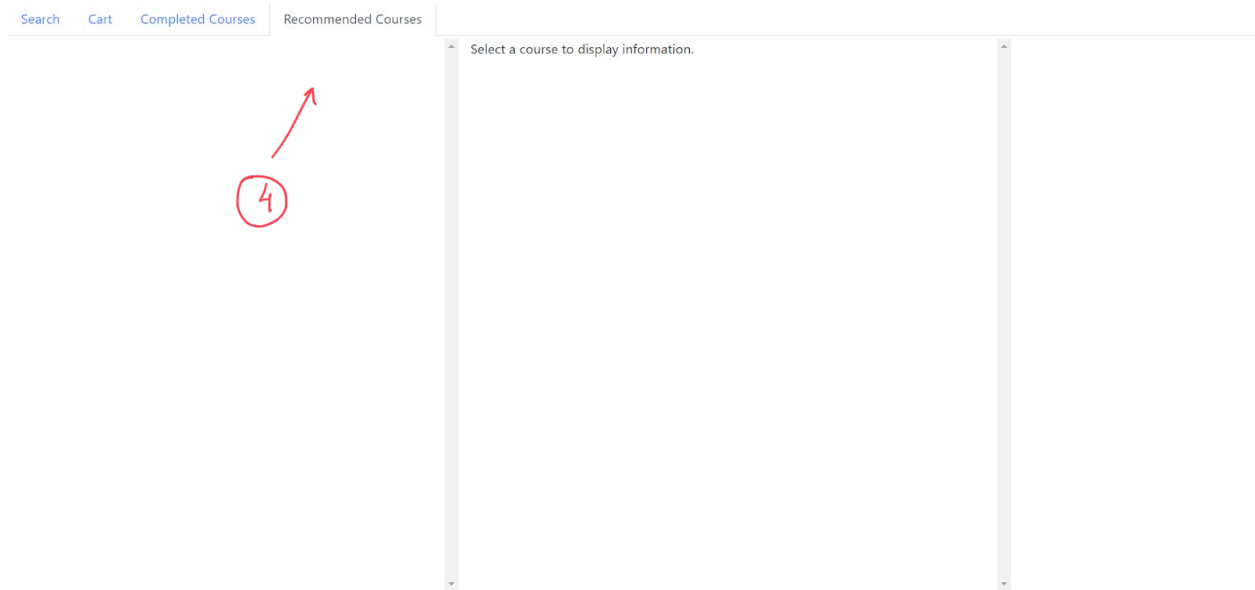
Search | Cart | Completed Courses | Recommended Courses

Select a course to display information.

④

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)
4	Provide a guideline on how to get a list of recommended courses and provide a button that redirects the user to the completed courses tab on click.	Yes
1	Provide an error message that is clearly visible (in red) when the user enters a maximum number of credits lesser than the minimum number of credits.	Yes
2	Provide a tooltip on hover over an info glyph icon near the search label saying that the search is based on keywords.	Yes
5	Provide a badge indicating the number of courses in the cart (in red, so that it's clear and catches the user's attention).	Yes
3	Use a warning modal with user-friendly error messages instead of browser alerts .	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.

4. The recommended courses section is initially empty and does not provide the user with clear guidelines on how to get the list of recommended courses. I improved the design by providing a guideline saying that the user has to rate at least one completed course to get a list of recommended courses. I also added a button that redirects the user to the completed courses tab on click for ease of use. This helps with recognition rather than relying on user recall.

Please rate courses you have completed to get a list of recommendations.   Go to completed courses!
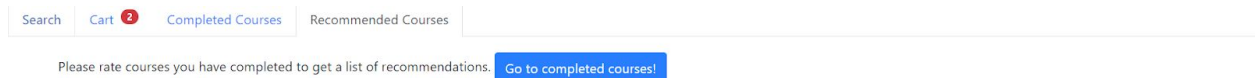
2. In my earlier design, it is not clear how the user could use the search input field to filter courses. I added a tooltip that says the courses can be filtered by typing in keywords in the search field. Thus by providing the user with documentation, I improved the usability of the search field.

← → ○ ⓘ localhost:3000   ☆ ✶≡ ⊞ 👤 ⋯

Search   Cart 1   Completed Courses   Recommended Courses

Search and Filter   Select a course to display information.

Search ⓘ ◀ Search for courses with
           specific keywords

Search

Subject

All   ▲▼

Interest Area

All   ▲▼

Credits

minimum   to   maximum

	3 Credits
PSYCH 202	
Introduction to	Add Course
Psychology	

	4 Credits
COMP SCI 537	
Introduction to Operating	Add Course
Systems	

	3 Credits
COMP SCI 300	
Programming 2	Add Course

	5 Credits
CHEM 104	
General Chemistry II	Add Course

	3 Credits
COMP SCI 200	
Programming 1	Add Course

	5 Credits
MATH 114	
Algebra and Trigonometry	Add Course

5. In my earlier design, the user did not receive any feedback on clicking the add course/section/subsection buttons. I added a badge beside the cart tab indicating the number of courses in the cart so that users can easily see this information from any page. This way the user has visibility of system status.

PSYCH 202   3 Credits
Introduction to
Psychology   Remove
Course

CHEM 104   5 Credits
General Chemistry II   Remove
Course

Select a course to display information.

1. In my earlier design, there was no error message if the user entered a maximum number of credits lesser than that entered in the minimum number of credits field. I added an error message for the same, thus helping the user with errors and understanding why no courses are shown in the list in the search page.

Search   Cart 2   Completed Courses   Recommended Courses

Search and Filter

Search🛈

Search

Subject

All

Interest Area

All

Credits

2   to   1

Please enter a valid range of credits.

Select a course to display information.

3. In my earlier design, the error messages saying that the user has already added the course to cart or does not satisfy the prerequisites is currently using alert boxes that are not user-friendly. To keep the design consistent across all browsers and devices and to provide the user with a clear error message, I

added a warning modal. There are error messages indicating whether the user has already added the course to the cart or does not satisfy the prerequisites.