

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.

- Recommended Courses Page (show the recommended courses): this would be one of the most critical components as one of the main aims of our website is to develop as a course recommendation system. Therefore, whether we show the recommended course properly is critical.

COURSE SEARCH
Search and Cart Completed Courses Recommended Courses

Recommended Courses

COMP SCI 537: Introduction to Operating Systems
Recommend this because you have gave high rating to course(s) with keyword(s): computer, science

PSYCH 456: Introductory Social Psychology
Recommend this because you have gave high rating to course(s) with keyword(s): psychology, science, behavior

COMP SCI 252: Introduction to Computer Engineering
Recommend this because you have gave high rating to course(s) with keyword(s): computer, science, programming

COMP SCI 400: Programming 3
Recommend this because you have gave high rating to course(s) with keyword(s): computer, science, programming, java

BIOLOGY 101: Animal Biology
Recommend this because you have gave high rating to course(s) with keyword(s): science

COMP SCI 354: Machine Organization and Programming

- The course component: this is a reusable for both search and cart sections. This component is critical as the search and cart sections are almost made up of “repetitions” of them. Therefore, any heuristics violation in this component would have a lot of negative effects on the whole search and cart sections. (below are screenshots for course component for search(left) and cart(right))

COMP SCI 252: Introduction to Computer Engineering
Credits: 2
Subject: Computer Science
Logic components built with transistors, rudimentary Boolean algebra, basic combinational logic design, basic synchronous sequential logic design, basic computer organization and design, introductory machine- and assembly-language programming.
Requisites: None
Keywords: computer, science, engineering, programming

Add Course

Hide Sections

Number	Instructor	Location	Meeting Times	Option
LEC_001	Joseph Krachey	1610 Engineering Hall	Friday: 2:25pm - 3:15pm Monday: 2:25pm - 3:15pm Wednesday: 2:25pm - 3:15pm	<div>Add Lecture</div>
LEC_002	Add Ibrahim	113 Brighton Psychology Building	Friday: 8:50am - 9:40am Monday: 8:50am - 9:40am Wednesday: 8:50am - 9:40am	<div>Add Lecture</div>
LEC_003	Add Ibrahim	113 Brighton Psychology Building	Friday: 12:05pm - 12:55pm Monday: 12:05pm - 12:55pm Wednesday: 12:05pm - 12:55pm	<div>Add Lecture</div>

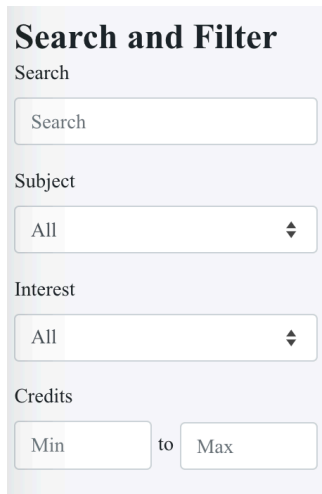
COMP SCI 354: Machine Organization and Programming
3 Credits
You did not meet the prerequisites yet!
Please complete ((COMP SCI 252)) before taking this course.

Add to WishList

Remove Course

Number	Meeting Times	Option
LEC_001	thursday: 2:30pm - 3:45pm tuesday: 2:30pm - 3:45pm	<div>Remove Lecture</div>
LEC_002	thursday: 4:00pm - 5:15pm tuesday: 4:00pm - 5:15pm	<div>Remove Lecture</div>

- The search bar: this is also important as it let users to know what courses they can take based on their interests and criteria so that they can accelerate their course search, which is important for search step for the users.



Search and Filter

Search

Search

Subject

All

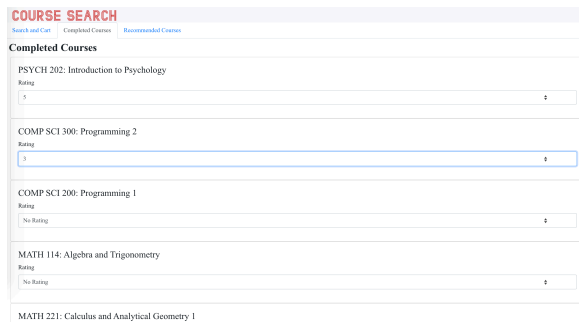
Interest

All

Credits

Min to Max

- Completed Courses Page: this page show both of the completed courses and provide users ability to rate. This is one of the most important parts of the website as we need to user to rate the course to generate recommendation. Therefore, have a proper and clear page to let the user rate correctly is critical for the recommendation system.



COURSE SEARCH

Search and Filter | Completed Courses | Recommended Courses

Completed Courses

PSYCH 202: Introduction to Psychology	Rating	5
COMP SCI 300: Programming 2	Rating	5
COMP SCI 200: Programming 1	Rating	No Rating
MATH 114: Algebra and Trigonometry	Rating	No Rating
MATH 221: Calculus and Analytical Geometry I		

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.)

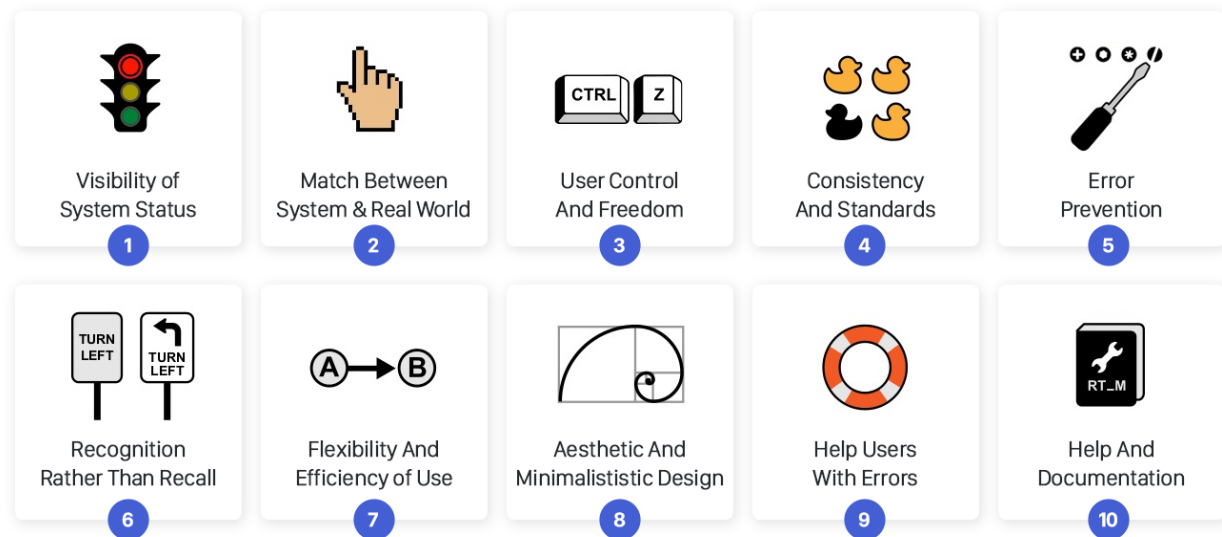


Image source: [UX Collective](#)

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	#	Component 4
Visibility of system status	1	Users are not informed when there are changes in the recommendations when they are on other pages (from where the changes to the recommendation are made)	4	After users choose to show sections, the button text changes from "Show Sections" to "Hide Sections" but this change is not really clear as it only changes in button text.				

<i>Match between real world & system</i>	2	The component does not present ways to fulfill the requisites for courses in the cart with requisites not fulfilled in a way that the user is really familiar to so that it might be hard to understand	5	In the drop down of Interests and subject. The options are not ordered in a logical way that the users are get used to, such as in alphabetical order.
<i>User control & freedom</i>				
<i>Consistency & standards</i>	3	After user add a course to the Wishlist (has requisites not fulfilled), the text in the button to add course to Wishlist changes to "Remove from the Wishlist". This makes the length of the words too long so that the button is enlarged, which breaks the consistency of buttons		
<i>Error prevention</i>				
<i>Recognition rather than recall</i>			6	Users are able to rate courses in the "Completed Course" page, but there are no clues about rating on the tab. (user will only know they can rate the courses when they are in this page)
<i>Flexibility & efficiency of use</i>				
<i>Aesthetic & minimalist design</i>				
<i>Help users with errors</i>				
<i>Help & documentation</i>				

COURSE SEARCH

Search and Filter

Completed Courses

Recommended Courses

Recommended Courses

COMP SCI 537: Introduction to Operating Systems

Recommended this because you have given high rating to courses(s) with keyword(s): computer, science

PSYCH 456: Introductory Social Psychology

Recommended this because you have given high rating to course(s) with keyword(s): psychology, science, behavior

COMP SCI 252: Introduction to Computer Engineering

Recommended this because you have given high rating to course(s) with keyword(s): computer, science, programming

COMP SCI 400: Programming 3

Recommended this because you have given high rating to course(s) with keyword(s): computer, science, programming, java

BIOLOGY 101: Animal Biology

Recommended this because you have given high rating to course(s) with keyword(s): science

COMP SCI 354: Machine Organization and Programming

Recommended this because you have given high rating to course(s) with keyword(s): computer, science, programming

1. no info about charges..

COMP SCI 537: Introduction to Operating Systems

Credits: 4

Subject: Computer Science

Input-output hardware, interrupt handling, properties of magnetic tapes, discs and drums, associative memories and virtual address translation techniques. Batch processing, time sharing and real-time systems, scheduling resource allocation, modular software systems, performance measurement and system evaluation.

Requisites: (COMP SCI 354 OR COMP SCI 400)

Keywords: computer, science, operating, system, systems

Add Course

Hide Sections

4. changes in button not really clear

Number	Instructor	Location	Meeting Times	Option
LEC_001	Andrea Arpaaci-Dusseau	1125 DeLaca Biochemistry Building	thursday: 11:00am - 12:15pm tuesday: 11:00am - 12:15pm	Add Lecture
DIS_301		2317 Engineering Hall	wednesday: 11:00am - 11:50am	Add Section
DIS_302		1325 Computer Sciences and Statistics	wednesday: 12:05pm - 12:55pm	Add Section
DIS_303		1325 Computer Sciences and Statistics	wednesday: 1:20pm - 2:10pm	Add Section
DIS_304		2255 Engineering Hall	wednesday: 3:30pm - 4:20pm	Add Section
DIS_305		1325 Computer Sciences and Statistics	wednesday: 4:15pm - 5:25pm	Add Section

COMP SCI 537: Introduction to Operating Systems

4 Credits

You did not meet the prerequisites yet!

Please complete ((COMP SCI 354) AFTER (COMP SCI 252)) OR ((COMP SCI 400)) before taking this course.

Remove from WishList

Remove Course

3. low priority as this is the only button violate the consistency

Number	Meeting Times	Option
LEC_001	thursday: 11:00am - 12:15pm tuesday: 11:00am - 12:15pm	Remove Lecture
DIS_301	wednesday: 11:00am - 11:50am	Remove Section
DIS_302	wednesday: 12:05pm - 12:55pm	Remove Section
DIS_303	wednesday: 1:20pm - 2:10pm	Remove Section
DIS_304	wednesday: 3:30pm - 4:20pm	Remove Section
DIS_305	wednesday: 4:15pm - 5:25pm	Remove Section

Search and Filter

Search

Search

Subject

All

Interest

All

Credits

Min

to

Max

5,

COURSE SEARCH

Search and Filter

Completed Courses

Recommended Courses

Completed Courses

PSYCH 202: Introduction to Psychology

Rating

5

5

COMP SCI 300: Programming 2

Rating

3

4

COMP SCI 200: Programming 1

Rating

No Rating

5

MATH 114: Algebra and Trigonometry

Rating

No Rating

5

MATH 221: Calculus and Analytical Geometry 1

Rating

No Rating

5

6. user will not know that this page also provides "rating" function when only saw the Tab.

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)
1	Highlight the “Recommended Courses” Tab once there are changes made to the page. Highlight it with background color “” to keep consistency with highlighting the cart section.	Yes
2	Change the way to present the info. E.g. for Comp Sci 537, we can present the info as below: Ways to fulfill the prerequisites: 1. COMP SCI 354 AFTER (COMP SCI 252) 2. COMP SCI 400 To make it clearer about the path to fulfill the prerequisites in a way that the users are more familiar with.	Yes
3	Make the length of all the button longer to make sure that all the texts in the button can be displayed in one line.	Yes
6	Change the tab of “Completed Courses” to “Completed Courses and Rating”	Yes
5	Sort the subjects and interests based on alphabet	Yes
4	Change the color of the button when user showed the section, i.e., different background color for “Hide Section”	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.

Reflection

- What you learned about usability in the process of applying the heuristics

With the heuristics, I learn that the usability is much more than getting the website functions work, but that we also need to present how it works in a clear and logical way. I focused on the effectiveness side a lot before but did not pay attention to the user satisfaction side. Now I understand that whether make user like the design would also impact the overall usability a lot. For example, in my design, I did bad on visibility of system status and the logics and clarity of presenting the info. For the users, they might not know that changing a rating would generate changes in the recommendations so that they may not go to the recommendation section and be confused about why they should rate and how the rating would use as the recommendations and ratings are in different pages and user are not informed of changes in the recommendation on the rating page. Also, although there is function that find the path to fulfill requisites, the path is not presented in a logical way that the users are get used to. In this way, even if we make the function to generate the recommendation works and find the path to fulfill the requisites and have a well-designed algorithm to do so, the users might simply not be able to know those recommendation or be confused by the path. So, the effectiveness is actually related and connected with satisfaction and neither of them could be ignored.

- Whether you gained any unexpected insights about your design.

When I apply the heuristics, I found that there are some details that I actually did not take into account when I designed for the project. I tried to consider every small point when finishing the previous assignments, such as adding error message to let the users change their search criteria when there is no course meets the criteria. However, without heuristics during the previous assignments, it is really hard to consider every tiny but important detail.

First, when I designed before, I focused more on macro and overall design, and functionalities instead of digging into small points and details. For example, I know that I need to tell the users that they need to change their search criteria when there are no course satisfies the current criteria (this meets the heuristics: Help users recognize, diagnose, and recover from errors) Also, I make the function get the path to fulfill requisites (which is a functionality). However, I did not take great care of smaller components, such as button sizes, which will break the consistency and did not show the path to fulfill requisites in a clear and logical way. Before, I focus more on getting it work and avoid fatal mistakes that will crash the system down. Now, I find that details are also of great importance for the usability, especially whether the information are presented clearly and logically to match what the users are familiar with, and whether the design is consistent, which will impact user's satisfaction and even affect the effect of the functionalities. Those details are what my previous assignment missed.

This inspection also makes me think more about navigation between the components. More specifically, whether I provide enough information to let the users know where each component are or inform users about what has changed. This is really important when I evaluate the visibility of system status and match between system and real world. When I look into my design with heuristics, I found that the navigation is not enough, especially in the way that the design did not inform users

of changes clearly. This might result in the failure of letting user use the website efficiently and might waste time confusing about the functionality of each section.

After the evaluation, I believe that my current design would improve especially on whether it shows the information clearly, whether it informs users the changes in system status, and would have a better consistency.

Screenshots: Implement 5 design fixes

- The recommended Courses tab is highlighted when changes was made to recommendation list (The background color and text color of the “Recommended Courses” is highlighted in the screenshot. Without highlight, it should have the same format as the other two tabs)
- Change the tab of “Completed Courses” to “Completed Courses and Rating”
- (Not implemented fully: the path to fulfill the requisites, but made some subtle modification to show it in a more logical way)

COURSE SEARCH

Search and Cart

Completed Courses and Rating

Recommended Courses

Search and Filter

Search

Subject

All

Interest

All

Credits

Min

to

Max

Course List

PSYCH 202: Introduction to Psychology

Credits: 3

Subject: Psychology

Behavior, including its development, motivation, frustrations, emotion, intelligence, learning, forgetting, personality, language, thinking, and social behavior.

Requisites: None

Keywords: psychology, behavior, emotion, intelligence, brain

Add Course

Show Sections

COMP SCI 537: Introduction to Operating Systems

Credits: 4

Subject: Computer Science

Input-output hardware, interrupt handling, properties of magnetic tapes, discs and drums, associative memories and virtual address translation techniques. Batch processing, time sharing and real-time systems, scheduling resource allocation, modular software systems, performance measurement and system evaluation.

Requisites: (COMP SCI 354 OR COMP SCI 400)

Keywords: computer, science, operating, system, systems

Course Cart

COMP SCI 537: Introduction to Operating Systems

4 Credits

You did not meet the prerequisites yet!

Ways to fulfill the requisites:

((COMP SCI 354) AFTER (COMP SCI 252))

OR ((COMP SCI 400))

Remove from WishList

Remove Course

Number	Meeting Times	Option
LEC_001	thursday: 11:00am - 12:15pm tuesday: 11:00am - 12:15pm	Remove Lecture
DIS_301	wednesday: 11:00am - 11:50am	Remove Section
DIS_302	wednesday: 12:05pm - 12:55pm	Remove Section
DIS_303	wednesday: 1:20pm - 2:10pm	Remove Section
DIS_304	wednesday: 3:30pm - 4:20pm	Remove Section
DIS_305	wednesday: 4:15pm - 5:25pm	Remove Section

- Change the color of the background of button when user show the section (from white to grey) to inform user the changes of the button text.

PSYCH 202: Introduction to Psychology

Credits: 3
Subject: Psychology

Behavior, including its development, motivation, frustrations, emotion, intelligence, learning, forgetting, personality, language, thinking, and social behavior.

Requisites: None
Keywords: psychology, behavior, emotion, intelligence, brain

[Add Course](#)

[Show Sections](#)

PSYCH 202: Introduction to Psychology

Credits: 3
Subject: Psychology

Behavior, including its development, motivation, frustrations, emotion, intelligence, learning, forgetting, personality, language, thinking, and social behavior.

Requisites: None
Keywords: psychology, behavior, emotion, intelligence, brain

[Add Course](#)

[Hide Sections](#)

Number	Instructor	Location	Meeting Times	Option
LEC_001	Jeff Henriques	105 Brogden Psychology Building	thursday: 9:30am - 10:45am tuesday: 9:30am - 10:45am	Add Lecture
LEC_002	Jeff Henriques	105 Brogden Psychology Building	thursday: 11:00am - 12:15pm tuesday: 11:00am - 12:15pm	Add Lecture
LEC_003	C. Shawn Green	105 Brogden Psychology Building	monday: 8:00am - 9:15am wednesday: 8:00am - 9:15am	Add Lecture
LEC_004	Patti Coffey	105 Brogden Psychology Building	thursday: 1:00pm - 2:15pm tuesday: 1:00pm - 2:15pm	Add Lecture
LEC_005	Sarah Gavac	105 Brogden Psychology Building	thursday: 2:30pm - 3:45pm tuesday: 2:30pm - 3:45pm	Add Lecture
LEC_006	Patti Coffey	101 Brogden Psychology Building	thursday: 2:30pm - 3:45pm tuesday: 2:30pm - 3:45pm	Add Lecture

- Sort the subject and the interest in alphabetical order so that it is more familiar to what users used in other sorting algorithm.

Search and Filter

Search

Subject

All

Interest

✓ All

algebra

algorithm

algorithms

analytical

animal

behavior

biology

brain

building

calculus

chemistry

computer

design

differential

discrete

ecology

electrical

emotion

engineering

evolution

genetics

geometry

Search and Filter

Search

Subject

✓ All

Biology

Chemistry

Computer Science

Mathematics

Psychology

Statistics

Min

to

Max