Heuristic Evaluation (Individual)

Due: Start of Thu/Fri studio (Nov 5-6)—*no credit if not turned in at start and no changes allowed after*

Objective

Apply Nielsen's adapted heuristics for evaluating and iterating on a user interface.

Learning Goal

Build familiarity with Nielsen's adapted heuristics. Understand the tradeoffs of heuristic evaluation compared to usability testing and other methods.

Role of Assignment for the Project

This feedback will eventually be synthesized into a report given to the team for the project you are evaluating, and you will in turn receive feedback from another team. This feedback will help identify issues in your prototype to improve upon before the high-fidelity version.

Assignment Overview

You have been hired as a design consultant to provide outside assistance to another team in your studio or a studio at the same time. Your CA will send you links to your assigned team's relevant materials. You will:

1. **Read through your assigned team's Assignment 6 (medium-fi prototype) materials.** This should help you get oriented on their project and prepare to evaluate their UI more critically. We recommend looking at their slides and README, and running through the prototype a couple of times.
2. **Conduct a heuristic evaluation of your assigned team's user interface**.** To do this, you will apply Nielsen's heuristics and the 2 new heuristics we've added (see Lecture 12-Heuristic Evaluation Slides and the HE list at the bottom of this spec). Focus on giving feedback on what is *currently implemented* rather than pointing out missing features.

   Make a copy of this spreadsheet to help you organize your heuristics by task. If the team has more than 3 tasks, add to the dropdown as needed. Though you are taking into consideration the tasks, we want you to evaluate the entire prototype. So, some of your evaluations may fall outside of the 3 specific tasks. If the violation occurs across all 3 tasks, label it using "All Tasks". If the violation doesn't fall under any of the specified tasks, label it using "Extra Violations". Please use the heuristics and numbering scheme from our lecture slides, also found at the end of this document. Further instructions below.

***If you're evaluating a speech-based interface or a multimodal interface with lots of voice interaction, these heuristics from this paper may be useful.*

CS 147 Autumn 2026: Assignment 7
Instructor: James Landay

Spreadsheet Instructions

Part I: Prototype Description and Tasks

Write a *one-sentence* description of the project you are evaluating. Make sure that your sentence includes the name of the product; the realization being used (ex. “Mobile app”); mention of the problem space; and a brief explanation of how the product provides a solution to the problem. Following the one-sentence description, specify each of the 3 tasks.

Part II: List of violations

Each violation in your list will include the heuristic violated (number and title), the related task, the violation severity, the problem description, the rationale for why it violates that heuristic, and a recommendation to fix the problem. Use the ratings defined in lecture: 0 = not a problem, 1 = cosmetic, 2 = minor, 3 = major, 4 = UI catastrophe.

Organize your list of violations from H1 to H12 (this means **sort** them). You should aim to identify *at least* 3 real violations for each task. Prioritize identification of violations of legitimate high severity (i.e., severity 3 or 4) in your investigation. Additionally, aim to cover a diverse spread of violations across different categories (from H1 to H12).

Here is an example description, rationale, and fix (formatting will look slightly different on the spreadsheet, but the idea is the same). The description should include what the violation is and where it was found. When appropriate, consider using images to explain your rationale, grounding your recommendations based on the target user, and/or sketching potential fixes to provide even better feedback.

1. H4 Consistency & Standards | 1. Simple Task | 3

   Description: The interface used the string “Save” on the first screen for saving the user’s information, but used the string “Store” on the second screen.
   Rationale: Users may be confused by this inconsistent terminology for the same function.
   Fix: Use “Save” on all screens.

Part III: Overall Recommendations

On the Summary tab, close with 1-2 paragraphs covering general impressions, trends identified across the heuristic evaluation, and *at least* 2 overall recommendations you have for improving the interface, including any problems you found that didn’t necessarily fit into the heuristics. For each recommendation, include what the recommendation is and a justification grounded in knowledge about the target user base, lecture slides, and/or outside research.

Deliverables

You will individually submit a link to your Google Spreadsheet report through this Google form. Make sure your CA can **EDIT** the file and *that the general access permissions are set so that anyone at Stanford University with the link can view the document.*

CS 147 Autumn 2026 website
https://hci.stanford.edu/courses/cs147/2026/au/

- **Individual Report**
  Please name your spreadsheet *[YourName]-[ProjectYouEvaluated]-HE*.

Examples

***Note: this assignment spec has undergone some changes, but the quality of the work in this example still stands (we required manual documentation rather than providing a spreadsheet when Example1 was written).*
Example1, Example 2 (24au)

Grading Criteria

You will be graded on coverage of the issues present in the current user interface design, clarity of your violation descriptions, and quality of your recommendations. Reports that focus excessively on features that are missing will be marked down.

Report (100 pts)
Project Description (5)
____ Accurately and succinctly describes the project

Exceed expectations with:
_____ Conveying the problem and solution

Violations (65)
____ Found a large percentage of the violations
____ Thorough coverage of violations present in each task
____ Heuristic variety in violations found; Non-repetitive violations
____ Found some less obvious violations in addition to the more obvious ones
____ Descriptions of violations clear and easy to understand
____ Rationale for the heuristic used is clear and valid

Exceed expectations with:
____ Finding more and a greater variety of violations, across all the tasks
____ Use of images or sketches to convey issues or fixes

Summary & Recommendations (30)
____ General impressions and trends noticed across violations found
____ Includes feedback and recommendations that don't fit into the violations

Exceed expectations with:
____ Considering project's values
____ Incorporating design patterns or specific graphic design principles
____ Grounding recommendations in details about the project (e.g., earlier assignments)

CS 147 Autumn 2026: Assignment 7
Instructor: James Landay

12 Usability Heuristics

Adapted from Nielsen (2nd version). These are 12 general principles for user interface design (including H11 and H12 added by the CS147 teaching staff). They are called “heuristics” because they are more in the nature of rules of thumb than specific usability guidelines.

H1. Visibility of system status
The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.

H2. Match between system and the real world
The system should speak the users' language, with words, phrases and concepts familiar to the user, rather than system-oriented terms. Follow real-world conventions, making information appear in a natural and logical order.

H3. User control and freedom
Users often choose system functions by mistake and will need a clearly marked "emergency exit" to leave the unwanted state without having to go through an extended dialogue. Support undo and redo.

H4. Consistency and standards
Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.

H5. Error prevention
Even better than good error messages is a careful design which prevents a problem from occurring in the first place. Either eliminate error-prone conditions or check for them and present users with a confirmation option before they commit to the action.

H6. Recognition rather than recall
Minimize the user's memory load by making objects, actions, and options visible. The user should not have to remember information from one part of the dialogue to another. Instructions for use of the system should be visible or easily retrievable when appropriate.

H7. Flexibility and efficiency of use
Accelerators -- unseen by the novice user -- may often speed up the interaction for the expert user such that the system can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

H8. Aesthetic and minimalist design
Dialogues should not contain information which is irrelevant or rarely needed. Every extra unit of information in a dialogue competes with the relevant units of information and diminishes their relative visibility.

H9. Help users recognize, diagnose, and recover from errors
Error messages should be expressed in plain language (no codes), precisely indicate the problem, and constructively suggest a solution.

H10. Help and documentation
Even though it is better if the system can be used without documentation, it may be necessary to provide help and documentation. Any such information should be easy to search, focused on the user's task, list concrete steps to be carried out, and not be too large.

H11. Accessible design
Users can interact with the system using alternative input methods. Content is legible with distinguishable contrast and text size. Key information is upfront and not nested for screen readers. Purely visual or auditory content has text-based alternatives for users with low vision and low hearing.

H12. Value alignment and inclusion
The design should encode values that users can understand and relate to. It should make a diverse group of users feel included and respected. The design should prevent the reproduction of preexisting inequities and not create additional burdens for disadvantaged populations.