

Heuristic Evaluation Workbook

Template

Evaluator

Date

Product

Task

Heuristic #01

Visibility of System Status

The design should always keep users informed about what is going on through appropriate feedback in a reasonable time.

1. Does the design clearly communicate its state?
2. Is feedback presented quickly after user actions?

✖ Issues

✔ Recommendations

Evaluator

Date

Product

Task

Heuristic #02

Match Between the System and the Real World

The design should speak the users' language. Use words, phrases, and concepts familiar to the user, rather than internal jargon. Follow real-world conventions, making information appear in a natural and logical order.

1. Will user be familiar with the terminology used in the design?
2. Do the design's controls follow real-world conventions?

✖ Issues

✔ Recommendations

Evaluator

Date

Product

Task

Heuristic #03

User Control and Freedom

Users often perform actions by mistake. They need a clearly marked "emergency exit" to leave the unwanted action without having to go through an extended process.

1. Does the design allow users to go back a step in the process?
2. Are exit links easily discoverable?
3. Can users easily cancel an action?
4. Is Undo and Redo supported?

✖ Issues

✔ Recommendations

Evaluator

Date

Product

Task

Heuristic #04

Consistency and Standards

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

1. Does the design follow industry conventions?
2. Are visual treatments used consistently throughout the design?

✖ Issues

✔ Recommendations

Evaluator

Date

Product

Task

Heuristic #05

Error Prevention

Good error messages are important, but the best designs carefully prevent problems 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.

1. Does the design prevent slips by using helpful constraints?
2. Does the design warn users before they perform risky actions?

✖ Issues

✔ Recommendations

Evaluator

Date

Product

Task

Heuristic #06

Recognition Rather Than Recall

Minimize the user's memory load by making elements, actions, and options visible. The user should not have to remember information from one part of the interface to another. Information required to use the design (e.g. field labels or menu items) should be visible or easily retrievable when needed.

1. Does the design keep important information visible, so that users do not have to memorize it?
2. Does the design offer help in-context?

✖ Issues

✔ Recommendations

Evaluator

Date

Product

Task

Heuristic #07

Flexibility and Efficiency of Use

Shortcuts — hidden from novice users — may speed up the interaction for the expert user such that the design can cater to both inexperienced and experienced users. Allow users to tailor frequent actions.

1. Does the design provide accelerators like keyboard shortcuts and touch gestures?
2. Is content and functionality personalized or customized for individual users?

✖ Issues

✔ Recommendations

Evaluator

Date

Product

Task

Heuristic #08

Aesthetic and Minimalist Design

Interfaces should not contain information that is irrelevant or rarely needed. Every extra unit of information in an interface competes with the relevant units of information and diminishes their relative visibility

1. Is the visual design and content focused on the essentials?
2. Have all distracting, unnecessary elements been removed?

✖ Issues

✔ Recommendations

Evaluator

Date

Product

Task

Heuristic #09

Help Users Recognize, Diagnose, and Recover from Errors

Error messages should be expressed in plain language (no error codes), precisely indicate the problem, and constructively suggest a solution.

1. Does the design use traditional error message visuals, like bold, red text?
2. Does the design offer a solution that solves the error immediately?

✖ Issues

✔ Recommendations

Evaluator

Date

Product

Task

Heuristic #10

Help and Documentation

It's best if the system doesn't need any additional explanation. However, it may be necessary to provide documentation to help users understand how to complete their tasks.

1. Is the visual design and content focused on the essentials?
2. Have all distracting, unnecessary elements been removed?

✖ Issues

✔ Recommendations