STAGE 1: THE INTIMIDATED INITIATE

Wow, justified-aligned columns are so orderly, and some of these words are big.
STAGE 2: THE DISILLUSIONED IDEALIST

This work is terrible and I hate it all!
STAGE 3:
THE RESIGNED PRAGMATIST

Hmm, I tried something kind of like this once and it was hard.
HEURISTIC EVALUATION
Heuristic Evaluation

“An agreed-upon set of usability best practices can help detect usability problems before actual users are brought in to further evaluate an interface.”

Main idea: Research/Design team or UX expert uses a set of heuristics to inspect for UX problems.
Other Resources


What Heuristics to Use...?
What Heuristics to Use...?

- Nielsen 1994 (Nielsen and Molich 1990)

- Ben Shneiderman’s Eight Golden Rules of Interface Design

- (Wikipedia) Gerhardt-Powals' cognitive engineering principles

- (Wikipedia) Weinschenk and Barker classification

- ...?
Nielsen’s 10 Usability Heuristics for User Interface Design

1. Visibility of system status
2. Match between system and the real world
3. User control and freedom
4. Consistency and standards
5. Error prevention
6. Recognition rather than recall
7. Flexibility and efficiency of use
8. Aesthetic and minimalist design
9. Help users recognize, diagnose, and recover from errors
10. Help and documentation
#1: Visibility of system status

“The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.”
#1: Visibility of system status
#1: Visibility of system status
#1: Visibility of system status
#1: Visibility of system status

- With Amazon Echo...?
#1: Visibility of system status

- With Amazon Echo...?
#1: Visibility of system status
#2: 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.”

•

•
#2: 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.”

- Familiar Language
- Leverage Familiarity with Real-World Objects and Activities
#2: Match between system and the real world
#2: Match between system and the real world

![Gauge indicating underperformance with 0 vs 987]
There are many reasons why NYC will lead smart building adoption. Ambitious legislation, strong incentives, and industry champions are some, but not all. The city offers a cheap renewable source of recent college grads with top skills, funding, state-of-the-art technology, growing population trends, an old housing stock in need of retrofit, smart city challenges, large number of urban incubators, and even the city size to attract partner cities (Helsinki, Paris).

There’s this mentality of doing, getting a quick buck, not building unicorns but small exits, and a mix of industries and businesses that’s hard to find somewhere else. In the end, there’s even this idea that if you can make it here, then you can make it anywhere. Time will tell who wins the smart building race at the end, but NYC sure has a good chance at it.
#2: Match between system and the real world
#3: 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.”
#3: 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.”

https://www.slideshare.net/hanochaloni/bad-error-messages
#3: User control and freedom
“Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.”
#4: Consistency and standards
#4: Consistency and standards

Hi and welcome to User Inyerface, a challenging exploration of user interactions and design patterns.

To play the game, simply fill in the form as fast and accurate as possible.

Please click HERE to GO to the next page.
#4: Consistency and standards
#4: Consistency and standards

https://www.slideshare.net/hanochaloni/bad-error-messages
#4: Consistency and standards
#5: 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.”
Two Types of Errors

**Slips** occur when users intend to perform one action, but end up doing another (often similar) action. For example, typing an “i” instead of an “o” counts as a slip; accidentally putting liquid hand soap on one’s toothbrush instead of toothpaste is also a slip. Slips are typically made when users are on autopilot, and when they do not fully devote their attention resources to the task at hand.
Two Types of Errors

**Mistakes** are made when users have goals that are inappropriate for the current problem or task; even if they take the right steps to complete their goals, the steps will result in an error. For example, if I misunderstood the meaning of the oil-pressure warning light in my car, and thought it was the tire-pressure monitor, no matter how carefully I added air to my tires, it would not fix the issue with my oil pressure...Mistakes are conscious errors, and often (though not exclusively) arise when a user has incomplete or incorrect information about the task, and develops a mental model that doesn’t match how the interface actually works.
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.

Include Helpful Constraints
#5: Error prevention
#5: 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.”

• Include Helpful Constraints
• Offer Suggestions
#5: Error prevention
#5: 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.”

- Include Helpful Constraints
- Offer Suggestions
- Choose Good Defaults
#5: 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.”

- Include Helpful Constraints
- Offer Suggestions
- Choose Good Defaults
- Use Forgiving Formatting
#5: Error prevention
#5: Error prevention
#6: 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 whenever appropriate.”
#6: Recognition rather than recall
#6: Recognition rather than recall
“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.”
#7: Flexibility and efficiency of use

![Keyboard Shortcuts and Menus](image)

### Application Menu Command | Shortcut
---|---
Save | Ctrl+S
Save As... | Shift+Ctrl+S
Check In... | Alt+Ctrl+S
Revert | F12
Export> | 
Quick Export as JPG | Alt+Shift+Ctrl+S
Export As... | Alt+Shift+Ctrl+S
Export Preferences... | 
Alt+Shift+Ctrl+S is already in use and will be removed from File > Export > Quick Export as JPG if accepted.
#7: Flexibility and efficiency of use
“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.”
#8: Aesthetic and minimalist design
#9: 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.”
#9: Help users recognize, diagnose, and recover from errors

- **Explicit** indication that something has gone wrong
#9: Help users recognize, diagnose, and recover from errors

- **Explicit** indication that something has gone wrong
- **Human-readable** language
#9: Help users recognize, diagnose, and recover from errors

- **Explicit** indication that something has gone wrong
- **Human-readable** language

[https://www.slideshare.net/hanochaloni/bad-error-messages](https://www.slideshare.net/hanochaloni/bad-error-messages)
#9: Help users recognize, diagnose, and recover from errors

- **Explicit** indication that something has gone wrong
- **Human-readable** language
- **Polite** phrasing
#9: Help users recognize, diagnose, and recover from errors

- **Explicit** indication that something has gone wrong
- **Human-readable** language
- **Polite** phrasing
- **Precise** descriptions of exact problems
#9: Help users recognize, diagnose, and recover from errors

- **Explicit** indication that something has gone wrong
- **Human-readable** language
- **Polite** phrasing
- **Precise** descriptions of exact problems
- **Constructive advice** on how to fix the problem
#9: Help users recognize, diagnose, and recover from errors

- **Explicit** indication that something has gone wrong
- **Human-readable** language
- **Polite** phrasing
- **Precise** descriptions of exact problems
- **Constructive advice** on how to fix the problem

https://www.slideshare.net/hanochaloni/bad-error-messages
#9: Help users recognize, diagnose, and recover from errors

- **Explicit** indication that something has gone wrong
- **Human-readable** language
- **Polite** phrasing
- **Precise** descriptions of exact problems
- **Constructive advice** on how to fix the problem
- **Visible** and highly noticeable
#9: Help users recognize, diagnose, and recover from errors

- **Explicit** indication that something has gone wrong
- **Human-readable** language
- **Polite** phrasing
- **Precise** descriptions of exact problems
- **Constructive advice** on how to fix the problem
- **Visible** and highly noticeable
- **Preserve** as much as the user's work as possible
#9: Help users recognize, diagnose, and recover from errors

- **Explicit** indication that something has gone wrong
- **Human-readable** language
- **Polite** phrasing
- **Precise** descriptions of exact problems
- **Constructive advice** on how to fix the problem
- **Visible** and highly noticeable
- **Preserve** as much as the user's work as possible
- **Reduce** the work of correcting the error
#10: 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.”
#10: Help and documentation

?
Ben Shneiderman’s Eight Golden Rules of Interface Design

1. Strive for consistency.
2. Seek universal usability.
3. Offer informative feedback.
4. Design dialogs to yield closure.
5. Prevent errors.
6. Permit easy reversal of actions.
7. Keep users in control.