

# Usability Heuristics and Laws of UX

## **Embedded Interface Design**

with **Bruce Montgomery**



# Learning Objectives

- Students will be able to...
  - Recognize and apply usability heuristics in UX analysis
  - Consider use in research, design, and test phases of UX work
  - Understand what design principles are and how they affect UX design
  - Review some common design principle sets
  - Look at common “laws” of UX, and consider how they may apply to UX design efforts



# Heuristics, Principles, and Laws

- By definition [1]
  - Heuristics are practical methods, rules of thumb, that may involve trial and error
  - Principles are more concrete fundamentals, a theorem that has applications across a given field
  - Laws are even more concrete statements of fact
- In our discussions of usability and UX related design heuristics, principles, and laws it is safe to consider all three to be guidelines to effective design based on lessons learned or areas researched by other UX practitioners



# Heuristic Evaluation

- Time: an hour to hours, based on how extensive the review and report out is; the method can be applied to any level of fidelity of design
- In a heuristic evaluation, reviewers, ideally experienced usability designers, review an interface and compare it against a set of accepted usability principles; the analysis results in a list of potential usability issues
- Note: a heuristic evaluation should not replace usability testing
  - Although heuristics provide criteria that affect your device's usability, the issues identified in a heuristic evaluation will likely be different than those found in a user-based usability test
- Reference [2]

# Usability Heuristics

## 1. **Visibility of system status**

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

## 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.

## 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.

- The most famous heuristics set is Jakob Nielsen's ten usability heuristics, presented here, originally developed in 1990 [3]
- Again, they are described as "general principles" for UX, or broad rules of thumb



# Usability Heuristics

## 4. **Consistency and standards**

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

## 5. **Error prevention**

Even better than good error messages is a careful design which prevents a problem from occurring. Eliminate error-prone conditions or check for them; present users with a confirmation before they commit.

## 6. **Recognition rather than recall**

Minimize user's memory load by making objects, actions, and options visible. The user should not have to remember information from one dialogue to another. Instructions should be visible or retrievable.

## 7. **Flexibility and efficiency of use**

Accelerators – unseen by a novice – may speed up the interaction for an expert user such that the system supports both inexperienced and experienced users. Allow users to tailor frequent actions.



# Usability Heuristics

## 8. Aesthetic and minimalist design

Dialogues should not contain information which is irrelevant or rarely needed. Extra information in a dialogue competes with relevant information and diminishes relative visibility.

## 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.

## 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.



# Heuristics Exercise

- Let's do this as a group
- I need a volunteer for the board
- We'll examine a device for each of the 10 standard heuristics





# Heuristics Exercise

- Let's do this as a group
- I need a volunteer for the board
- We'll examine a device for each of the 10 standard heuristics



- 1. Visibility of system status**  
The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
- 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.
- 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.
- 4. Consistency and standards**  
Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.
- 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.
- 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.
- 7. 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.
- 8. 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.
- 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.
- 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.

# Heuristics Exercise

- Did that work?
- Did the heuristics fit examining the device, or would you change them?
- Do you think user testing would find different issues?



- 1. Visibility of system status**  
The system should always keep users informed about what is going on, through appropriate feedback within reasonable time.
- 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.
- 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.
- 4. Consistency and standards**  
Users should not have to wonder whether different words, situations, or actions mean the same thing. Follow platform conventions.
- 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.
- 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.
- 7. 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.
- 8. 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.
- 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.
- 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.

# Other UX/Usability Heuristics

- Tog's Principles of Interaction Design [4]
    - Good reference, but too long for heuristic evaluations...
  - Mobile Interfaces [5]
    - Heuristics include style of interaction and privacy
  - Medical Device Safety [6]
    - Includes clear task closure and informative feedback
  - Child E-learning Devices [7]
    - Includes interactivity and motivation to learn
  - NNGroup's Mobile/Web Usability Guidelines [8]
    - By license only, based on usability test series
  - General Embedded/Connected Device Heuristics?
    - Is it realistic to do so? Research area?
- Heuristics are often developed for specific types of systems – heuristic analysis is generally effective, but always to be used along with not instead of user testing

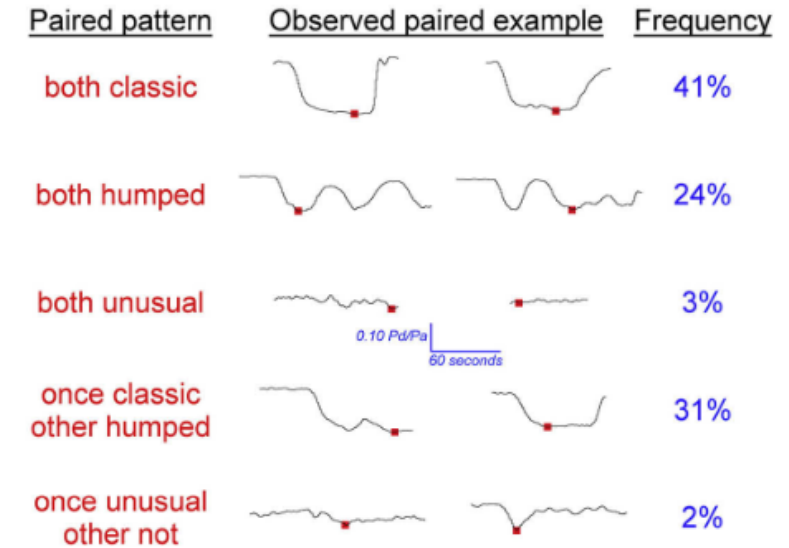


# Design Principles

- Time: large effort to develop, easier to adopt and apply to any level of fidelity of design
- Design principles, like usability heuristics, are standard guidelines for design considerations, often for special topics or concerns
- Can be used to guide aspects of a design and could be used as heuristics for a design evaluation; generally not as specific as a style guide
- Examples
  - Edward Tufte – Information Design
  - RNIB – Designing for Elderly or Disabled
  - NCSU – Seven Principles of Universal Design

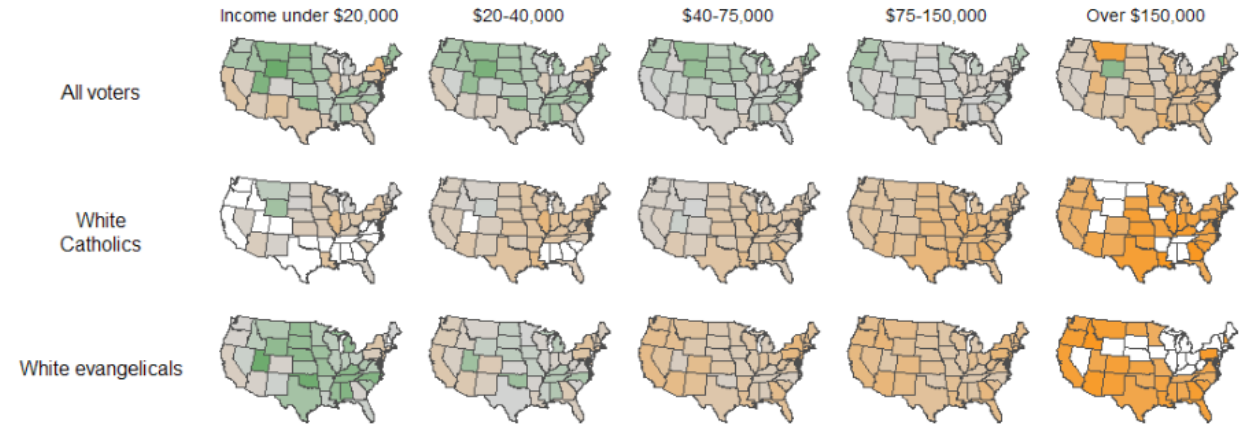
# Tufte – Information Design and Data Visualization

- Edward Tufte – Professor Emeritus at Yale
  - Statistician, political scientist, writer, sculptor
- Books:
  - Visual Display of Quantitative Information, Envisioning Information, Visual Explanations, Beautiful Evidence
- Known for his detailed books and pointed lectures, his criticism of PowerPoint, the Small Multiples (image next page) and Sparklines (see image [9]) methods, his work on the Shuttle Challenger commission, and his sculptures
- I took 15 pages of notes during a half-day lecture he gave in 2004



# Tufte – Information Design and Data Visualization

- His Grand Principles
  - Show Comparisons
  - Show Causality
  - Show Multivariate elements
  - Integrate words, numbers, and images
  - Say where data comes from
  - The quality of a presentation equals the quality, relevance, and integrity of the content
  - Use small multiples (as shown)
  - Show and embed scales of measure
  - Annotate everything
  - Use proven design templates
- Image from [10]



# Designing for Older or Disabled Adults

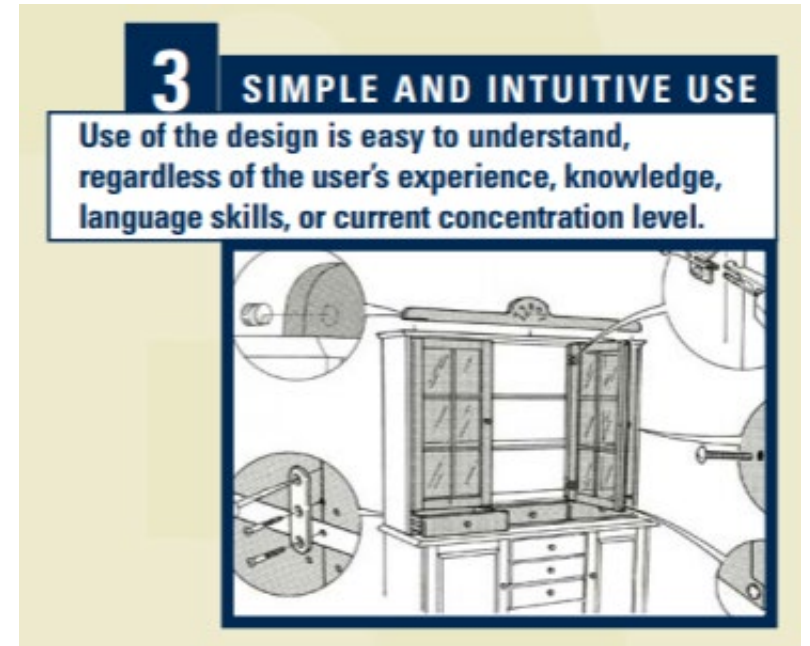
- 35 question checklist from Royal National Institute of Blind People (RNIB)
- Example Questions:
  - 6. Is the product easy to orientate? Could a visually impaired and/or older person easily locate the front, back, top, and bottom of the product?
  - 7. Does the product require little physical strength to use? Consider for example, an older person with arthritis – would they be able to lift, open, turn, grip or rotate the product to use it effectively?
- Checklist at [11]
- Also recommend the book: Fisk et al. – Designing for Older Adults [12]





# Seven Principles of Universal Design

- Developed in 1997 at N. Carolina State U. in Center for Universal Design – poster download and image from [13]
- More focused on physical device usages
- Principles include:
  1. Equitable Use
  2. Flexibility in Use
  3. Simple and Intuitive Use
  4. Perceptible Information
  5. Tolerance for Errors
  6. Low Physical Effort
  7. Size and Space for Approach and Use





# Some Common “Laws” of UX

- Miller’s Law of Short Term Memory Load: 7 +/- 2 items
  - Careful to apply to items in a person’s memory – not items in a menu
- Fitts’ Law: Mathematical equation for time to acquire a target
  - Users take longer to find smaller links or buttons and those things further from an expected home position
- Hick-Hyman Law: The time it takes to make a decision based on the number of choices
- Doherty Threshold: Productivity increases dramatically when neither the computer or user has to wait (latency <400 mSec)
- References [14], [15]



# Some Common “Laws” of UX

- Power Law of Practice: Time to perform a task decreases linearly with practice trials
  - If you know nothing, 50% of info is learned quickly, the other 50% takes more time
- Pareto (80/20) Rule: Majority of time spent on top tasks or typical problems
- Zipf’s Law: Most frequent word occurs twice as often as next frequent
- Aesthetic Usability Effect: Users often find a design more aesthetically pleasing to be more useful
- Parkinson’s Law: Any task will inflate until all of the available time is spent
- References [14], [15]



# Summary

- Heuristic Evaluation, in combination with user-based testing, is a powerful method to find typical design issues early on
- Although we discuss it in UX design, applying heuristics and related evaluation can also have application in analysis and verify/validate phases
- Using other appropriate principles and laws that apply to your particular designs and users will strengthen your design and evaluation efforts
- Next up, sketching methods...



# Next Steps

- Project 3 due ~~Monday 10/21~~ Wednesday 10/23 before class
- Wednesday: Sketching lecture, Project 4/5/6 (aka SuperProject), Project 3 demos in the lab
- Reviewed Super Project Proposals
- New quiz up this Friday
- Class staff available to help
  - Shubham - Tues 12-2 PM, Fri 3-5 PM in ECEE 1B24
  - Sharanjeet - Tues 2-3 PM, Thur 2-3 PM in ECEE 1B24
  - Bruce - Tue 9:30-10:30 AM, Thur 1-2 PM in ECOT 242
- Final Exam is set
  - Tuesday Dec 17 7:30 PM - 10 PM ECCR 1B51
  - Final will be open notes and Canvas based, you'll need a PC



# References

- [1] <https://www.lexico.com/en/definition/heuristic>
- [2] <https://www.usability.gov/how-to-and-tools/methods/heuristic-evaluation.html>
- [3] <https://www.nngroup.com/articles/ten-usability-heuristics/>
- [4] <http://asktog.com/atc/principles-of-interaction-design/>
- [5] <https://www.hindawi.com/journals/tswj/2014/434326/>
- [6] [www.sciencedirect.com/science/article/pii/S1532046403000601](http://www.sciencedirect.com/science/article/pii/S1532046403000601)
- [7] [www.jsoftware.us/vol5/js0506-14.pdf](http://www.jsoftware.us/vol5/js0506-14.pdf)
- [8] <https://www.nngroup.com/reports/mobile-website-and-application-usability/>
- [9] [https://www.edwardtufte.com/bboard/q-and-a-fetch-msg?msg\\_id=0001OR](https://www.edwardtufte.com/bboard/q-and-a-fetch-msg?msg_id=0001OR)
- [10] [https://statmodeling.stat.columbia.edu/2009/07/15/hard\\_sell\\_for\\_b/](https://statmodeling.stat.columbia.edu/2009/07/15/hard_sell_for_b/)
- [11] <http://www.designingconnectedproducts.com/35-questions-checklist/>
- [12] Designing for Older Adults, Fisk et al., 2009, CRC Press
- [13] [https://projects.ncsu.edu/ncsu/design/cud/pubs\\_p/docs/poster.pdf](https://projects.ncsu.edu/ncsu/design/cud/pubs_p/docs/poster.pdf)
- [14] <http://measuringu.com/hci-laws/>
- [15] <https://lawsofux.com/>

