

# Foundations of HCI

Material from Designing the User Interface by Shneiderman and Pleasant  
and  
The Design of Everyday Things by Don Norman



# Guidelines



# Navigation

- National Cancer Institute's guidelines (2006) - Now a book: [https://www.usability.gov/sites/default/files/documents/guidelines\\_book.pdf](https://www.usability.gov/sites/default/files/documents/guidelines_book.pdf)
- Standardize task sequences
- Ensure that embedded links are descriptive
- Use unique and descriptive headings
- Use radio buttons for mutually exclusive choices
- Develop pages that will print properly
- Use thumbnail images to preview larger images

# Accessibility Guidelines

- Section 508 published by the Access Board
- World Wide Web Consortium (W3C) adapted guidelines: <https://www.w3.org/TR/WCAG20/>
- Websites should be:
  - Perceivable: Text alternatives, Time-based media, Distinguishable, Alternate presentation
  - Operable: Keyboard access, Time, Navigation, Not seizure prone
  - Understandable: Readable, Predictable, Error prevention
  - Robust: Current and future compatibility particularly with assistive technologies

# Organizing the Display

- Consistency of data display - keep a dictionary to ensure terminology, abbreviations, formats, colors, capitalization, etc. are standardized
- Efficient information assimilation by the user - formatting (columns, decimal numbers lining up), information should be organized in a way that is appropriate and relevant to the task at hand
- Minimal memory load on the user - user's shouldn't have to remember info from screen to screen
- Compatibility of data display with data entry - displayed data should be linked to how data needs to be entered
- Flexibility for user control of data display - users should be able to get the information from the display in the form most convenient for the task on which they are working

Involve users in the  
design process whenever  
possible

# Getting User's Attention

- Intensity - two levels
- Marking - underlining, box, arrow, indicator
- Size - up to four sizes
- Fonts - up to three fonts
- Inverse video - inverse coloring
- Blinking - blinking displays/blinking color changes should be used with discretion
- Color - up to four standard colors (additional colors may be used occasionally)
- Audio - soft tones for positive feedback, harsh sounds for emergency conditions

With regard to data entry, what do you think makes the task easier from your experience?



# Data Entry

- Consistency of data-entry transactions
- Minimal input actions by user
- Minimal memory load on users
- Compatibility of data entry with data display
- Flexibility for user control of data entry

# Principles

Do all users think the  
same way?

# Determine Users' Skill Levels

- Design should begin with an understanding of the intended users: population profiles (age, gender, physical and cognitive abilities, education, cultural or ethnic backgrounds, training, motivation, goals, and personality)
- Might break users into categories:
  - Novice
  - Knowledgeable intermittent users
  - Expert frequent users
- Can you think of an example where a website is adaptable based on user preference?

# Identify the Tasks

- Most successful strategies involve long hours of observing and interviewing users
- What tasks should be supported?
  - Everything - can lead to clutter
  - Minimal - PalmPilot's designers ruthlessly limited functionality to guarantee simplicity and they were successful

Gulf of  
Execution



How the customer explained it



How the Project Leader understood it



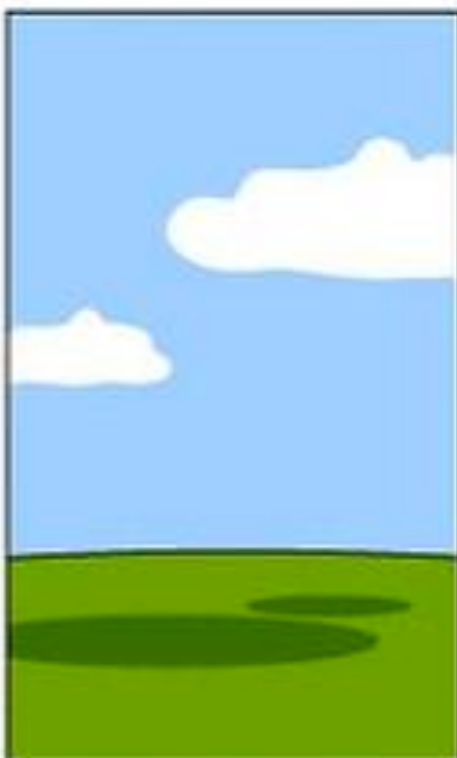
How the Analyst designed it



How the Programmer wrote it



How the Business Consultant described it



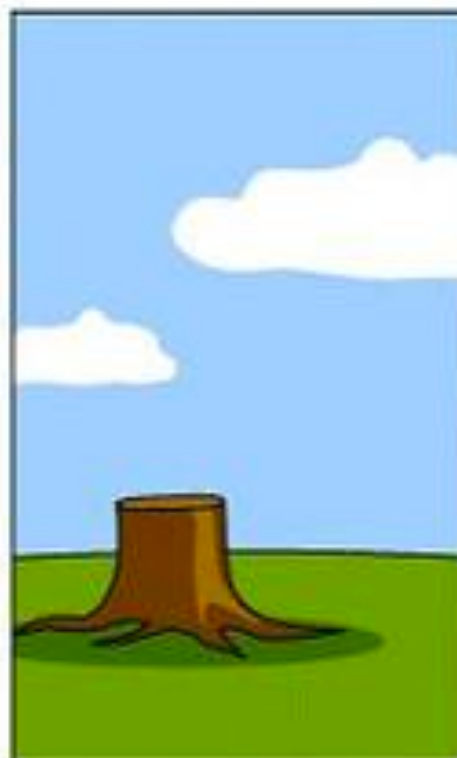
How the project was documented



What operations installed



How the customer was billed



How it was supported



What the customer really needed

# Identify the Tasks

- Atomic actions - the actions users execute with a single command, menu selection, or other action
- How do you choose the most appropriate set of atomic actions?
  - Too small - users become frustrated by the large number of actions necessary to accomplish a high-level task
  - Too large - users will need many such actions with special options, or they will not be able to get exactly what they want from the system

# Identify the Tasks

- Frequent tasks should be simple and quick to carry out
- Relative frequency - need to understand the frequency with which users will need to accomplish a specific task
- May want to create a matrix to identify the actions needed by various groups
- Word processor example:
  - How do you think the frequent actions, less frequent actions, and rare actions might be performed?



# Identify the Tasks

- Word processor example:
  - Frequent actions: pressing special keys, such as arrow keys, insert, and delete
  - Less frequent: pressing a single letter plus the Ctrl key, selection from a pull-down menu (e.g., underscore, bold, save)
  - Infrequent actions: sequence of menu selections, form fill-ins (e.g., printing format)

# Choose an Interaction Style

- Direct manipulation: visual representations (e.g., trash can on the desktop)
- Menu selection: users can scan a list of potential actions and choose what is most appropriate
- Form fill-in: allowing the user to enter the information directly rather than go through menu options can simplify the task for the user
- Command language: gives the user power (e.g., macros)
- Natural language: response to natural language (e.g., Siri)

## **Advantages**

### **Direct manipulation**

Visually presents task concepts

Allows easy learning

Allows easy retention

Allows errors to be avoided

Encourages exploration

Affords high subjective satisfaction

### **Menu selection**

Shortens learning

Reduces keystrokes

Structures decision making

Permits use of dialog-management tools

Allows easy support of error handling

### **Form fill-in**

Simplifies data entry

Requires modest training

Gives convenient assistance

Permits use of form-management tools

### **Command language**

Flexible

Appeals to “power” users

Supports user initiative

Allows convenient creation of user-defined macros

### **Natural language**

Relieves burden of learning syntax

## **Disadvantages**

May be hard to program

May require graphics display and pointing devices

Presents danger of many menus

May slow frequent users

Consumes screen space

Requires rapid display rate

Consumes screen space

Poor error handling

Requires substantial training and memorization

Requires clarification dialog

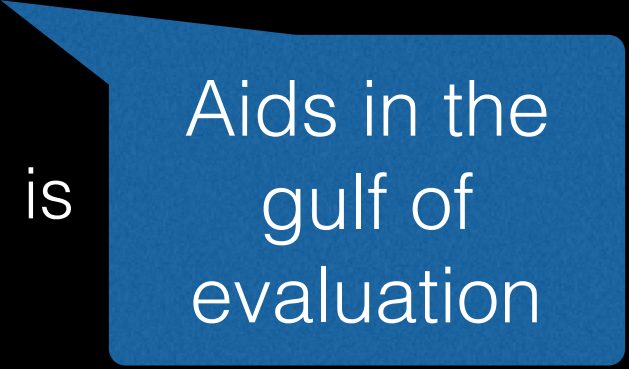
May not show context

May require more keystrokes

Unpredictable

# Eight Golden Rules of Interface Design

1. Strive for consistency
2. Cater to universal usability - novice to expert differences, age ranges, disabilities, and technology diversity
3. Offer informative feedback - for every action, there should be system feedback
4. Design dialogs to yield closure - confirm that the transaction is complete
5. Prevent errors - don't let users make serious errors if you can prevent them
6. Permit easy reversal of actions
7. Support internal locus of control - how have browsers assisted in this?
8. Reduce short-term memory load



Aids in the  
gulf of  
evaluation



# Error Prevention

- Critical!
- Superior error message: specific, positive in tone, and constructive
- Norman: prevent “slips” by...
  - organizing screens and menus functionally
  - designing commands and menu choices to be distinctive
  - making it difficult for users to take irreversible actions
  - providing feedback about the state of the interface (changing the cursor)
  - designing for consistency (Tuesday’s assignment)
- Correct actions
- Complete sequences





# Design-by-levels



- Conceptual Level: user's "mental model" of the system
- Semantic Level: describes the meanings conveyed by the user's input and by the computer's output display
- Syntactic Level: defines how the user actions that convey semantics are assembled into complete sentences that instruct the computer to perform certain tasks
- Lexical Level: deals with device dependencies and with the precise mechanisms by which users specify the syntax



<https://youtu.be/bxNSCuiPiDs>

# Seven Stages of Action

1. What do I want to accomplish?
2. What are the alternative action sequences?
3. What action can I do now?
4. How do I do it?
5. What happened?
6. What does it mean?
7. Is this okay? Have I accomplished my goal

# Seven Stages of Action

- Goal > Plan, Specify, Perform > World  
Bridge of Execution
- World > Perceive, Interpret, Compare > Goal  
Bridge of Evaluation



# Four Principles of Good Design

1. State and action alternatives should be visible
2. There should be a good conceptual model with a consistent system image
3. The interface should include good mappings that reveal the relationships between stages
4. Users should receive continuous feedback

# Declarative Memory

- What is the phone number of a friend?
- What is Beethoven's phone number?
- What is the capital of:
  - Brazil?
  - Wales?
  - US?
  - Estonia?

# Procedural Memory

- In the house you lived in three houses ago, as you entered the front door, was the doorknob on the left or right?

# Three Levels of Processing

1. Visceral - allows us to respond quickly and subconsciously
2. Behavioral - home of the learned skills, triggered by situations that match the appropriate patterns
3. Reflective - conscious cognition; place where deep understanding develops

# How do we overcome...?

- Poorly formed conceptual models? (ex: thermostat)
- Misplaced blame? (ex: exit door, enter key)
- Learned helplessness? (ex: math)

What is human error  
really?

How can technology  
accommodate users?