User Interface (UI) Design

Overview

- What is UI?
- How to design UI?
  - Design principles
  - Data display guidelines
- Miscellaneous

What Is UI?

- An effective communication medium between a human and a computer

Primary Styles of UI

- Direct manipulation
    - the user interacts with objects on the screen
    - E.g., drag a file to a "trash bin"

- Menu selection
    - E.g., select the "delete" on menu for a file

- Form fill-in
    - E.g., fill a file name and click "delete" button

Primary Styles of UI (Cont.)

- ## Command language
  - Type in delete command with the filename as a parameter

- ## Natural language
  - Type in natural language description, which will be parsed and executed
  - E.g., "delete the file named xxx"

What Is UI Design?

- Definition
  - Following a set of interface design principles, design identifies interface objects and actions and then creates a screen layout that forms the basis for a user interface prototype
- Goal
  - Easy to understand, learn, and use

Typical UI Design Errors

- lack of consistency
- too much memorization
- no guidance / help
- no context sensitivity
- poor response
- arcane/unfriendly

Mandel's Three Golden Rules

- Place the user in control
  - "What I really would like is a system that reads my mind"

- Reduce the user's memory load
  - The more a user has to remember, the more error-prone the interaction will be

- Make the interface consistent
  - "Things that look the same should act the same"

Place the User in Control

- Define interaction modes (UI) which do not force users into unnecessary actions
  - E.g., spell check
- Provide flexible interaction
  - E.g., keyboard commands, mouse movement
- Allow user interaction to be interruptible or undoable
  - E.g., Automatic save, undo, redo

Place the User in Control (Cont.)

- Allow for streamline interaction as skill levels advance (customization)
- Hide technical details from user

Reduce the User's Memory Load

- Reduce demand on short-term memory
  - E.g., autofill, single sign-on
- Establish meaningful defaults
  - "N/A", "Please specify…"
- Define intuitive shortcuts
  - E.g., "alt-P" to invoke "print" function

Reduce the User's Memory Load (Cont.)

- Base visual layout on a real-world metaphor, when possible
  - E.g., bill pay process: checkbook + check register
- Disclose information in stages
  - Use hierarchy for choices

Make Interface Consistent

- Allow understanding of current task in context
  - Window titles, graphical icons, consistent color usage, forward, backward
- Maintain consistency across a family of SW products
- If users have expectations from past interactive models, try not to make changes

Eight Golden Rules of Dialog Design

Schneiderman,1982

- Strive for consistency
- Enable frequent users to use shortcuts
- Offer informative feedback
- Design dialogs to yield closure
- Offer simple error handling
  - Try to make serious errors impossible
- Permit easy reversal of actions
- Support internal locus of control
- Reduce short-term memory load

Kinds of Users

- Novice
  - Have little knowledge about usage
  - Use small vocabulary of familiar terms
  - Give informative feedback
- Knowledgeable intermittent users
  - Know task but may forget specific details
- Frequent users
  - Want to accomplish tasks rapidly with a few keystrokes or clicks

Getting User Attention

- Use sparingly in an interface:
  - Intensity (e.g.,boldface vs. regular font)
  - Marking (e.g., underlining)
  - Size (only 4 font sizes)
  - Choice of fonts (<=3)
  - Blinking
  - Inverse video
  - Color (<=3)
  - Audio

Error Handling

- Describe the problem in the language user can understand, in non-judgmental manner
- Provide constructive advice for recovery
- Indicate any negative consequences
- Message associated with visual or audio cue

Help Facility

- How does user request help?
- How is help presented?
  - Separate window, 1-2 line suggestion at a fixed screen location, pointer to document
- How does user return to normal mode?
- Is help flat or structured?

How to Design UI?

- Understand what users need
  - Types of users?
  - Tasks users will perform with the system
    - Use-cases
  - Design task is similar to design of rest of the system
  - Offer interactions that "fit" users requirements