Designing Help, Program Flow and the Web

CS 160: User Interfaces
Maneesh Agrawala and Jeffrey Nichols

Material drawn from Seth Horrigan, Tim Wright and Jakob Nielsen

Interactive Prototype Assignment

Returned at the end of class

- Report Average: 56.5 out of 60
- Presentation Average: 21.75 out of 25
Final Presentation and Report

Due May 4\textsuperscript{th}

- Goals
  - Use the results of the Pilot Usability Study to improve your prototype
  - Implement any final functionality
- Presentation to instructors on May 4\textsuperscript{th}
- Poster session sometime that week (May 6\textsuperscript{th}?)

Today

- Errors and Help
- Aesthetics and Visual Flow
- Visual Design for the Web
Errors and Help
Errors and Help

Exercise (2 minutes)

– List 4 different errors that can occur in your group project’s user interface

– How many of these are system errors, as compared to user errors?

System Errors

• Write in the user’s language
  – Not "winword.exe" caused a segmentation fault at #F34EA01.
  – You need to understand your user group to do this well

• Precisely indicate the problem
• Constructively suggest a solution
User Errors

- **Slips** are when a user formulates the correct goal, but carries it out incorrectly
- **Mistakes** are a failure to formulate the correct goal
- **Lapses** are a failure to carry out an action (often when part of a sequence is skipped)
- **Mode errors** occur when the action would have been correct if the interface had been in a different mode

Mistakes

There are two common types of mistakes:

- **Knowledge-based mistake**: Incorrect decision/action because of a failure to understand the situation.
- **Rule-based mistake**: Understand the situation, but making a wrong decision.
Why categorize?

- **Slips** are when a user formulates the correct goal, but carries it out incorrectly
- **Mistakes** are a failure to formulate the correct goal
- **Lapses** are a failure to carry out an action (often when part of a sequence is skipped)
- **Mode errors** occur when the action would have been correct if the interface had been in a different mode

  The difference matters because it indicates the method needed to fix the user interface

Possible Causes of Errors

- Incorrect mapping of cause to effect
- Inadequate background to understand the information
- Unclear understanding of system status
- Misjudging information importance
Helping Users Learn

• How do we help users learn our system so they make fewer errors?

Help (doesn’t)

• Extra feature that can confuse users
• Spreading expensive jam onto stale toast isn’t going to make it taste better

• In a 1987 study of 52,576 help sessions:
  – 23% of all requests found no help
  – 36% of people who found help reported the help was useful (28% of total requests)
Helping Help Help

• People want answers, and want them quickly

• Descriptive questions; "What is this?"
• Procedural questions; "How do I do this?"
• Guidance questions; “What should I do?”
• Interpretive questions; "Why did that happen?"
• Navigational questions; "Where am I?“, “Where is X?"

Types of Help

• F1 help
• Hover-over help
• Separate window help
• Keyword search
• Google
• Balloon help
• Apple Guide – step-by-step
• Clippy
• Wizard
• Tutorials – videos, embedded in the program, Video Professor
• Friends
• Manuals
Cost of Help

• What is the least expensive form of help?
  – A computer interface that doesn’t need help

• What is the most expensive form of help?
  – Asking a friend

Experts and Beginners

• Who are they?
• How do we design for them?
Beginners

- User Description
  - System knowledge:
    - None
  - Domain Knowledge:
    - Unknown
  - Proficiency:
    - Low

How Beginners will Behave

- Few tasks
- Many errors
- Dependence on help (not just heavyweight help)
- Limited use of options or alternatives
Supporting Beginners

- Few options
- Visible help
- At most one task per screen
- Wizards
- Provide acquisition facilities
  - Highly visible
  - Aesthetically pleasing
  - Concentrate on ordinary, standard, typical tasks

Experts

- User Description
  - System knowledge:
    - High
  - Domain Knowledge:
    - High
  - Proficiency:
    - High
How Experts will Behave

• Many tasks
• Few errors
• Little use for Help
• Idiosyncratic style of interaction
• High use of options or alternatives
• Primary concern is efficiency and productivity

Supporting Experts

• Efficient Interaction
• Fast
• Many tasks per screen
• Provide production facilities
  – Conventional techniques to support expert use:
    • Ctrl+x, ctrl+c, ctrl+v
  – Uncluttered, customizable workspace
  – Simple icons on toolbars and dockable toolbars
  – Features that rely on user’s memory rather than visibility
E.g. Unix-style Command Line

- How many people are beginners?
  - `% cp ~/Desktop/myhouse.png ~/Desktop/pictures/myhouse.png`

- How many people are experts?
  - `% for file in $(find . -name ".png" -print ) ; do convert \
    -size 800x800 $\{file\} -resize 800x800 $\{file/\}.png-small.png \ 
    ; done`

- Most users of software are “perpetual intermediates” or “improving intermediates”

How Intermediates will Behave

- Expanding number of tasks
- System limitations become frustrating
- Intermittent need for help
- More extensive experimentation
- Evolving and changing patterns of interaction
Interfaces for Intermediates

- Allow exploration through interaction
- Show alternate mechanisms to perform tasks
- Provide transitional facilities
  - Visible shortcuts
  - Customizable interface

Aesthetics and Visual Flow
Aesthetics and Visual Flow

- How do we design something that is aesthetically pleasing?
- How do we make our design easy to comprehend?

Designing for Visual Flow

- Proximity
  - Keep related items together
- Alignment
  - Nothing should be placed arbitrarily
- Repetition
  - Repeat visual elements throughout the design (widgets, etc)
- Contrast
  - Either the same, or Very Different
Proximity

- Group related items together
- Keep unrelated items apart

Proximity

Some principles of proximity
  - Limit how much you put on one page
  - Avoid filling all corners
  - Make whitespace unequal, use it to emphasize elements
  - Group related things, don’t group unrelated things
Alignment

Visually connect elements to something else in the design

Some principles of Alignment

- Find a **strong line and use it**
- Align with something else
  - Even if it is far away
- Avoid combining multiple alignments
  - Left, centered, right, justified
- Use centered alignments sparingly
Alignment

- Some principles of Alignment
  - Find a strong line and use it
  - Align with something else
    - Even if it is far away
  - Avoid combining multiple alignments
    - Left, centered, right, justified
  - Use centered alignments sparingly

Alignment: Grids
Repetition

• Repeat aspects throughout your design
  – Layouts, fonts, grids

Look and Feel

• Consistent, repeated elements of software or web site design
  – Interaction design, and visual design
• Consistency gives a sense of “place”
  – You know where you are
  – You know which program you are using
  – You don’t have to learn new pages from scratch
• Increases learnability and thus usability
Contrast

• If two items are not the same, you can make them **Really Different**

Contrast

• Concord – typefaces drawn from the *same type family*
• Conflict – typefaces drawn from very similar type families
• Contrast – typefaces drawn from very different type families

• Avoid conflict: choose concord or contrast
• No more than two type families per screen (this slide has too many)
Designing for Visual Flow

Color

- Use color to reinforce, not as primary code
- ~10% of males (<1% females) have some form of color-blindness
- Keep in mind that color contrast affects readability

<table>
<thead>
<tr>
<th>Black on cream works well</th>
<th>White on black can too</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blue on cream is pretty safe</td>
<td>Red text can be painful</td>
</tr>
<tr>
<td>Colors opposite on the color wheel cause fatigues</td>
<td>Avoid similar colors</td>
</tr>
</tbody>
</table>
Visual Flow

Programs have a visual flow
Visual Flow

• This is especially important when designing for the web

• How do you determine the flow people draw from your design?
  – Observe!

If you want to learn more...

Designing Visual Interfaces: Communication Oriented Techniques
Kevin Mullet, Darrell Sano
People read web pages in an “F-Shaped” pattern
Implications of the F Pattern

• People won't read your text thoroughly
  – Word-by-word and exhaustive reading is rare.

• The first two paragraphs must state the most important information

• Start subheadings, paragraphs, and bullet points with information-carrying words

“Banner Blindness”

• Scanning is more common than reading
• People ignore things that look like ads
Implications of Banner Blindness

- Avoid putting important information in the header or side bars
- Assume that users will not see most of the fancy details you put at the top and sides
- Corollary: people consider pages that appear to have ads less reliable and authoritative

Representing Numbers

- Show numbers as numerals
  - Numerals catch the wandering eye
    - Numbers represent facts
    - Numbers look different than the surrounding text
      - 2415 looks different than two thousand fifteen in a block of text
- Numbers larger than a million are special
  - Represent one million as 1,000,000
  - Represent two trillion as 2 trillion, not 2,000,000,000,000
  - Generally, explain numbers over a billion
    - “1 trillion (or 1 million millions)”
Formatting

- Fancy and non-standard formatting is often counter-productive
  - Over-emphasis causes data to be perceived as decoration

Breadcrumbs

“Breadcrumbs” are a way to show system status
Breadcrumbs

- Term comes from Hansel and Gretel
  - More than just being able to backtrack, shows where the user is in the hierarchy

- Allow people to get to something else they saw

- Gives people an idea of how they got there

Next Time: ??