User Interface Design
User Interface Design

• Think of examples

• Good examples, personal favorites, smart UIs

• Poor examples, the ones you really hate to use

• and reasons for your verdict
This Lecture

• **Cover some basics of UI design**
  - This is the field of Human Computer Interaction

• **Won’t cover implementation technologies**
  - Java Swing, jQuery+CSS, YUI, Adobe Flex, etc

• **Won’t cover special interfaces such as cell phones, fax machines, cars, etc**
Principle

• UI design is more like film-making than bridge-building
  - About communication
  - Requires understanding audience
  - Requires specialized skills
  - Requires iteration
Principle cont.:

• Design for people
  - people's tasks, goals, and values drive development
  - work with users throughout the process
    • (rare, regular, extreme users)
  - assess decisions from the vantage point of users, their work, and their environment
  - pay attention to people's abilities and situation
  - talk to actual(!) experts
Observing Users: Need finding techniques

• "You can observe a lot just by watching" Yogi Berra

• 5 key things to learn with participant observation
  1 What do people do now?
  2 What values and goals do people have?
  3 How are these particular activities embedded in a larger ecology?
  4 Similarities and differences across people
  5 ... and other types of context, like time of day
User Interface Design

- After data modeling?
  - for information systems. - Yes
  - for video games. - No

- By a specialist?
  - for mass-market software. - Yes
  - for in-house systems. - No
Design alternatives

- Novice users
- Expert users
Design alternatives

- Novice users
  - Menus
  - Make it look like something else (analogy)
  - Simple
- Expert users
  - Commands
  - Specialize to make users efficient
  - Powerful
Human Factors

1. People have limited short-term memory (about 7 items of information)

2. Users make mistakes, especially under stress or under information overload

3. Users have differing capabilities and interaction preferences
UI Design Principles

• User Familiarity
  - UI should use terms/concepts drawn from the experience of users

• Consistency
  - Comparable operations activated in the same way

• Minimal Surprise

• Recoverability from errors

• User Guidance
  - UI should provide context-based help and meaningful feedback when errors occur
Examples

• Familiarity:
  - What objects should an air traffic control software provide to users?

• Consistency:
  - How many of you use emacs/vi as well as a different text editor?

• User Diversity:
  - Some users may be color-blind
Golden Rules

• Place the user in control
• Reduce the user’s memory load
• Be consistent
Place the User in Control

• No modes (i.e., different modes of operation: e.g., editing vs viewing)
  - Use a new window instead of a new mode
  - Make modes visible
• Undo
  - Users make errors, probably several in sequence
• Macros
  - For power users
• Hide technical details
User Interaction

• **Direct manipulation**
  - User interacts directly with objects on screen (e.g., with mouse or finger)
  - Fast and intuitive but requires visual metaphor

• **Menu selection**
  - Avoids user errors but slow for experienced users

• **Form fill-in**
  - Simple data entry but could be long

• **Command language (Unix)**
  - Powerful and flexible, high learning curve
Obtrusive Assistance

Microsoft Office Assistant “Clippy the Paperclip”

Further reading:
Nonobvious choices

<table>
<thead>
<tr>
<th>Effects</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Strikethrough</td>
<td>Shadow</td>
<td>Small caps</td>
</tr>
<tr>
<td>Double strikethrough</td>
<td>Outline</td>
<td>All caps</td>
</tr>
<tr>
<td>Superscript</td>
<td>Emboss</td>
<td>Hidden</td>
</tr>
<tr>
<td>Subscript</td>
<td>Engrave</td>
<td></td>
</tr>
</tbody>
</table>
Reduce Memory Load

- Reduce demand on short-term memory
- Establish meaningful defaults
- Define intuitive shortcuts
- Disclose information progressively
- Use real-world metaphors
- Speak user’s language
- Let user recognize, not remember
Common techniques

- Menus with keyboard shortcuts
- Dialog boxes
- Tabs
- Toolbar
Be Consistent

- Use visual interface standards
  - for operating system
  - for organization
  - for product or set of products
- Show context - keep user from getting lost
  - E.g., with “trail”
- System should explain itself
UI Patterns
Patterns are

- Things that repeat
- Solutions to common problems
- Tried and true solutions
- Expert knowledge
UI Patterns and Techniques

• Jenifer Tidwell: Designing Interfaces
• http://www.designinginterfaces.com/
• accessible online at SWEM:
• https://catalog.swem.wm.edu/Record/3309452
Two panel selector

From: Kathy E Gill <kathy@DOTPARAGON.COM>
Date: Wed June 18, 2003 2:34:43 PM Us/Eastern
To: Gill-WEB@ACM.ORG
Subject: (summary) Question: Field Designation: 'Information Architecture'
Reply-To: Kathy E Gill <kathy@DOTPARAGON.COM>

Regarding my request for information about the use of "Information Architect" as a title and the (apparent) heavy-handedness of the Examiner's Board in Ohio, I discovered that the topic has been discussed throughout our "community".

Here is my summary. Thanks to all for the replies.

SIGIA-L REFERENCES (5)

If you follow this link, you'll see an active discussion on this topic going on through the The American Society for Information Science and Technology (ASIST) special interest group in Information Architecture (SIG-IA), under "BANNED IN OHIO".

http://www.info-arch.org/lists/sigia-1/0306/
Two panel selector

• When to use?
  - You want the user to see the overall structure of the list, but you also want the user to walk through the items at her/his own pace, in an order of her/his choosing
  - Physically, the display is large enough to show two separate panels at once

• Why?
  - It reduces physical effort
  - It reduces visual cognitive load
  - It reduces the user’s memory burden
One-window drill-down
One-window drill-down

• When to use?
  - Many pages of content hyperlinked or hierarchically organized
  - You are building for a device with tight space restrictions
  - Even if you build for a desktop or laptop screen, you may have a complexity limit

• Why?
  - Web-browser metaphor
  - Keep it simple
Extras on demand
Extras on demand

• When to use?
  - There's too much stuff to be shown on the page, but some of it isn't very important.

• Why?
  - Simple UI is often better than a complex one, especially for new users, or users who don't need all the functionality you can provide.
Step by step instructions

1. What type of flight do you need?
   - Roundtrip
   - One way
   - Multiple destinations

2. Are your travel plans flexible?
   - I have specific travel dates
   - My travel dates are flexible (popular US routes only)

3. Where and when do you want to travel?
   - Leaving from:
     - Select from the list
   - Going to:
     - Select from the list

   - Departure month:
     - September 2004
   - Trip length:
     - 1 – 10 nights

4. Who is going on this trip?
   - Adults: (age 19 to 64)
   - Seniors: (age 65+)
   - Children: (age 0 to 18)
Step by step instructions

• When to use?
  - You are designing a UI for a task that is long or complicated, and that will be novel for the user
  - Those of you who design the UI knows more than the user does about how best to get the task done.
  - The user must be willing to surrender control over what happens when.
    • Making decisions is an unwelcome burden for people doing certain things: "Don't make me think, just tell me what to do next."

• Why?
  - By splitting up the task into a sequence of chunks, you effectively simplify the task
  - Could be frustrating to experts: know your users well
Progress Indicator

36% of ch04.doc Completed

Saving:
ch04.doc from tine-tripper.com

Estimated time left: 2 sec (1.10 MB of 3.20 MB copied)
Download to: P:\Jenfer_Tidwell\Patterns\...\ch04.doc
Transfer rate: 981 KB/Sec

✔ Close this dialog box when download completes

[Buttons: Open, Open Folder, Cancel]
Progress Indicator

• When to use?
  - A time-consuming operation interrupts the UI, or runs in the background, for longer than two seconds or so.

• Why:
  - Users get impatient when the UI just sits there.
  - Experiments show that if users see an indication that something is going on, they're much more patient, even if they have to wait longer than they would without a progress indicator.
Visual Framework
Visual Framework

• Design each page to use the same basic layout, colors, and stylistic elements, but give the design enough flexibility to handle varying page content

• When to use?
  - Look like one thing, deliberately designed; it should be easy to use and navigate

• Why?
  - users know where they are and where to find things
The Most Famous Example...
Clear Entry Points

• Google’s interface presents few, clear, task-oriented entry points
  
  - Useful for novice, infrequent users
  - “Instant gratification”
Center Stage
Center Stage

• Put the most important part of the UI into the largest subsection of the page or window; cluster secondary tools and content around it in smaller panels

• Why?
  - You should guide the user's eyes immediately to the start of the most important information (or task)
## Titled sections

<table>
<thead>
<tr>
<th>Products</th>
<th>Resources</th>
<th>Support</th>
<th>Purchase</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acrobat family</td>
<td>Government</td>
<td>Download Acrobat Reader</td>
<td>Adobe Store</td>
</tr>
<tr>
<td></td>
<td>Education</td>
<td>Other downloads</td>
<td>Volume licensing</td>
</tr>
<tr>
<td></td>
<td>Partners &amp; developers</td>
<td>Support home</td>
<td>Other ways to buy</td>
</tr>
<tr>
<td></td>
<td>Adobe Studio</td>
<td>Training</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Events &amp; seminars</td>
<td>Forums</td>
<td></td>
</tr>
<tr>
<td></td>
<td>All</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Titled sections

• Define separate sections of content by giving each one a visually strong title, and then laying them all out on the page together

• When to use?
  - There's a lot of content on the page, but you want to make the page easy to scan and understand
    • You can group the content into thematic or task-based sections that make sense to the user.

• Why?
  - It makes the information architecture obvious
  - The human visual system always looks for bigger patterns, whether they're deliberate or not
Card Stack

You can change which page to use for your home page.
Address: about:blank

Use Current  Use Default  Use Blank
Card Stack

• When to use?
  - There's too much material on the page
  - the user's attention becomes distracted

• Why?
  - The labeled "cards" structure the content into easily-digestible chunks, each of which is now understandable at a glance
Closable panels
Closable panels

• Put sections of content onto separate panels, and let the user open and close each of them separately from the others

• When to use?
  - There's too much stuff to present on the page, but you want it all only one click away
  - Titled sections, card stacks, extras on demand

• Why?
  - It can contain sections of wildly differing sizes.
  - The user can open several sections at once.
Movable panels
Movable panels

• When to use?
  - Allow users to configure
  - You may want users to feel some sense of ownership of the software, or at least have fun playing with it

• Why?
  - People like to rearrange their environment to suit their working style
Diagonal balance
Diagonal balance

• Arrange page elements in an asymmetric fashion, but balance it by putting visual weight into both the upper-left and lower-right corners

• Why?
  - contribute to a balanced composition on the screen
  - an ideal visual flow for users who speak left-to-right languages
Progressive Enabling
Progressive Enabling

• Starting with a UI that's mostly disabled, guide a user through a series of steps by enabling more of the UI as each step is done

• When to use?
  - You don't want to force the user to go page by page at each step
    • you'd like to keep the whole interface on one page

• Why?
  - The UI itself tells the possible consequences of some choices
    • if I turn this checkbox on, then I have to fill in these four text fields that just got enabled
  - The user can't do things that would get him/her into trouble
UI patterns

• Traditionally learned from experience
• Good to have standard names
• Good to have catalog to jog memory
• Still takes practice
Acknowledgements

• Many slides courtesy of Rupak Majumdar