CSE 403
UI Requirements & Design

Material in part from Marty Stepp and Valentine Razmov, past 403 classes.
Admin stuff

- Grade database should be working
- See the Admin section on the class wiki:
  - You should have accounts on cubist, unix groups are being set up
  - Your bugzilla is available
- Reading Summary4 is posted
  - 2 architecture readings
  - Due Wednesday in class
  - Bonus point if you turn it in on Monday in class
- Mailing list setup going ok?
- Questions on SRS due Tues?
Usability and software design

- **usability**: the effectiveness with which users can achieve tasks in one software environment
  - Studying and improving usability is part of Human-Computer Interaction (HCI).
Good UI design and usability

- Usability and good UI design are closely related.
- A bad UI can have unfortunate results...

### Usability and Good UI Design

**Although the Democrats**
are listed second in the column on the left,
you are the third hole on the ballot.

<table>
<thead>
<tr>
<th>REPUBLICAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>George W. Bush</td>
</tr>
<tr>
<td>Dick Cheney</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DEMOCRATIC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Al Gore</td>
</tr>
<tr>
<td>Joe Lieberman</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>LIBERTARIAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harry Browne</td>
</tr>
<tr>
<td>Art Olivier</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>GREEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ralph Nader</td>
</tr>
<tr>
<td>Winona LaBouke</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SOCIALIST WORKERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>James Harris</td>
</tr>
<tr>
<td>Margaret Trove</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NATURAL LAW</th>
</tr>
</thead>
<tbody>
<tr>
<td>John Hageden</td>
</tr>
<tr>
<td>Nat Goldhaber</td>
</tr>
</tbody>
</table>

**Punching the second hole casts a vote for the Reform Party.**

<table>
<thead>
<tr>
<th>REFORM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pat Buchanan</td>
</tr>
<tr>
<td>Ezola Foster</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SOCIALIST</th>
</tr>
</thead>
<tbody>
<tr>
<td>David McReynolds</td>
</tr>
<tr>
<td>Mary Cal Hollis</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CONSTITUTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Howard Phillips</td>
</tr>
<tr>
<td>J. Curtis Frazier</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>WORKERS WORLD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Monica Moorehead</td>
</tr>
<tr>
<td>Gloria Ll Riva</td>
</tr>
</tbody>
</table>

**Write-In Candidate**
To vote for a write-in candidate, follow the directions on the long stub of your ballot card.
Achieving usability

- Some methods to achieve good usability are:
  - User testing / field studies
  - Evaluations and reviews by UI experts
  - Card sorting
  - Prototyping
    - Paper prototyping, code prototyping

Good UI design focuses on the *user*, not on the developer or on the system environment
Paper prototyping

**Paper prototyping:** a means of usability testing where representative users perform tasks by interacting with a paper version of a user interface.

- **Why paper prototype?**
- **Why not just code up a working prototype?**
  - much faster to create than code
  - can change faster than code
  - more visual bandwidth (can see more at once)
  - more conducive to working in teams
  - can be done by non-technical people
  - no fear of cost of changes
Comparison of techniques

<table>
<thead>
<tr>
<th></th>
<th>paper prototype</th>
<th>UI builder</th>
<th>actual implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>ease of use</td>
<td>xx</td>
<td>x</td>
<td>-</td>
</tr>
<tr>
<td>fast turn-around</td>
<td>xx</td>
<td>x</td>
<td>-</td>
</tr>
<tr>
<td>flexibility, control</td>
<td>xx</td>
<td>x</td>
<td>-</td>
</tr>
<tr>
<td>executable</td>
<td>-</td>
<td>x</td>
<td>xx</td>
</tr>
<tr>
<td>team design</td>
<td>xx</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>
Where does P.P. fit in?

- At what point in the software lifecycle should we do (paper) prototyping? When would it be most useful to do it? Why?

- We talk about requirements being about "what" and design being about "how." Which is paper prototyping?
  - PP helps us uncover requirements and also upcoming design issues
  - do PP during or after requirements; before design
A P.P. usability session

- user is given tasks to perform using paper prototype
- session can be observed by people or camera
- one developer can "play computer"
Let’s move into UI design…”

(from Designing the User Interface, by Ben Schneiderman of UMD, noted HCI/UI design expert)
UI design

- When should we use:
  - A button?
  - A check box?
  - A radio button?
  - A text field?
  - A list?
  - A combo box?
  - A menu?
  - A dialog box?
  - Other..?
UI design - buttons, menus

- Use **buttons** for single independent actions that are relevant to the current screen.
  - Try to use button text with verb phrases such as "Save" or "Cancel", not generic: "OK", "Yes", "No"
  - use **Mnemonics or Accelerators (Ctrl-S)**

- Use **toolbars** for common actions.

- Use **menus** for infrequent actions that may be applicable to many or all screens.
  - Who likes menus?? Provide another way to access the same functionality (toolbar, hotkey, etc)
Checkboxes, radio buttons

- Use **check boxes** for on/off switches, when any one switch can be toggled irrespective of the others (often correspond to boolean values).

- Use **radio buttons** for related choices, when only one choice can be activated at a time (often corresponds to enum / constant values).
Lists, combo boxes

- use **text fields** (usually with a label) when the user may type in anything they want

- use **lists** when there are many fixed choices (too many for radio buttons to be practical) and you want *all* choices visible on screen at once

- use **combo boxes** when there are many fixed choices, but you don't want to take up screen real estate by showing them all at once
UI design - multiple screens

- use a **tabbed pane** when there are many screens that the user may want to switch between at any moment

- use **dialog boxes** or **option panes** to present temporary screens or options
An example UI

• What can we say about this GetInfo dialog? Did the designer choose the right components? Assume 30 properties.
How about this one

The Installation Wizard has detected an out of date version of Internet Explorer. Visual Studio comes with the latest version of Internet Explorer 4.01. This updated version of Internet Explorer 4.01 is an essential component of Visual Studio 6.0 Enterprise Edition and installation is required.

When you click on the Next button the Internet Explorer 4.01 installation program will start. Installation of Internet Explorer 4.01 requires you to reboot your computer.

Click Next to install Internet Explorer 4.01.
One more to analyze
Good UI: Apple Mac
Poor UI
Bad error messages

- **AK-Mail**: Do you really want to delete the selected folder?
  - Please enter 'YES' to start the operation.
  - Options: OK, Cancel

- **Eye Candy**: Are you sure you want to delete 'Ridges'?
  - Options: OK, NEe

- **Microsoft Access**: Wrong button!
  - This button doesn't work.
  - Solution: Try another.

- **Document Wizard Result**: Conversion complete!
  - Press View Result to view resulting documentation.
  - Options: View Result

- **www.wvfiremarshal.org - JavaScript Application**: Welcome to the West Virginia State Fire Marshal Online information center. This site is best viewed using Explorer or Navigator versions 4.0 or later and a display setting of 800x600.
  - Options: OK, Cancel

- **Dialog**: CuteFTP is currently working. If you press Disconnect, the session will be interrupted. Do you want to disconnect?
  - Options: OK, Help

  [Check box] Don't show this dialog again
UI Hall of Shame finalists

Let's learn from some more bad examples 😊

Interface Hall of Shame
http://homepage.mac.com/bradster/iarchitect/shame.htm

Web pages that suck
http://www.webpagesthatssuck.com/
Back to good UI design …

- **visibility**: Ability for user to find controls that are meant to be interacted with.
  - Where are they?
  - What is their state? ("Is this setting on or off?")

- **feedback**: Response from the control to the user before, during, or after an interaction.
Affordances in UI design

affordance: A physical property of an object that indicates or influences how it is to be used

- 3D buttons stand out and are more likely to be clicked
- Thick corner bars encourage resizing
Users don't read

- VS.

CSE 403, Spring 2007
Common web usability problems

http://www.useit.com/jakob/webusability/
http://www.useit.com/alertbox/9605.html

Layout

- Clutter
- Bad assumptions about user's screen resolution
- Requires horizontal scrolling
- Poorly chosen colors
- Frames
- Splash screens
- Poor / missing navigation controls (Back, Forward, Home)
- Text is not scannable (can't be read quickly)
- Doesn't follow standard design conventions
More web usability problems

- **Content**
  - Most important content isn't on the first page / screenful
  - Nondescriptive headings
  - Contains ads (or things that appear to be ads)
  - Important site content is contained in PDF documents
  - Isn't designed to be easily indexed by a search engine (HTML title, meta tags, page text, link text, etc.)

- **Links**
  - Links that don't say where they go
  - Badly chosen link text (such as "Click here for more info")
  - Links that forcibly open a new browser window
  - Links opened by complex Javascript needlessly
  - Visited links don't appear in a different color
More web usability problems

- **Features**
  - Poorly performing site search
  - Having a web search feature (why??)
  - Not having a site map or other means to navigate the site
  - Relying on non-standard plugins or browser versions (e.g. Overly reliant on Flash, Java applets, etc.)

- **Accessibility**
  - Text forced too small for elderly / visually impaired users
  - Lack of ALT text and non-image data for visually impaired users
  - Tiny links (hard to click for motor-impaired users)
Suggestions for good web design

- Place your *name and logo* on every page and make the logo a link to the home page.
- Provide *search* if the site has more than 100 pages.
- Write straightforward and *simple headlines* and page titles that clearly explain what the page is about.
- Structure the page to *facilitate scanning* and help users ignore large chunks of the page in a single glance.
- Instead of cramming everything about a product or topic into a single, infinite page, use *hypertext to structure the content* space into a starting page that provides an overview and several secondary pages that each focus on a specific topic.
- Use *link titles* to provide users with a preview of where each link will take them, before they have clicked on it.
Suggestions for web design

- Use *relevance-enhanced image reduction* when preparing small photos and images: instead of simply resizing the original image to a tiny and unreadable thumbnail, zoom in on the most relevant detail and use a combination of cropping and resizing.

- Ensure that all important pages are *accessible* for users with disabilities, especially blind users.

- *Do the same as everybody else*: if most big websites do something in a certain way, then follow along since users will expect things to work the same on your site.

- *Test your design with real users* as a reality check. People do things in odd and unexpected ways, so even the most carefully planned project will learn from usability testing.
UI exercise

- Get together with your group

  - Design a top level interface for *Viva la Taco*

    - Find locations [nearest you]
    - Find hours of operation
    - See menu
      - Option of vegetarian menu
    - Place an order for pickup