Today’s Lecture

- Welcome Back!
- Paper Prototyping
  - User Testing
- Final Exam

Reminders
- Blog Post (Week)
- Project 3
Final Exam

- Take Home or Two Hour Exam
  - Take Home
    - Programming – WPF / Surface
    - Written Answers
    - Much > 2 hours (4+ hours)
  - Normal Final
    - Two Hours
    - Open Book / Open Note
    - Some Programming but much less picky

CIFS: Don’t forget them!
Hall of Fame / Shame?
Why Prototype?

- Get feedback easier, cheaper
- Experiment with alternatives
- Easier to change or throw away

Was your final project alpha prototype worthy?

Low fidelity          High fidelity
Other Prototype Considerations

☐ Breadth vs. Depth
  ■ How many features to cover?
  ■ How well are they implemented?

☐ Examples
  ■ Word Processor
    ☐ Leave out printing, spell checking

☐ Horizontal vs. vertical
  ■ Horizontal – all breadth, little depth
  ■ Vertical – One area in depth
  ■ Horizontal – Usability (most common)
  ■ Vertical – Risky sub-area
Fidelity – User-interaction

☐ Look
  ■ Hand-drawn vs. toolkit widgets

☐ Feel
  ■ Mouse / keyboard
  ■ Point to “click”
  ■ Write input
  ☐ Tablet / Surface?
Paper Prototype

- Interactive paper mockup
  - Sketches of screen appearances
  - Paper pieces show windows, menus, dialogs

- Interaction is natural
  - Pointing = mouse
  - Writing = keyboard

- Person simulates computer operation
  - Putting down / picking up pieces
  - Writing responses
  - Describing what is going on
Why paper?

- Faster
  - Sketch vs. code

- Easier to change
  - Change between or even during
  - No code investment

- Focuses attention on big picture
  - Don’t waste time on details
  - Suggestions vs. nitpicking

- Non-programmers can help

- Clear model vs. view separation

Example: CAD Sketch
Tools

- Poster board
- Unlined index cards
- Restickable Glue
- Post-It Notes
- Correction Tape
- Photocopier
- Transparencies
- Pens, markers, scissors, tape
Tips

☐ Larger than life
☐ Monochrome
☐ Tricky visual feedback w/audio
  ■ Tool tip, drag and drop, progress bar
☐ Keep pieces organized
  ■ Envelopes, folders
Examples
Size Matters
Big / Dark Writing
Roles

☐ Computer
  ■ Design simulator
  ■ No feedback not from computer

☐ Facilitator
  ■ Present interface / tasks
  ■ Encourage user to think “aloud”
  ■ Keeps user on task

☐ Observer
  ■ Takes notes (lots of them)
  ■ Does not speak
What do you learn?

☐ Conceptual model
  ■ Do the users understand it?

☐ Functionality
  ■ Does it do what is needed?

☐ Navigation / task flow
  ■ Can users find their way around?

☐ Terminology
  ■ Do users understand the labels?

☐ Screen content
  ■ What needs to go on the screen?
What you can’t learn

- Look, color, feel
- Feel: Fitt’s Law
- Response time
- Are small changes noticed?
  - Everything is noted in a paper prototype
- Exploration vs. deliberation
  - Thrashing / exploration
  - Users do what you ask, not much tinkering with a paper prototype

Many studies show that low fidelity prototypes illustrate usability as well as high-fidelity prototypes.
Wizard of Oz

☐ Man behind the curtain
  ■ Simulate technology not available
    ☐ Voice recognition
    ☐ Artificial intelligence
  ■ Games
    ☐ Dynamic difficulty (human managers)
Users are human beings

- Human beings have been abused in the past
  - Yale electric shock study
  - MIT Fernald School study
  - Tuskegee syphilis study

Point

- Codes of conduct for human interactions
- Research related studies must be approved
  - ND – Institutional Review Board
User Pressure

- Performance anxiety
- Intelligence test
- Compare self w/others
- Feel stupid in front of observers
- Compete w/others
Respect your users

- Don’t waste time
- Make user comfortable
- Informed consent
- Privacy
- Control to the user
How many users?

- **Landauer-Nielsen Model**
  - Every tested user finds a fraction $L$ of usability problems ($L \approx 31\%$)
  - If user tests are independent, $n$ users will find a fraction $(1-(1-L)^n)$

- How many
  - 15 users to find 99% of one iteration
  - 5 users to find 85% on three iterations

- Rough rule of thumb
  - 3-5 users per user class
Flaws

☐ Finding of issues may be much less
  ■ Spool & Schroeder -> $L \sim 8\%$

☐ $L$ may vary from problem to problem
  ■ Interface differences
  ■ Task complexity
  ■ Individual differences

☐ Take home point
  ■ Difficult to predict how many users to work with
  ■ More is almost always better 😊
Split into groups

- Final project (5 minutes)
  - Sketch out with your group members
  - Sketch major screens / interfaces

**Wednesday:** Bring it to class, demo with other people in the class
Questions?

- Weekly Blog
- Project 3
- Alpha Submission