Questions?

• Homework #3
  – Due today

• Homework #4
  – Out today, due in 2 weeks

• Project #4
  – Out today due in 3 weeks

• Grad Project
  – Due in 3 weeks
    • Presentation in 2 weeks
Class survey

• Grading
  – Homeworks and projects now state points assigned to each question
  – “Learning by doing” implies that you have to ask questions
    • **Grading reflect how well you followed the process taught in class**

• Scheduling
  – Point well taken and will improve in the future
  – More homework and reading

• More participation from students (in class)
Usability heuristics

• “Rules of thumb” that describe features of usable systems
  – Can be used as design principles
  – Can be used to evaluate a design

• Pros and cons
  – Easy and inexpensive
    • *Performed by expert*
    • *No users required*
    • *Catch many design flaws*
  – More difficult than it seems
    • *Not a simple checklist*
    • *Cannot assess how well the interface will address user goals*
Usability Engineering

• Introduced by Nielsen (1994)
• Can be performed on working UI or sketches
• Required a small set (3-5) of evaluators to examine the UI
  – Check compliance with usability principles
    • Each evaluator works independently
    • Go through the interface several times
  – All reviews are aggregated in one final usability report
Nielsen's evaluation phases (1-2)

• Pre-evaluation training
  – Provide the evaluator with domain knowledge if needed

• Evaluation
  – First step: get a feel for flow and scope
  – Second step: focus on specific elements
    • *Multiple passes approach is better*
    • *Create a list of all problems*
Nielsen's evaluation phases (3-4)

• Severity rating
  – Performed after individual evaluations are aggregated
  – Establishes a ranking between problem
  – Reflects frequency, impact and persistence
    • Cosmetic, minor, major and catastrophic

• Debriefing
  – Discuss outcome with design team
  – Suggest potential solutions
  – Assess how hard things are to fix
Neilsen’s heuristics

- Simple and natural dialog
- Speak the users’ language
- Minimize user memory load
- Consistency
- Feedback
- Clearly marked exits
- Shortcuts
- Prevent errors
- Good error messages
- Provide help and documentation
Simple and natural dialog

• Present information in natural order
  – See class on graphic design

• Use windows frugally
  – Avoid complex window management

• Remove or hide irrelevant or rarely needed information
  – They compete with important information on screen
    • Pro: Palm Pilot
    • Against: Dynamic menus

• Use Occam’s razor
  – less to learn, to get wrong, to distract...
Speak the users’ language

- Use a language compatible with users’ conceptual model
  - Example: withdrawing money at an ATM

- Use meaningful mnemonics, icons and abbreviations
Minimize user memory load

- Promote recognition over recall
  - Recognition is easier than recall

- Describe expected input clearly
  - Don’t allow for incorrect input

- Create orthogonal command systems
  - Using generic commands that can be applied to all interface objects
Consistency

• Consistency promotes skills acquisition and/or transfer

• Be consistent in
  – Command design
    • Same action, same effect in equivalent situations
  – Graphic design
    • Input format
    • Output format
  – Flow design
    • Similar tasks are handled in similar ways
Feedback (I)

- Users should always be aware of what is going on
  - So that they can make informed decision
  - Be specific:

  ![Feedback: Toolbar, cursor, ink](image)

  - Provide redundant information
Feedback (II)

• Different feedback time scales
  – Shall I wait for that task to finished or go for coffee?
    
    .1s  Causality
    1s  Delay but user’s flow of thought is uninterrupted
    10s  Difficult to stay focused
    > 10s  User will switch to another task while waiting

• Different techniques
  – Short transaction: hour glass cursor
  – Longer transaction: estimate of time left
    • An overestimate is always better!