Computer/Human Interaction

Lecture 12

Overview:

- Requirements to Design
- Activity Design
  - Effectiveness
  - Comprehension
  - Satisfaction

Requirements to Design

- Problem Scenarios highlight requirements, set the scene
- Design transforms people's activities
  - New technology
  - New tasks
  - New experiences
  - Feedback into process

SBD Design Stages

- Activity Design – specify system functionality, i.e., backend of application
- Information Design – specify presentation of information, i.e., frontend UI of application
- Interaction Design – specify mechanisms for accessing and manipulation information

Activity Design

- Specify system functionality, i.e., backend
- Done first to determine what is possible
- Easier than information or interaction design
- Make progress quickly, focus on what system does rather than how it does it
- Cannot really analyze UI requirements until we know what the system does
- Main idea is to envision new activities

Activity Design Goals

- Design for effectiveness – meet real needs, more general than just efficiency
- Design for comprehension – users should be able to understand and predict system behavior
- Design for satisfaction – user should feel tasks are motivating and lead to feelings of accomplishment

Design for Effectiveness

- More than just efficiency/productivity. Real question: Is it the "right" solution?
- Innovation is good, but can be too much
  - Build on what is already working
  - Engage stakeholders in cooperative design
- Determine what parts of a task to support with technology
  - Leverage other aspects of work context, determine how task-related information is distributed (distributed cognition)

Design for Effectiveness

- Consider general solutions vs. needs of specific tasks.
- Example: on-line clothes shopping
  - Virtual models
  - Shopping lists
  - Arrange by type or by outfit

Design for Comprehension

- Want to know how users think about a task
- Cannot directly observe
  - Observe behavior, reaction, comments
  - Infer a *mental model* (vs. designer's model)
- Use *metaphors* that users understand to explore new ideas. E.g., typewriter -> computer, shopping carts

Design for Comprehension

- Leverage existing knowledge
  - Anticipate and support analogy
  - Look for ways to "break" current understanding.
- Example: computer is like a typewriter
  - Computer has keyboard with letters, digits, symbols, shift key; Enter key is like Return key
  - Computer also has insert mode and formatting

Design for Satisfaction

- Even if useful and comprehensible, users need to want to use a tool
- Automating tedious tasks is good, but may remove sources of reward or accomplishment
  - Workers apply personal expertise and knowledge to collect right information and make right decisions
- Needs of the individual vs. needs of the group
  - Work may not benefit the worker. E.g., collaboration requires individuals to check in/out documents, leave log records, etc.