CS 350: Computer/Human Interaction
Lecture 09 Overview

- Discuss Homework 2
- Refining activity design
- Coherence and completeness

CS 350: Computer/Human Interaction
Exercise 2, page 107

- Analyze the shopping cart used by your favorite on-line shopping system. In what ways does it mismatch the behavior of real-world shopping carts? Do these mismatches help or hurt the activities of selecting and ordering products?

CS 350: Computer/Human Interaction
Exercise 3, page 107

- One effect of templates on exhibit construction is that the first steps are automated – standard components are predefined and serve as a guide for the student importing his or her own content. Use Tradeoff 3.6 to critique this proposal. To what extent does a student exhibitor engage in “knowledge work”? How does the template affect this?

CS 350: Computer/Human Interaction
Tradeoff 3.6

- Tradeoff 3.6: Automating tedious and error-prone steps improves job satisfaction, BUT automation of some activities may undermine motivation and self-esteem.

CS 350: Computer/Human Interaction
Refining Activity Design

- Recall: activity design is about envisioning functionality
- Iterative process – design ideas emerge, are tried out, elaborated, and refined
- Common techniques for refinement
  - System object’s point of view (POV)
  - Participatory design

CS 350: Computer/Human Interaction
System Object’s Point of View

- Consider computational aspects implied by scenario, responsibility-driven design similar to object-oriented design
- Anthropomorphism: object as a sentient entity. “How do I get created? How do I behave? What could I do to be useful?”
- May prompt ideas of when software should take initiative or be responsive to user actions
CS 350: Computer/Human Interaction

Participatory Design

- Revisit individuals or groups from field studies
- Present activity scenarios to get feedback
- Use claims analysis and/or object POV

Coherence

- **Coherence**: no contradictions or inconsistencies over the whole set of scenarios
  - One team of designers
  - Reuse actors and task information
  - Insert same design features in multiple scenarios
  - Formal analysis, e.g. matrix of objects and tasks

Completeness

- **Completeness**: considered all needs
  - Generally, cannot enumerate all possible activities
  - Main issue is missed opportunities
  - Consider new activities not previously done