Lecture 2

- Homework 1 posted, due on Thursday at beginning of class
Outline

- Chapter 2 - The Wheel: A Lifecycle Template
The Wheel

- Iterative, evaluation-centered, UX lifecycle template

- **Iteration**: All or part repeated for purpose of exploring, fixing, or refining design

- **Lifecycle** is a structured framework consisting of a series of stages and corresponding activities

- **Template** is instantiated for each project to create a *process*.
  - Based on project resources, goals
The Wheel

• Why is a process important?
• Process acts as scaffolding to ensure project is on track and helps novices become experts
• Engineering process converts usability from a "last minute add-on" to an integral part of product development
Four basic abstract activities

- Design
- Analyze
- Evaluate
- Implement
How these fit into the Wheel

[Diagram showing the cycle of design, prototype, analyze, and evaluate with iterative processes indicated at each step.]
The process activities for UX

• Analyze
  – Understanding work/play domain, user work, and user needs
    • Chapter 3: Contextual inquiry
    • Chapter 4: Contextual analysis
    • Chapter 5: Extracting requirements
    • Chapter 6: Synthesizing design-informing models
The process activities for UX

• Design
  – Creating conceptual design, interaction behavior, and look and feel
    • Chapter 7: Design thinking, ideation, and sketching
    • Chapter 8: Mental models and conceptual design
    • Chapter 9: Design production
The process activities for UX

• Prototype
  – Realizing design alternatives
    • Chapter 11
  – Often done in parallel with design
The process activities for UX

- Evaluate
  - UX goals, metrics, and targets: Chapter 10
  - Verifying and refining interaction design
    - Fully rigorous methods: Chapters 12, 14-17
    - Rapid evaluation methods: Chapter 13
Flow among activities for UX

- Activities can iterate, overlap
- Process managed with activity transition criteria
- Main goal: Move forward to production
Managing progress within lifecycle

- Team must be able to decide:
  - When to leave an activity
  - Where to go after any given activity
  - When to revisit a previous process activity
  - When to stop making transitions and proceed to production

- Answers depend on transition criterion at end of each process activity
- Based on whether designers have met goals and objectives for current iteration
Managing progress within lifecycle

• Resources limits
  – Especially time and budget
  – Can trump other criteria for stopping process
Project parameters influence choice of process

• Tolerance for risk
  – Of things going wrong
  – Of features or requirements being missing
  – Of not meeting needs of users

• The less tolerance for risks—the more need for rigor and completeness in process
Project parameters influence choice of process

- Project goals: goal-oriented process choices
- Project resources: budget, schedule, person power, skills (people with extensive experience and maturity need less rigorous process)
Project parameters influence choice of process

- Type of system being designed (for example, mp3 player vs. air traffic control system)
- Development organizational culture (organizational history, traditions, market position, urgency to market)
- Stage of progress within project
Mapping project parameters to process choices
The system complexity space
Interaction complexity

- About intricacy or elaborateness of user actions
- Includes cognitive density
- Difficulty of tasks with system
  - Low interaction complexity - smaller, easier tasks (example, ordering flowers from a Website)
  - High interaction complexity is – larger, more difficult tasks, often requiring special skills or training (example, manipulating a color image with Adobe Photoshop)
Work domain complexity

- About intricacy and technical nature of corresponding field of work
- High work domain complexity
  - Convoluted and elaborate work flow mechanisms
  - Collaborative work flow
  - Dependencies and constraints
  - Example, geological fault analysis for earthquake prediction
Work domain complexity

• Low work domain complexity
  – Way system works within setting relatively simple
  – Example, Website for buying flowers
Influence of system type on process choice

- Complex interaction, complex work domain → strong requirement for rigorous process
  - Example, air traffic control system, air traffic controller deciding landing orders for incoming airliners
Air traffic control domain

- Has complex interaction
- Complex work domain
- Work flow and collaboration among large number of work roles and user types
- Extreme focus on error and risk avoidance
- Emphasis on compliance to rules
Influence of system type on process choice

• Simple interaction, complex work domain
  – User tasks relatively simple and easy to understand
  – But domain complexity calls for more attention to contextual inquiry and analysis, modeling, and requirements
  – Need insight into internal system complexity and complex rules and compliance requirements
  – Example, tax preparation software
Influence of system type on process choice

- Simple interaction, simple work domain
  - Smaller Websites, certain interactive applications, some commercial products
  - Can be need for focus on emotional impact factors such as aesthetics, fun, joy of use
  - Example, designing a new mp3 music player
Influence of system type on process choice

• Complex interaction, simple work domain
  – Emphasis on design, ideation, and sketching, plus evaluation within real usage
  – Example, designing a digital watch
  – Attention needed for interaction design: task interaction structure, screen layouts, user actions, metaphors
User interface team

Create interaction design concepts

Design

Interaction designer

Visual designer

Understand user work and needs

Analyze

User Researcher

UX manager

Verify and refine interaction design

Evaluate

Technical writer

Usability analyst

Realize design alternatives

Prototype

Interactive prototype programmer
User interface team

- Project manager
- Lead UX designer
- User researcher
  - For contextual inquiry and other work domain analysis activities
- Users, user representatives, customers, and subject matter experts
User interface team

• Interaction designer
  - For ideation and sketching, conceptual and detailed design, and low-fidelity prototyping activities

• UX analyst or evaluator
  - For planning and performing UX evaluations, analyzing UX problems, and suggesting redesign solutions
User interface team

• Visual/graphic designer
  - For designing look and feel and branding and helping interaction designers with visual aspects of designs

• Technical writer
  - For documentation, help system design, and language aspects of interaction designs
User interface team

• UI programmer
  – For programming interactive high-fidelity UX design prototypes
  – To implement interaction designs in system software

• UX manager
  – Overall responsibility for UX process
Iteration

• Iteration is good, but not enough
  – Iteration can help you converge locally
  – Cannot just start with any old design and iterate yourself to quality user experience

• Start iteration early
  – Design, not implementation
UX iteration scope is limited

- UX iteration not the same as multiple passes through whole development
- Limited to small, early, lightweight, inexpensive part of overall lifecycle process

Diagram:
- Inquiry into work domain, requirements, modeling
- Interaction design requirements, UX goals
- Interaction design to support work
- Constraints & problems
- Revisit work domain if necessary
- Feedback about UX problems
- Prototype requirements
- Prototype evaluation
- Rapid prototyping
Overview of analysis activities