Week 2: Usability Engineering Lifecycle

CS285 Usability Engineering
Tasks Analysis

• Gather background information about users & their tasks
• Collect and analyze data from observations of and interviews with users as they do real work in their actual work environments
• Construct and validate a model of the users’ current task organization
User Task Organization Model

A possible configuration

Customer Support Representative

Sales Support

• New Sale
  • Policy Type A
  • Policy Type B

Change Requests

• Beneficiary
• Renewal Date
• Address
• Phone
• Claims
• Change Application

Information Requests

• Change Quote

Customer Support

Follow-up

Office Administration

• Supplies
• Facility
• Directory
• Telephone Bill
• Backup
• Postage
• Bank Deposits

Change Requests

• Address
• Phone

Information Requests

Policy Management

New Sale

• Incomplete Application
• Missing Evidence
User #1

Customer Support Representative

- Customer Support
  - Simple Changes
    - Customer
      - Address
      - Phone
  - Claims
    - Policy
      - Beneficiary
      - Renewal Date

- New Sale
  - New Application
  - Incomplete Application
  - Missing Evidence

- Sales Support
  - Policy Change
    - Change Application
    - Change Quote

- Office Administration
  - Daily
    - Telephone Bill
    - Backup
    - Postage
    - Bank Deposits
  - Occasional
    - Supplies
    - Facility
    - Directory
Platform Capabilities/Contraints

• Identify the user interface-related capabilities and constraints of hardware & software platforms (+ development tools)
• Review any platform documentation
• Interview technical staff
• Document platform capabilities & contraints
• Validate documented platform capabilities & contraints
General Design Principles

• Identify relevant general interface design principles and guidelines
• Use style guides for the look and feel of products to run on a specific platform:
  – Macintosh Human Interface Guidelines
  – Common User Access (CUA) (IBM)
  – Open Software Foundation/Motif Style Guide
  – The Windows Interface Guidelines for Software Design
General Design Principles

• International Standards
  – ISO 13407 Human Centred Design Processes for Interactive Systems
  – ISO 14915 Design of the User Interface of Multimedia Applications
  – ISO9241 Ergonomic Requirements for Office Work with Visual Display Terminals (VDTs)

• Industry Standards
  – Web Services for Remote Portlets (WSRP) specification and JSR 168/268 standardized Java portlet API
HCI Design Recommendations

• Physical Factors
  – Avoid displaying saturated red text on a saturated blue background (or vice versa) --> “chromostereopsis”
  – Use motion only for getting and keeping the user’s attention
  – Relevant characteristics vary less from person to person than cognitive/affective factors
“chromostereopsis”
Universal Accessibility

• Audiences with special needs /varying capabilities
  – Physically impairments (visual, hearing, motor, cognitive, selective disturbance)
  – The inexperienced or technophobic
  – The socially excluded
  – Older users
  – Non-English users
Guidelines for Inclusiveness

• Disability Rights Commission, DRC (http://www.drc.org.uk/accessibility.aspx)
• IBM Accessibility Centre (http://www.ibm.com/able)
• Web Accessibility Initiative (WAI) by W3C (http://www.w3.org/WAI/Resources)
• USA Federal Regulations (www.access-board.gov/508.htm)
HCI Design Recommendations

• Cognitive factors (factors in how people process information)
  – When listing options for user selection (e.g., in a menu or list box), present them in an order that makes sense to the user’s task, grouping them if there are more than just a few. If there is no logical order, list them alphabetically
  – Provide keyboard “shortcuts” or “accelerators” for commonly used functions and menu items
HCI Design Recommendations

• Affective Factors (emotions, values, preferences, satisfaction)
  – Design to put users in control of the interaction
  – Provide for some user customization of the æsthetics of the interface
How to use standards?

• Select which of the available standards you are going to use as your starting point
• Develop a project standard by tailoring the standards you selected to use --> Identify the specific recommendations that apply to your project and determine how you are going to apply them.
• Apply the recommendations -->Refer to them when making Interaction Design decisions
• Revise and refine your project standard as necessary to accommodate new information and considerations that may arise during product development
2 - Design/Testing/Development

• Apply design principles, style guides, usability specifications, contextual user task organisation models, conceptual models
• Prototyping
• Evaluation with users – Formative Evaluation
• Accept / Re-design and re-evaluate with users
Design

• Re-engineer Current User Task Organization Model (consider Task Scenarios and Use Cases)
• Using Conceptual Models
• Ensuring that the user's tasks are achieved
• Facilitating the attainment of user's goals
• Within social and technical constraints and limitations
Task Scenarios

+ 

Use Cases

+ 

Users Task Organization Model

→ 

Re-engineered Users Task Organization Model & Task Sequence Model

- "User Interface Architecture"

  - Delineate underlying task organizational and navigational structure

- Conceptual Model Design

  - Set of conventions regarding the presentation of that structure
Conceptual Model Design

• High level presentation rules that map onto Task Organization Models

• Make decisions on
  – Product or process oriented conceptual models
  – Product (documents, tools) or processes to support
  – A set of presentation rules
  – How major information and functionality will be divided across displays
  – Defining major navigational pathways

• Document alternative conceptual design models
Developing Conceptual Models

- Which interaction mode would best support user activities?
- Is there a suitable interface metaphor to help users understand the product?
- Which interaction paradigm will the product follow?
Interaction Mode

• Conceptual Model based on Activities
• Types of activities
  – Instructing
  – Conversing
  – Manipulating and navigating
  – Exploring and browsing
  – E.g., Computer game, drawing package, applications to support some (work) process – “process-oriented”
Interaction Mode

• Conceptual Models based on Objects
  – Perspectives are structured around real-world objects
  – “Product-oriented” – there are clear, identifiable work products that users individually create, modify and maintain
  – Choice of objects as a basis is related to the choice of interface metaphor
Interface Metaphor

- Use of familiar knowledge
- Combined with new knowledge to help users understand the system
- Only a partial mapping between application and the real thing (metaphor)
- Look for metaphors in users’ tasks descriptions
- Evaluate the choice of suitable metaphor
What makes a good metaphor?

• Provide structure, familiar structure
• High degree of relevance, match expectations
• Easy to represent with visual and audio elements, including words
• Users understand
• Extensible, any extra useful aspects
Interaction Paradigms

• Design philosophies that help design exploration
  – Desktop paradigm, WIMP (windows, icons, menus, & pointers)
  – Ubiquitous computing
  – Pervasive computing
  – Wearable computing
  – Augmented reality
  – Affective computing
Prototypes

• Aid in discussing ideas with stakeholders
• Serve as communication devices amongst design team members
• Support evaluation of Conceptual Models
• Compare among alternatives
• Are user requirements met?
• Are user tasks well supported?
Low-fidelity prototypes

• Does not look very much like the final
• Uses paper (& cardboard – Index Cards)
• Simple, cheap, and quick
• Flexible and encourage exploration and modification
• Meant for design exploration only (e.g., issues of content, structure, screen layouts)
Low-fidelity prototypes

- **Storyboards**
  - Sketches showing how user might make progress through a task with a design
  - Stepping through a scenario
- **Sketching**
  - Using simple boxes, stick figures, arrows, explanatory texts/words, etc.
- **Wizard-of-Oz**
High-fidelity prototypes

• Looks and feels much like the final product
• Simple prototyping tools: Macromedia Director, Visual Basic, etc.
• Powerful tools: full-fledged development environments
Building prototypes

- Choose/select range and depth of functionality to assess
- Horizontal prototyping (wide range of function but with little depth)
- Vertical prototyping (providing a lot of detail but only for a few functions)
- Evolutionary prototyping (evolving into a final product)
- Throwaway prototyping (final product is built from scratch)
Iterative CM Evaluation

• For each iteration
  – Plan the test and develop supporting materials
  – Run test users and collect data according to test plan
  – Analyze and interpret the data to formulate redesign recommendations

• At the end of each test
  – Modify the Conceptual Model design(s)
  – Modify the Conceptual Model prototype(s)
  – Modify test plan and materials
Iterative Evaluation

• Planning and preparing
  – Decide on usability goals to test
  – Identify type and range of users
  – Identify the tasks
  – Design test tasks
  – Design test and develop test materials
  – Design and assemble test environment
  – Recruit/schedule pilot test users
  – Run pilot test
  – Revise test procedure and materials
  – Recruit/schedule test test users
Iterative Evaluation

• Conducting the tests
  – Run test and collect data
  – Summarize data
  – Analyze and interpret data
  – Draw conclusions and formulate recommended design changes
  – Document and present results