Design Guide

Adapted from guide originally created by Prof. Jaime Ruiz
Design Space

• Design space is both conceptual and a real tool
• Conceptually, refers to the full range of possibilities for addressing identified problem
  – Infinite in scope
• Realistically, refers to a mapping of one or two dimensions of the design
• Helps to organize and suggest possibilities
• Example: Digital vs. physical, manual vs. automatic, input modalities, output modalities
Making Design Concrete

• Design process requires lots of cognitive resources
  – Lots of information to consider
  – Lots of information unknown
  – Lots of possible solutions to consider
  – Lots of possible details to consider for final solution form

• Make *tangible representations* of every design idea to cope with this complexity
  – Write everything down!
  – Sketch!
  – Create example prototypes!
Benefits of Tangible Representations

• Offload cognition into environment
• Reflect on and evaluate the design
• Gather feedback from stakeholders
• Iterate the design
• Communicate the design to others
• Compare it to other design ideas
• Find holes in one’s understanding of the possible solution
• Discover unintended side effects (positive/negative) of the design idea
Design Tools

- Low-fidelity prototypes (paper-based)
- Wizard-of-Oz prototypes (paper, cardboard)
- High-fidelity prototypes (functional mock-ups)
- Scenarios
- Storyboards
Why prototype?

• Faster, cheaper than building functional system
  – One bug in implementation can wreck evaluation

• Evaluation
  – Roughness of system encourages “right” level of evaluation

• Communication
  – More clearly communicate intents
Types of Prototypes

Horizontal vs. Vertical:
- Horizontal
  - Broad overview of system at cost of great detail
- Vertical
  - Detailed rendition of one small part of system

Low Fidelity vs High Fidelity
- Low fidelity
  - Extremely coarse approximation of system using common physical materials
  - Intent is to get primary ideas across
- High fidelity
  - Systems nearly identical to proposed production version
Representational Determinism

Zhang (1997)

• Notion that external representations, tools guide and constrain cognitive behavior
  – If all you know is VB, then all of your solutions will be VB-like

• Low fidelity prototypes don’t presume any UI conventions
  – Wide design palette
  – Helps avoid design fixation
  – But also requires you to be more creative
Low Fidelity Prototyping

• Use commonly available physical materials to mock up designs
  – Paper, cardboard, cereal boxes, Post-Its
  – Wood
  – Foam
  – Any other random piece of junk that helps you realize the vision for your system

• Can iterate on-the-fly
  – Sketching information visualizations

• Broader design palette than computer-based tools offer
Paper Prototyping

- Start with blank sheets of paper
- As design starts to solidify, sketch unvarying system states on piece of paper / cardboard
- Use Post-Its to represent dynamic content
  - Drop-down menus
  - Views (e.g., graphs) that change
  - Drag-n-drop items
- Can also use tracing paper, transparencies for dynamic content
Paper Prototyping Tips

• When in doubt, make a mark
  – Can always erase or redraw
  – Takes more time to debate putting something on the paper than actually doing it

• If you think your idea is wrong, still make the mark
  – Prove to yourself the idea is wrong
  – Reflect on why it is wrong once it’s on paper

• Goal is to continually move forward
  – Nothing worse than a blank canvas
Wizard-of-Oz

• Low-fidelity prototypes can be brought to life
• Person fills in for computation
• Have different system states already built
  – Or build on-the-fly based on user demands
Prototyping Sessions

• Master physical media and physical space to support design
  – Use walls to post work models, brainstorming ideas, design ideas
• Walls become “external memory store”
  – Support communication to people within, outside project team
• Covering wall with your models, design ideas also encourages and boosts morale
  – Shows what work has gotten done
Designing the Interaction

Goals:

• To engineer the interaction with the system from the user’s perspective

• Tools
  – Scenarios and storyboards
  – Mental models of interaction
Scenarios
• Plain language description of interaction with software
• Includes
  – Goals, Expectations, Actions, and Relations
• Allows user to understand how software will be used
• Need appropriate level of detail for stage in design
  – During early stage evaluation:
    • Are goals and action/reactions reasonable (grounded in reality)
  – During late stage evaluation:
    • A set of tasks that users can perform with software
• Can be conveyed in many ways
  – Text only, text with example screenshots
  – Storyboards, “acted out”, movie, etc.
• Create alternatives to explore the design space
• Consider a handheld inventory management system
• Consider errors, worst-case scenarios, etc.
Scenario example – text only

Tom presses the on button on his smartphone and is presented with a screen where he can select his username and input his password. After logging in, he is presented with an alphabetically sorted list of application icons. He clicks on “Inventory Management.” Pointing the smartphone camera at the box on the shelf, he presses the “scan” button on the screen to scan the QR-code on the side of the box into the system.
Scenario example - storyboards

- Makes scenario real
- Shows people, words, screenshots, whatever is appropriate