Design Thinking & Prototyping

SWE 432, Fall 2016
Design and Implementation of Software for the Web
Today

- How do we set ourselves up to build good interfaces from the start?
- What is the iterative process by which we start out with a lot of ideas, and end up with some good, end result interface?

For further reading:
http://interchangeproject.org/2013/11/02/paper-prototyping/
Why sketch?

• Design is process of creation & **exploration**
• Sketching offers **visual** medium for exploration, offering cognitive scaffolding to externalize cognition
• Sketches let us explore many alternative designs
Why alternatives?

- Important to think broadly about a wide range of possible designs
  - What are the different ways in which user might do $x$?
- Rather than reimplement the status quo, alternatives offer options for doing things differently, enabling analysis of which is best
  - Important to challenge preconceptions and think deeper
- Rather than develop a single idea, sketching enables exploration and consideration of multiple designs, allowing examination of pros and cons
- Expert designers often create *many* alternatives
  - 10, 50, 100 alternative designs
Sketching vs. Prototyping

Sketching:
- Buxton Design Exploration Sketches
  - For design
  - Getting the right design
  - Experimenting, exploring, being creative
  - Goal: Support ideation to find a great design solution

Prototyping:
- Low-Fidelity Design Refinement Prototypes
  - For UX engineering
  - Getting the design right
  - Following the UX process
  - Goal: Support iterative refinement of a given design
Physical sketches

- Production tools for sketching:
  - whiteboards, blackboards, cork boards, flip chart easels
  - post it notes
  - duct tape, scotch tape, push pins, staples
  - marking pens, crayons, spray paint
  - scissors, hobby knives, foam core board
  - duct tape
  - bits of cloth, rubber
Sketches are Sketchy

• Not mechanically correct and perfectly straight lines
• **Freehand**, open gestures
• Strokes may miss connections
• Resolution & detail **low** enough to suggest is concept
• Deliberately **ambiguous** & abstract, leaving “holes” for imagination
Benefits of Sketching

• No “programming” needed! Fast turnaround
  • Costs less
  • Allows more iterations
• Human computer
  • Can be (re)programmed quickly
  • Cannot crash
  • Changes can be made on the fly
• Developers feel less affection for status quo because changes are easy
• Rough “sketchy” appearance
  • Emphasizes content instead of appearance
  • Avoids low-level critiques of visual detail
  • Users are more willing to criticize high-level problems and less willing to blame themselves if something doesn’t work
Rules for sketching

- Everyone can sketch; you do not have to be artistic.
- Most ideas conveyed more effectively with sketch than words.
- Sketches are quick and inexpensive to create; do not inhibit early exploration.
- Sketches are disposable; no investment in sketch itself.
- Sketches are timely; made in-the-moment, just-in-time.
- Sketches are plentiful; entertain large # of ideas w/multiple sketches of each.
Sketches include annotations

- Annotations explain what is going on in each part of sketch & how
Sketches support design exploration
Fidelity of sketches & mockups

storyboard

wireframe

prototype

low

(many details left unspecified)

definition

fidelity

high

(more polished & detailed)
Sketching Example:
News Viewer
Article Layout through movable windows (DADA) - drag and drop articles

- Movable windows
- Clickable
- Layered by importance
News Timeline

- 2am in & out
- Articles sorted by time
- Could have just pictures
**UI/UX Wireframe**

<table>
<thead>
<tr>
<th>FLEXIBLE News Paper</th>
<th>Sports</th>
<th>Tech</th>
<th>Ecstasy</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Even boxes?
- Different size boxes with same format every time?
Conceptual design
Designer’s mental model

• Conceptualization of the envisioned system
  • what it is
  • how it is organized
  • what it does
  • how it works
User’s mental model

- Comes up from existing interactions with systems
- Users form cause & effect relationships to form theories that guide actions
Conceptual design

• Goal: match users’ mental model
• Tool: Metaphor - analogies from existing system
  • Offers expectations about what system does & what can be done
• Examples
  • Email <—-> physical mail
  • Backup software <—> time machine
  • OS desktop <—> top of a desk
Storyboards
Storyboard for Disney’s Melody: Adventures in Music (1953)
Source: Michael Sporn Animation
Now students
listen close
to me -

Cause today
we're gonna talk
about Mel O'dee

Now all the
professors and
cats have found
Storyboards for UI design

• Sequence of visual “frames” illustrating interplay between user & envisioned system
• Explains how app fits into a larger context through a single scenario / story
• Bring design to life in graphical clips - freeze frame sketches of user interactions
• “Comic-book” style illustration of a scenario, with actors, screens, interaction, & dialog
Crafting a storyboard

• Set the stage:
  • Who? What Where? Why? When?
• Show key interactions with application
• Show consequences of taking actions
• May also think about errors
Example elements of a UI storyboard

• Hand-sketched pictures annotated with a few words
• Sketch of user activity before or after interacting w/ system
• Sketches of devices & screens
• Connections with system (e.g., database connection)
• Physical user actions
• Cognitive user action in “thought balloons”
Example: ticket kiosk

Ticket buyer walks up to the kiosk

Displays “Occupied” sign on wraparound case

Sensor detects user & starts immersive process

Detects people with ID card
Example: ticket kiosk

Greets buyer and asks for PIN

Buyer selects “Boston symphony at Burruss Hall”

Shows recommendations & most popular categories

Plays music from symphony, shows date & time picker
Frame transitions

• Transitions between frames particularly important
• What users think, how users choose actions
• Many problems can occur here (e.g., gulfs of execution & evaluation)
• Useful to think about how these work, can add thought bubbles to describe
Storyboarding Fail

1: Auto-login to Piazza app

2: Network error

3: Asked for password
Wireframes
Wireframes

- Lines & outlines ("wireframes") of boxes & other shapes
- Capturing emerging interaction designs
- Schematic designs to define screen content & visual flow
- Illustrate approximate visual layout, behavior, transitions emerging from task flows
- Deliberate unfinished: do not contain finished graphics, colors, or fonts
### Example

<table>
<thead>
<tr>
<th>Photo collections (primary navigation)</th>
<th>Main content display pane (photos from selected collection, meta data of photos, views)</th>
<th>Related and contextual information pane</th>
</tr>
</thead>
</table>

Web page banner

Information concerning design patterns. Other issues, such as design, and so on.

While on the topic of increasing physical efficiency, let’s consider the effective improvement of the performance of the system. The key to achieving this is through optimizing the interaction between the components of the system.

In order to optimize the interaction, we need to analyze the performance. For instance, if the performance of the system is not satisfactory, we need to identify the issues and optimize accordingly.

The system’s efficiency can be enhanced by improving the interaction between its components. This can be done by optimizing the interaction and ensuring that the system performs efficiently.

In conclusion, optimizing the interaction between the components of the system is crucial for improving its performance and enhancing efficiency. By focusing on optimizing the interaction, we can ensure that the system operates efficiently and effectively.

LaToza/Bell

GMU SWE 432 Fall 2016

33
Example
Wireframes

• Can be used to step through a particular scenario
• Focus on key screens rather than every screen
• Tools can help
  • Can be made clickable
  • Can use stencils & templates; copy & edit similar screens
Example tool - Balsamiq
Prototyping
Prototyping

• How do you know your system design is right before you invest the time to build it?
• Answer: prototyping!
  • Evaluation performed **before** investing resources in building finished product
  • Early version of system constructed much **faster** & with less expense used to evaluate & **refine** design ideas
# Fidelity of prototypes

<table>
<thead>
<tr>
<th>Kind of Iteration</th>
<th>Purpose</th>
<th>Types of Prototypes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ideation and</td>
<td>To support exploring ideas, brainstorming, and discussion (so design</td>
<td>Sketches, fast and disposable mockups, ultralow fidelity</td>
</tr>
<tr>
<td>sketching</td>
<td>details are inappropriate)</td>
<td>Evolution from hand-drawn paper, computer-printed paper, low-fidelity wireframes,</td>
</tr>
<tr>
<td>Conceptual design</td>
<td>To support exploration and creation of conceptual design, the high-level</td>
<td>high-fidelity wireframes, to pixel-perfect interactive mockups (to communicate</td>
</tr>
<tr>
<td></td>
<td>system structure, and the overall interaction metaphor</td>
<td>with customer)</td>
</tr>
<tr>
<td>Intermediate design</td>
<td>To support interaction design for tasks and task threads</td>
<td>Evolution from paper to wireframes</td>
</tr>
<tr>
<td>Detailed design</td>
<td>Support for deciding navigation details, screen design and layout,</td>
<td>Detailed wireframes and/or pixel-perfect interactive mockups</td>
</tr>
<tr>
<td></td>
<td>including pixel-perfect visual comps complete specification for look</td>
<td></td>
</tr>
<tr>
<td></td>
<td>and feel of the “skin”</td>
<td></td>
</tr>
<tr>
<td>Design refinement</td>
<td>To support evaluation to refine a chosen design by finding and</td>
<td>Medium to high fidelity, lots of design detail, possibly a programmed prototype</td>
</tr>
</tbody>
</table>
Interactivity of prototypes

- Scripted, click through prototypes
  - Prototype w/ **clickable** links to move between screens
- Live action storyboard of screens
- Simulates real **task flow**, but w/ static content
- Fully-implemented prototypes
  - Usually **expensive** to implement actual system
  - But can build key piece of system first to evaluate
Wizard of Oz

• Goal: **simulate** actual system w/o building it
  • Want user to interact **as if** they were interacting w/ real system
  • Helps explore how users would interact w/ novel interaction if it were to exist
• Example: natural command line (Good et al 1984)
  • Users typed in commands to interact w/ computer
  • Commands intercepted by hidden human who interpreted commands & executed them
Paper prototypes

- **Low fidelity** prototype w/ paper mockups
- Goal: get feedback from users early w/ very low cost interactive prototype of envisioned interaction design
Paper prototyping (1)

- Set a realistic deadline
- Gather set of paper prototyping materials
- Work **fast** & do not color within the lines
- Reuse existing sketches & mockups
- Make underlying paper mockups of key screens
Paper prototyping (2)

- Use paper cutouts & tape onto full-size transparencies as “interaction sheets” for moving parts, making modular by including only a small amount
- Do not write or mark on interaction sheets
- Be creative
- Reuse at every level
- Cut corners wherever possible (trade accuracy against efficiency)
- Make a “this feature not implemented” message
Paper prototyping (3)

- Include “decoy” user interface objects not needed for expected tasks
- Accommodate data value entry by users with blank transparencies
- **Organize** materials to manage complex task threads
- **Pilot** test thoroughly
Welcome to videoStage!

1. Enter a search term above.
2. Click on a video to select it.
3. Click [GO] to send selected videos to the stage.
Welcome to videoStage!

1. Enter a search term above.
2. Click on a video to select it.
3. Click [GO] to send selected videos to the stage.
1. Enter a search term above.
2. Click on a video to select it.
3. Click [GO] to send selected videos to the stage.
video Stage

Currently listening to music.
Advantages of prototyping

• Offers concrete baseline for communication between users & designers
• Provides conversation “prop” to communicate concepts
• Allows user to “take design for a spin”
• Give project visibility & buy-in with customers
• Encourage early user participation and involvement
• Give impression that design is easy to change
• Afford designers immediate observation of user performance & consequences of design decisions
Conceptual Design of Transit Card Vending

• Design an interface for a machine that vends transit cards
• The machine accepts cash, coins, and credit cards
• The machine sells and reloads transit cards
• Transit cards can be loaded with:
  • Passes - valid for unlimited travel in the given period (1, 7, 30 days)
  • Value - Direct proxy for cash, used to pay fares
• Things to think about:
  • How does user decide to reload vs buy new card?
  • Can a card have both value and a pass on it? How does that work?