Design process

CS 347
Parastoo Abtahi & Danaë Metaxa
Reminders

Abstract revisions due Wednesday

Continue to participate and contribute to our discussions

Author a generative (rather than evaluative) commentary
“This paper has so many problems:”

“This paper inspired me to develop an idea:”
Commentary strategies

Future research directions that this paper inspires for you
Why the paper seems important
Observations of novel methodology
Why the paper is effective at getting its message across
How the paper has changed your opinion or outlook on a topic
Three "I like" s and one "I wish"
Feedback

Course Overview

INTRO

week 1  Intro to Interaction; Intro to Social Computing
week 2  Intro to Design; Interaction

DEPTH

week 3  Methods; Interaction
week 4  Social Computing
week 5  Design
week 6  AI+HCI; Algorithm Audits
week 7  Accessibility*; ICT4D
week 8  Foundations; Cognition
week 9  Collaboration; Visualization
week 10 (Programming); Critiques of HCI

BREADTH

INTRO

week 1  Intro to Interaction; Intro to Social Computing
week 2  Intro to Design; Interaction

DEPTH

week 3  Methods; Interaction
week 4  Social Computing
week 5  Design
week 6  AI+HCI; Algorithm Audits
week 7  Accessibility*; ICT4D
week 8  Foundations; Cognition
week 9  Collaboration; Visualization
week 10 (Programming); Critiques of HCI

BREADTH
Design is not a static process.

It can be studied, supported, and improved.
Brainstorming process
Early-stage design tools

Design

Implement
Programming tools
WYSIWYG design tools
Rapid prototyping tools

Evaluate
Study strategies
Cognitive modeling
Recall: reflective practitioner

Design is not a “plan, then do” praxis.
Instead, the designer is engaged in an ongoing conversation with the design.

Critically, it’s only by observing the result of the doing can the designer engage in reflection, allowing them to improve.
Recall: reflective practitioner

Implication:

To improve the process, encourage more rapid reflection, or improve the quality of the reflection

To improve the tools, create alternatives that make reflection easier to do or more informative
Recall: wizard-of-oz prototypes

[Kelley, TOIS ’84]

An iterative design methodology for user-friendly natural language office information applications

“Central to the methodology is an experimental simulation which I call the OZ paradigm, in which experimental participants are given the impression that they are interacting with a program that understands English as well as another human would.”
Today

Wicked problems

Pattern languages
  What are they?
  How do we learn about modern design patterns?

Why are we good (or bad) at design?
  Design fixation, and techniques to break out of it
  Feedback
Wicked problems

So, if you care to find me
look to the western sky
As someone told me lately
everyone deserves a chance to fly
“Math is easy. Design is hard.”

Jeffrey Veen, former VP of Product Design at Adobe
Why is design hard?

Design: “the transformation of existing conditions into preferred ones” [Simon 1969]

What is that makes design so hard to do well?

Why is it hard to find preferred solutions that are better and creative?
Wicked problems

[Rittel and Webber, Policy Sciences 1973]

The reason that design is often described as challenging is because it must solve **wicked problems**

Wicked problems are problems whose requirements are contradictory or unknown: no global optimum

- Urban planning: I can widen the streets, but this will create more traffic in the non-major streets
- Pedagogy: In CS 147, students simultaneously want to learn more design, and more technical depth—but nobody wants to extend past ten weeks
Design + wicked problems

[Zimmerman, Forlizzi, and Evenson, CHI '07]

Designers often wrestle with wicked problems

Example: Wikipedia reduced the amount of spam it got, at the cost of newcomers. It could be more welcoming to newcomers, at the cost of quality.

To solve wicked problems: integrate known facts, engineering opportunities, and user research to create a new perspective
“Certain phenomena are ‘artificial’ in a very specific sense: They are as they are only because of a system's being molded, by goals or purposes, to the environment in which it lives.”
Understanding the history and recognizing oppression.
Because we neglected to collect their stories in the past, we fail to know them in the present. Reviving their accounts informs our contemporary understanding of what innovation looks like and, in turn, shapes possibilities for building technology otherwise.”
Design patterns
The trouble with design

Design is a praxis with many degrees of freedom, but also many ways to screw it up.

Temptations to be different or creative can wind you up with terrible designs.

How do we maintain breathing space for new ideas while not accidentally stepping off the ledge?
Design patterns

[Alexander 1977]

Originated in urban planning

"Each pattern describes a problem which occurs over and over again in our environment, and then describes the core of a solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice."
Design patterns

[Alexander 1977]

"Metropolitan regions will not come to balance until each one is small and autonomous enough to be an independent sphere of culture": patterns include…

Community of 7000: not too many people, not too few people

Subculture boundary: e.g., SF Chinatown is bounded by the banking area

Identifiable neighborhood: < 500 people in local neighborhood units

Neighborhood boundary: subcultures in neighborhoods need restricted physical access
"Establish community and neighborhood policy to control the character of the local environment" : patterns include…

- Four story limit: keep most buildings <4 stories
- No more than 9% of land dedicated to parking
- Embellish sacred and meaningful sites in the neighborhood
- Ensure that the community include a balance of people at every stage of the life cycle
A **pattern language** is a set of design patterns, collected together, organized, and connected to each other.

A **pattern language** is a set of design patterns, collected together, organized, and connected to each other.

Connections to macro patterns

Focal pattern

Connections to micro patterns

- Work community
- Identifiable neighborhood
- Subculture boundary
- Accessible Green
- Positive outdoor space
- Garden wall
- Tree places
Interaction design patterns
[van Duyne, Landay and Hong, ’06]

Web design, much like urban planning, can be characterized by design patterns

Examples...

- News mosaics
- Distinctive HTML titles
- Quick-flow checkout
- Floating windows
Interaction design patterns

Web design also features pattern languages

Introduction

Material Design is a visual language that synthesizes the classic principles of good design with the innovation of technology and science.

Goals
Webzeitgeist

[Kumar et al., CHI '13]

Crawl the web and index large-scale design elements

Main idea: what happens if we start data mining design patterns, rather than user behavior?
ERICA: Interaction Mining Mobile Apps

Biplab Deka, Zifeng Huang and Ranjitha Kumar
{deka2, zhuang45, ranjitha}@illinois.edu

Data-Driven Design Group
University of Illinois at Urbana-Champaign
Why are we good (or bad) at design?

Design fixation and functional fixedness
Flare and focus

Buxton [2007] model of the design process lays out an ideal model

Flare: expansions of the possibility space

Focus: selections from within options
Flare and focus

Buxton [2007] model of the design process lays out an ideal model

**Flare:** expansions of the possibility space

**Focus:** selections from within options

In reality, flares and focuses occur within phases as well as you reflect
**Design fixation**

[Jansson and Smith Design Studies 1991]

**Fixation** is when we blindly adhere to an idea. Colloquially, we fall in love with our idea.

Fixation leads to lower-quality solutions, because we under-explore.

Vis-à-vis Schön, design fixation halts reflection in action.

(This is why we’re not so excited when you start a class with your project idea already firmly in mind.)
Quantity or Quality?

[Bayles and Orland 2001]

Can forcing yourself to try multiple options combat fixation and produce better designs?
Quantity or Quality?

[Bayles and Orland 2001]

“While the quantity group was busily churning out piles of work—and learning from their mistakes—the quality group had sat theorizing about perfection, and in the end had little more to show for their efforts than grandiose theories and a pile of dead clay”
Parallel prototyping

[Dow et al. 2010]

serial prototyping condition

parallel prototyping condition
Parallel design → more clicks

<table>
<thead>
<tr>
<th></th>
<th>Clicks per million impressions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parallel</td>
<td>445</td>
</tr>
<tr>
<td>Serial</td>
<td>398</td>
</tr>
</tbody>
</table>
Recall: How many designs?

[Tohidi et al. 2006]

“We found that when presented with a single design, users give significantly higher ratings and were more reluctant to criticize than when presented with the same design in a group of three.”

Let’s return to why this happened…
Functional fixedness

Designers are trained to question assumptions, and to creatively recombine the tools at their disposal. However, people exhibit a cognitive bias, functional fixedness, toward using objects only in the ways we’ve seen them used before.

Classic example: attach a candle to a wall so that the candle won’t drip on the floor. You can only use (1) a book of matches, (2) a box of thumbtacks.
Feedback
External reflection

Sometimes, the designer can execute reflection by looking themselves at the outcome. Other times, we need external feedback.

We are familiar with feedback coming from an expert, as in the atelier model

Could it come from elsewhere?
Peer assessment

[Kulkarni et al., TOCHI ’14]

How can we provide feedback to teach design to millions?

Case in point: Scott Klemmer’s HCI class on Coursera: thousands of submissions, thousands of students

Calibrated peer assessment: training students to give feedback on each others’ design assignments

What is “calibrated” about it? A similar training exercise to what you do before grading commentaries — give feedback on labeled examples first
See you on Wednesday :)

Action items


Before class on Wed: abstract revisions

Email: cs347@cs.stanford.edu  Course website: cs347.stanford.edu