CSE440: Introduction to HCI
Methods for Design, Prototyping and Evaluating User Interaction

Lecture 14: Patterns

Nigini Oliveira
Manaswi Saha
Liang He
Jian Li Zheng
Jeremy Viny
<table>
<thead>
<tr>
<th>Nov 12</th>
<th>Nov 13</th>
<th>Nov 14</th>
<th>Nov 15</th>
<th>Nov 16</th>
</tr>
</thead>
<tbody>
<tr>
<td>Veterans Day</td>
<td>Patterns</td>
<td>Ngin's office hours</td>
<td>Interface Implementation</td>
<td>Section</td>
</tr>
<tr>
<td>3d - Usability Testing Review</td>
<td>11:30 - 12:50</td>
<td>10:00 - 12:00</td>
<td>11:30 - 12:50</td>
<td>10:30 - 11:20</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CSE 218</td>
<td></td>
<td>11:30 - 12:20</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1:30 - 2:20</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2:30 - 3:20</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3e - Digital Mockup</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Nov 19</th>
<th>Nov 20</th>
<th>Nov 21</th>
<th>Nov 22</th>
<th>Nov 23</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recap and Exam Q&amp;A Session</td>
<td>Ngin's office hours</td>
<td>Thanksgiving</td>
<td></td>
<td>Thanksgiving</td>
</tr>
<tr>
<td>11:30 - 12:50</td>
<td>10:00 - 12:00</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OUG 141</td>
<td>CSE 218</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Nov 26</th>
<th>Nov 27</th>
<th>Nov 28</th>
<th>Nov 29</th>
<th>Nov 30</th>
</tr>
</thead>
<tbody>
<tr>
<td>NO CLASS</td>
<td></td>
<td>Ngin's office hours</td>
<td>Exam</td>
<td>Section</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10:00 - 12:00</td>
<td>11:30 - 12:50</td>
<td>10:30 - 11:20</td>
</tr>
<tr>
<td></td>
<td></td>
<td>CSE 218</td>
<td>OUG 141</td>
<td>PAR 112</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>11:30 - 12:20</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>OUG 138</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1:30 - 2:20</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>2:30 - 3:20</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3f - Getting the Design Right</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>3g - Presentation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>[critique in section]</td>
</tr>
</tbody>
</table>
Course Progress

**Framing the problem**
- User research
- Competitive analysis
- Data analysis and summary

**Exploring the solution space**
- Brainstorming
- Ideation through sketching

**Finding a good solution**
- Scoping
- Consideration of constraints
- Scenarios, storyboards, personas
- Design rationale

**Refining the solution**
- Wireframes
- Lo-fi prototypes
- Early evaluations
- Mockups/mid-fi prototypes
- Additional evaluations
Patterns

“… a general repeatable solution to a commonly-occurring usability problem in interface design…”

Patterns
Patterns
Site-branding
Value Proposition
What can I do here?
Information above the fold
Patterns
Consistency to reinforce branding
User location within a site
### Weezer (2001)

**Our best price:** $6.99  
**List Price:** $10.97 ($Save: $4.08)

**Find out more...**
- Full product info  
- Product Reviews

**Not ready to buy?**
- Add to your Wish List  
- Preorder this item. May also be available...  

**Compare our Prices!**

<table>
<thead>
<tr>
<th>Seller</th>
<th>Price</th>
<th>Total Price</th>
<th>Seller Comments</th>
<th>Seller Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Half.com</td>
<td>$6.99</td>
<td>$10.20 Media Mail</td>
<td>best buyer</td>
<td>More info...</td>
</tr>
<tr>
<td>$8.00</td>
<td>695</td>
<td>Media Mail</td>
<td>Perfect condition</td>
<td>More info...</td>
</tr>
<tr>
<td>$8.25</td>
<td>696</td>
<td>Media Mail</td>
<td>Disk, case, and liner all in excellent condition</td>
<td>More info...</td>
</tr>
<tr>
<td>$8.30</td>
<td>697</td>
<td>Media Mail</td>
<td>Perfect condition</td>
<td>More info...</td>
</tr>
</tbody>
</table>

**Like New**
- Sorted by Price

<table>
<thead>
<tr>
<th>Price</th>
<th>Total Price</th>
<th>Seller</th>
<th>Seller Comments</th>
<th>Seller Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>$7.75</td>
<td>Media Mail</td>
<td>custodian (469)</td>
<td>best buy</td>
<td>More info...</td>
</tr>
<tr>
<td>$8.00</td>
<td>695</td>
<td>Media Mail</td>
<td>Perfect condition</td>
<td>More info...</td>
</tr>
<tr>
<td>$8.25</td>
<td>696</td>
<td>Media Mail</td>
<td>Disk, case, and liner all in excellent condition</td>
<td>More info...</td>
</tr>
<tr>
<td>$8.30</td>
<td>697</td>
<td>Media Mail</td>
<td>Perfect condition</td>
<td>More info...</td>
</tr>
</tbody>
</table>

**Vary Good**
- Sorted by Price

<table>
<thead>
<tr>
<th>Price</th>
<th>Total Price</th>
<th>Seller</th>
<th>Seller Comments</th>
<th>Seller Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>$10.45</td>
<td>Media Mail</td>
<td>luckkah (44)</td>
<td>perfect</td>
<td>More info...</td>
</tr>
<tr>
<td>$11.20</td>
<td>Media Mail</td>
<td>steveent (82)</td>
<td>perfect</td>
<td>More info...</td>
</tr>
<tr>
<td>$9.00</td>
<td>695</td>
<td>Media Mail</td>
<td>Great shape...first class ship</td>
<td>More info...</td>
</tr>
</tbody>
</table>
Personalized recommendations

- Weezer (2001)
  - People who bought "Weezer (2001)" also bought:
      - Weezer
      - Save $6.97 - Best price: $5.00
    - Pinkerton CD, Release Year: 1996
      - Weezer
      - Save $10.95 - Best price: $6.00
    - Hybrid Theory CD, Release Year: 2000
      - Linkin Park
      - Save $11.68 - Best price: $6.29

About this album

Song List
1. Don't Let Go
2. Photograph
3. Hashpipe
4. Island In The Sun
5. Crab
6. Knock-Down Drag-Out
7. Smile
8. Simple Pages
9. Glorious Days
10. O Girlfriend

Album Credits
Ken Allerdyce, Engineer
Ric Ocasek, Producer
Secondary information

Album Notes
Weezer: Rivers Cuomo (vocals, guitar); Brian Bell (guitar); Matt Sharp (bass); Patrick Wilson (drums). Recorded at Cello Studios, Los Angeles, California in December 2000. In 1994 Weezer burst onto the music scene, reaching platinum status with their debut, and in the process proving that there was still room in an airbrushed MTV world for unrepentant power pop played by decidedly non-airbrushed guys. Following a brief sojourn into semi-deconstructionism, 1997's Pinkerton, the four men who make up Weezer serve up a third offering, Weezer 2001, returning to the sound and producer of their successful debut. Nowhere does producer Ric Ocasek define his trademark refined power pop style more than with Weezer. Unlike the immediate, obvious pop hooks of the string of singles on the first album, though, the songs on Weezer 2001 may take a few listens to settle in. However, once the subtle-yet-undeniable refrains of such tracks as "Crab," "Don't Let Go," and first single "Hash Pipe" make their way into your skull, they're there to stay, as furious, fuzzy, layered guitars compliment Rivers Cuomo's raw, vulnerable vocals. While this disc clocks in at less than a half-hour long, it packs more hooky wallop than many double live albums.

Product Reviews

Editorial Reviews
Spin (01/01/2002)
Ranked #9 in Spin's Albums of the Year 2001 - ...They are the new Cars...Alternative Press (2/02, p.65) - Ranked #13 in AP's 25 Best Albums of 2001 - ...This will either touch your heart or make you air-guitar like a beast...Rolling Stone (6/7/01, p.110) - 4 stars out of 5 - ...A totally crunk geek-punk record, buzzing through 10 excellent tunes in less than half an hour with zero filler and enough psychosexual contortions to buy leader Rivers Cuomos shrink another hot tub...Q Magazine (8/01, p.142) - 4 stars out of 5 - ...Harks back to the keenly observed power pop of their '94 debut, and there isn't a bad apple in the bunch...

Customer Reviews
Rated 4.3 out of 5.0 by 29 raters.
» Read Customer Reviews
» Rate this item

Portions of this page Copyright 1998-2001 Muze Inc. Muze® For personal non-commercial use only. All rights reserved. Provider 1
Do we still know where we are?
Do we still know where we are?
Avoid surprises!
Provide support
Provide support
Where am I in the process?
Where am I in the process?
Where am I in the process?
Process funnel
Extraneous info and links are removed to focus customers
How much more to do?
How much more to do?
Make it easy to sign up
Why this completed outdated example?
Why this completed outdated example?

It’s still the same today :)
Activity

5min

Go to your favorite shopping site and find the differences!
Design Equals Solutions

Design is about finding solutions
Design Equals Solutions

Design is about finding solutions

Designers often reinvent
- Hard to know how things were done before
- Why things were done a certain way
- How to reuse solutions
Design Patterns

Design patterns communicate common design problems and solutions

A tool for knowledge sharing
Design Patterns in Architecture

Design patterns communicate common design problems and solutions

Somewhere in the community at least one big place where a few hundred people can gather, with beer and wine, music, and perhaps a half-dozen activities, so that people are continuously criss-crossing from one to another.
Design Patterns in OOP and software engineering

Nice patterns for designing window-based user interfaces in Smalltalk

The Gang of Four Book!
Using Design Patterns

Not too general and not too specific

use a solution “a million times over, without ever doing it the same way twice”
Using Design Patterns

Not too general and not too specific
use a solution “a million times over, without ever doing it the same way twice”

Design patterns are a shared language
for “building and planning towns, neighborhoods, houses, gardens, and rooms”
Beer hall is part of a center for public life
Beer hall needs spaces for groups to be alone ALCOVES
Web Design Patterns

Communicate design problems & solutions

- how to create navigation bars for finding relevant content
- how to create a shopping cart that supports check out
- how to make e-commerce sites where people return & buy
Navigation Bar (K2)

Problem: Customers need a structured, organized way of finding the most important parts of your Web site
Navigation Bar (K2)

Solution diagram

Captures essence on how to solve problem
Patterns Support Creativity

Patterns come from successful examples
  so successful that lots of people are familiar with their paradigms
  interaction techniques/metaphors
    that work well across many sites (e.g., shopping carts)

Not too general and not too specific
  you need to specialize to your needs

Patterns let you focus on the hard, unique problems of your design situation
Principles, Guidelines, Templates

Principles?
patterns are not that general

Guidelines?
patterns discuss trade-offs,
show good examples, and tie to other patterns

Style guides?
patterns are not too specific,
can be specialized to a design

Templates?
patterns illustrate flows and relationships among pages

Patterns help design without over-constraining
UI Pattern Form

**Problem:** Problems are related to the usage of the system and are relevant to the user or any other stakeholder that is interested in usability.

**Use when:** a situation (in terms of the tasks, the users and the context of use) giving rise to a usability problem.

**Principle:** a pattern is usually based on one or more ergonomic principles such as user guidance, or consistency, or error management.

**Solution:** a proven solution to the problem.

**Why:** How and why the pattern actually works, including an analysis of how it may affect certain attributes of usability.

**Examples:** Each example shows how the pattern has been successfully applied in a real life system. This is often accompanied by a screenshot and a short description.

**Implementation:** Some patterns provide implementation details.

---

**References:**

Example of a UI pattern

Undo in Microsoft Word using a visualization of the command queue

Example of a UI pattern

<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem</td>
<td>Users do actions they later want reverse because they realized they made a mistake or because they changed their mind.</td>
</tr>
</tbody>
</table>
Example of a UI pattern

Use when

You are designing a desktop or web-based application where users can manage information or create new artifacts. Typically, such systems include editors, financial systems, graphical drawing packages, or development environments. Such systems deal mostly with their own data and produce only few non-reversible side-effects, like sending of an email within an email application. Undo is not suitable for systems where the majority of actions is not reversible, for example, workflow management systems or transaction systems in general.

Both novice and expert users may want to reverse their actions, either because of mistakes or changes in intention. Expert users may want to use the history of their actions for more specific manipulation of the data in the application. For example, in a graphical modeling application, users may want to undo work on some specific object while keeping later work done on other objects.
Example of a UI pattern

Solution

Maintain a list of user actions and allow users to reverse selected actions.

Each 'action' the user does is recorded and added to a list. This list then becomes the 'history of user actions' and users can reverse actions from the last done action to the first one recorded. This is also called a Linear Multi-level Undo.
Example of a UI pattern

Why

Offering the possibility to always undo actions gives users a comforting feeling. It helps the users feel that they are in control of the interaction rather than the other way around. They can explore, make mistakes and easily go some steps back, which facilitates learning the application’s functionality. It also often eliminates the need for annoying warning messages since most actions will not be permanent.
Example of a UI pattern

As in all MS Office applications, in Word 2000 the users can see the history of their actions and undo one or more of them. The actions are briefly described and the users can select a range of actions to be undone. After selecting undo, users can even redo the actions.

In Photoshop a selective undo is also possible. By moving the slider, users can do the normal multi-level undo but they can also drag an action into the trashcan and thereby do a selective undo. Operations that depended on that action are automatically deleted as well if they are not relevant anymore.
Example of a UI pattern

Implementation

Most implementations of multi-level undo are based on the Command (Gamma et al 1995) pattern. When using the Command pattern, most functionality is encapsulated in Command objects rather than in other controlling classes. The idea is to have a base class that defines a method to "do" a command, and another method to "undo" a command. Then, for each command, you derive from the command base class and fill in the code for the do and undo methods. The "do" method is expected to store any information needed to "undo" the command. For example, the command to delete an item would remember the content of the item being deleted.
More examples

**Captcha**

**Problem**
The application needs confirmation that the action that is about to happen is done by a human rather than an automated machine.

**Solution**
Present users with a mangled image containing numbers and letters that humans can still 'decipher' but is hard for machines to read.

**Verify Your Registration**

* Enter the code shown: [Image of the code 'duswcz']

This helps Yahoo! prevent automated registrations.

http://www.welie.com
More examples

Date Selector

Problem
Users need to select a date or a time period

Solution
Use a combination of an edit box and a graphical clickable calendar

From www.expedia.com

http://www.welie.com
Pre-patterns

Patterns require broad adoption and examples
  Many version of the same basic idea
  Shown successful in many contexts
  That is what makes them patterns

This is challenging in novel domains

Pre-patterns are based in weaker evidence
  Can help speed diffusion of techniques and results
  Can help see relationships among ideas
Pre-patterns and Anti-patterns

When you see advice, consider its depth
   Result of an individual study
   Pre-pattern based on some meta-analysis
   Established pattern

Be aware of misapplying patterns
   And be aware of anti-patterns
Dark Patterns

A design that has been carefully crafted to trick people into doing things, such as buying insurance with their purchase or signing up for recurring bills.

Example: Disguised Ads

Ads that are disguised as other kinds of content or navigation, in order to get users to click on them.
Dark Patterns

A design that has been carefully crafted to trick people into doing things, such as buying insurance with their purchase or signing up for recurring bills.

Example: Disguised Ads

Ads that are disguised as other kinds of content or navigation, in order to get users to click on them.
Dark Patterns

A design that has been carefully crafted to trick people into doing things, such as buying insurance with their purchase or signing up for recurring bills.

Example: Disguised Ads
   Ads that are disguised as other kinds of content or navigation, in order to get users to click on them

Example: Friend Spam
   A site or game asks for your credentials, then goes on to publish content or send out bulk messages
Dark Patterns

After Lawsuit Settlement, LinkedIn’s Dishonest Design Is Now A $13 Million Problem

Hopefully, this will be a lesson to other companies who use dark UX patterns to trick their users.

Anyone who has ever signed up, or even known anyone who has signed up, for LinkedIn has probably found themselves on the receiving end of dozens of follow-up emails, inviting you to “expand your professional network.” Even worse, they’re virtually impossible to opt-out of. It’s a shifty use of dark UX patterns by a company that should know better. Now, LinkedIn is going to be paying for it as part of a class-action lawsuit, to the tune of $13 million.

Presented in San Jose’s U.S. District Court, the key issue in Perkins v. LinkedIn is spam. Namely, during the user sign-up process, LinkedIn claims that it “will not store your password or email anyone without your permission.” Despite this, LinkedIn sends automated follow-up email reminders on a new user’s behalf to any contacts harvested from their webmail accounts, which are presented in such a way as to appear as if they came directly from the user.

Under California law, the sitting judge says has deemed this illegal. Consequently, if you were a member of LinkedIn’s “add connection” program between September 2011 and October 2014, you can submit a claim to get a payout.
Pattern exercise

In groups of two, read and answer the handout.

(Submit a PDF with your answers to CANVAS.)
Ask me something!