CS160 / 260A: User Interface Design

The Design Cycle & Sketching

01/27/14

Berkeley

University of California
Enrollment

1. Decisions were sent earlier today; we could not admit everyone.
2. Decisions are **final** and we will not keep admitting students from the wait list.
3. Dropping the course after this week negatively affects your team.
4. Official database update may have latency.
Font Selection in Keynote
Font Selection in PowerPoint
Design Assignment

Convenience samples?
Be specific in guiding interviews to the question(s) you really care about.
Best interviews often happen in context.
Spelling and grammar matter!
Please check before submitting.
Next Assignments

Wednesday 5:30pm:
Reading response (cs160/cs160Reading)

Friday, 11:59pm:
Individual Design Assignment, Part 2
Brainstorm ideas, select favorite, prototype solution (draw it!), get feedback.
You are allowed to brainstorm in groups.
All other steps are individual!

Individual Programming Assignment
Android Crypto-currency converter
Pick up class account forms at the end of this lecture
Main Page

CS 160 is an introduction to Human Computer Interaction (HCI). You will learn to prototype, evaluate, and design a user interface. You will be expected to work within a group of four or five students in this project-based course. The project topic will be proposed by your group, and your implementation will be tailored to your users’ needs based on interviews with them.

In contrast to most of the other CS classes at Berkeley, CS160 does not (only) focus on particular algorithmic techniques or computer technologies. Instead, you will make use of technology to develop your applications, and you will acquire some expertise in the development environment you choose. The focus of the course is on developing a broad set of skills needed for user-centered design. These skills include ideation, needs assessment, communication, rapid prototyping, algorithmic implementation and evaluation.

**Project Theme:** This semester, projects will focus on mobile applications that make use of sound input as a key feature of the application. Mobile applications present unique opportunities (e.g., sensors, microphone, camera) and challenges (e.g., text input). Your team will be developing applications using the Google Android SDK. You can use your own personal Android device for development. We encourage teams to purchase a recent Android device, e.g., the Nexus 7 tablet ($230) if none of your team members have a suitable device yet. We will also have a small number of Android phones and tablet devices available to borrow for the semester (one per team); these are older models and limited in number.

Announcements  [edit]

- First Day of Class: Wednesday, January 22, 2014.
  You must at sign in during this class if you are on the wait list to be considered for admission.

Schedule  [edit]

**Week 1**  [edit]

W Jan 22: Introduction  [ Slides | Live Lecture Video (later also archived on YouTube) ]

  Overflow: After 306 Soda fills up, please go to our overflow room, 310 Soda. We will post a video of the first lecture online as
The Design Cycle and Brainstorming

Readings [edit]


Optional Readings [edit]

- How to Run a Design Critique ᴡ Scott Berkun.
- How to Give and Receive Criticism ᴡ Scott Berkun.

Reading Responses [edit]

Weishi Wu - 1/22/2014 21:14:05 [edit]

Comparing the three stages mentioned at class. I think the Lewis & Rieman design cycle provides more important supporting details, yet follows the same way as 'design, prototype, evaluate'. They also focus on identifying target user group's need, starting with low fidelity prototype instead of high fidelity, gain design experience from existing UI, etc. These details are important aspects to concentrate on for UI designer.

The basic point and design processes mentioned from the article is well presented. While comparing to software design & analysis process to airplane industry is inappropriate, at least in these days when there are tons of new apps in the market every day. The testing methods between those two industries are biased, and the cost of software testing is getting down with usually shorter design/constructing period. Testing the a jet airline's structure could use complex CAD, FEM software like Ansys/Adams plus expensive real plane stress test, which can take a long period and spend a significant amount of money. The app development period is much shorter.

For today user group meeting for large system product analysis, in the design stage, instead of just talking to random user group, I would interview with extreme users, but test my final product with normal ones. This is because by talking to extreme users, for example, if I plan to design a pc game user interface I probably will spend time talking to those crazy pc gamers first. From it designer will gain more inspiring thoughts because this type of user spend a lot time with the products, and will definitely yield some surprises/ideas/improvement about the application/product. Then evaluating the high fidelity product with normal users since they will be the majority of consumers.

Tristan Jones - 1/22/2014 22:04:05 [edit]

1) The design talked about in lecture is pretty similar to what Lewis and Reiman talk about in their book: the only caveat is that theirs has a bit more steps. Besides "test the design" and "plagiarize", if you expand out all the steps of "design, prototype, evaluate" you get pretty much the same thing. Lewis & Reiman just spell out the implicit steps. I guess that's good for a book, but for a quick mnemonic I think "design, prototype, evaluate" wins for simplicity.
Grading Guidelines

Convince us that you’ve carefully thought about the assigned reading. Scale:

1. Exceptional (10/10 pts)
2. Fine (~8 pts)
3. Lacking (5-6 pts)
4. No answer (0 pts)
Help with Programming Assignment

Section on Thursday
Start attending your assigned section.

Recommended:
Follow the official Android tutorials. E.g.:
Building Your First App
http://developer.android.com/training/basics/firstapp/index.html

Office hours:
Eric: 651 Soda Hall Alcove, M 11am-12pm
Bjoern: 533 Soda Hall, M 4:30-5:30pm and by appointment
Maneesh: 535 Soda Hall, T 4-5pm and by appointment
Steve: 510 Soda Hall (Visual Computing Lab), W 1-2pm
Brittany: 651 Soda Hall Alcove, Th 2:30-3:30pm
Concepts you will need to learn

Creating layouts and widgets
Edit XML file or use Eclipse GUI editor

Referring to widgets in Java code
e.g.: findViewById(R.id.foobar);

Adding event listeners
code that gets called when the user interacts with widgets

Showing output on the screen
e.g., TextView.setText()
Topics for Today

1. The Design Cycle
2. Brainstorming
3. Sketching
4. Critique
The Design Cycle
Design

Prototype

Evaluate
The Design Process

1. Acceptance
2. Analysis
3. Definition
4. Ideation
5. Idea selection
6. Implementation
7. Evaluation
The Design Process [Koberg & Bagnall]

1. Acceptance
2. Analysis
3. Definition
4. Ideation
5. Idea selection
6. Implementation
7. Evaluation

Design Process
Design Cycle Over Project Lifespan

1. Acceptance
2. Analysis
3. Definition
4. Ideation
5. Idea selection
6. Implementation
7. Evaluation

Number of Ideas under consideration

Project timeline

Final product
Evaluation reveals problems with design. Re-design requires cycling the process.
Prototype implementations eventually increase in fidelity to reach final product
Waterfall Model (Soft. Eng.)

- Initiation
  - Application Description
    - Analysis
    - Requirements Specification
      - Design
      - System Design
    - Implementation
  - Product
**Comparison**

**Focus differs**

WF has no feedback

High cost of fixing errors: increases by 10x at each stage

Iterative design finds problems earlier

True for modern web applications?
Design Process in Action
Video: The Deep Dive

How well do they follow the cycle?
What do they do for each step of the cycle?
How many cycles do you think they went through?
Brainstorming
Enhancing Creativity

Thinking outside the box:
Draw a series of 4 straight lines through all the points below, without lifting pen from paper:
Why Is This Hard?

We adopt expectations about the solution
Based on conventions
Based on what we believe the questioner expects
IDEO’s Brainstorming Rules

1. Sharpen the Focus
2. Playful Rules
3. Number your Ideas
4. Build and Jump
5. The Space Remembers
6. Stretch Your Mental Muscles
7. Get Physical

Aim for quantity
Hope for quality
Sharpen the Focus

Posing the right problem is critical – neither too narrow, nor too fuzzy

Not “bicycle cup-holders” but “helping cyclists to drink coffee without accidents”
Number Your Ideas

Obvious but very useful

Helps keep track of them when the brainstorm is successful (and 100 or more ideas are in play)

Allows ideas to take on an identity of their own
Build and Jump

**Build to keep momentum on an idea:**
“shock absorbers are a great idea; what are other ways to reduce coffee spillage on bumps?”

**Jump to regain momentum when a theme tapers out:**
“OK, but what about hands-free solutions?”
Concept Refinement

Premature idea rejection is a serious barrier to good design.

One big differentiator between good designers and great ones is the latter’s ability to successfully develop unusual ideas.

This requires a strong instinct to be able to distinguish fatal vs. minor flaws in an idea.
The Space Remembers

Covering whiteboards or papering walls with text is extremely useful in group work.

It’s a very effective form of external (RAM) memory for group

Even better, its shared RAM. Helps group share understanding
Stretch your Mental Muscles

**Warm-ups:** word games, puzzles

Get immersed in the domain: go visit the toy shop, or the bicycle shop, phone shop etc…

**Props:** Bring some examples of the technology to the brainstorm
Get Physical

Sketch

Make models

Act out

Moggridge, Designing Interactions, p.732
Sketching
Suggestion: Keep a Design Journal

From B. Buxton, Sketching User Experiences
Storyboards
Storyboard for Disney’s Melody: Adventures in Music (1953)
Source: Michael Sporn Animation
1. NOW STUDENTS
   LISTEN CLOSE
   TO ME -

2. CAUSE TODAY
   WE'RE GONNA TALK
   ABOUT MEL O'DEE

3. NOW ALL THE
   PROFESSORS AND
   CATS HAVE FOUND
Storyboards for UI Design

Goal: Understand how your product or application fits into a larger context.
Shows a single scenario / tells a single story
Start by setting the stage:
Then show key interactions with your application
Zoom back out and show the consequences of using the application
Could be satisfaction, but also think about errors
Example: Public Transit Storyboard

Tom, a CS student, is on campus and wants to take a trip to SF by BART.

"I wonder when I'll have to start walking down hill to catch the train..."

So he pulls out his phone and launches the "C.I.T." app.

Using dropdown menus, he selects the downtown Berkeley BART, and 10 minutes of walk time
Critique this storyboard in two ways:

1) Formally (i.e., how well does the storyboard itself convey the idea)
2) Content (i.e., critique the application idea itself)

The CIT app shows a graphical overview of how much time Tom has left. To see more detail,...

... He tilts his phone to landscape mode. He has 8 minutes before he has to leave...

Enough time to grab a quick espresso at Brewed Awakening!
If time: exercise

Draw a storyboard for a “friend finder” app.
Show a concrete scenario that motivates use
Show the interaction how to select a friend and find out where they are (or don’t find out, if they’ve set privacy preferences).
Conclude with a successful or unsuccessful meeting.
Critique

How to give & receive constructive criticism
What is a critique?

Show a project in progress through sketches and prototypes

Solicit feedback from peers (*small groups work best*)

History: Studio art education

http://www.flickr.com/photos/pjchmiel/2972140234/
What is the point of a critique?

Show off how great your project is.

Get honest reactions, ask for input on open questions.

Q: How is a critique different from a brainstorm?

http://www.flickr.com/photos/crystiancruz/2353909834/
Designer: Frame the discussion!

State Explicitly: What would you like comments on?
- Overall idea?
- Specific interactions?
- Usability?
- Technical Feasibility?
- Pixel-level graphic design?

Take a dispassionate stance (this is hard!)
- Show alternatives where possible
  (makes comparison easier)
Critic: How to avoid deaf ears

Comments are about the **design**, not the designer.

Point out positive aspects – be specific

Not: “I like this, but…”

Instead: “The layout effectively communicates the hierarchical nature of the data. However, …”

Ask for alternatives instead of offering solutions

Not: “You should really change X”

Instead: “Have you considered alternatives for X?”
For Wednesday

Lecture topic: Sound-based interaction

Reading: Don Norman, Design of Everyday Things (we’ll cover this in a future lecture)
Don’t forget to submit your reading response before class!
cs160 class accounts

Come to the front of the room to pick them up.