Formative Studies + Prototyping on Easy Mode
Reading Reflection

• Based on important themes from earlier in the course, come up with at least two additional cons around relying on think-aloud data, beyond the cons listed in the readings.
• Is there anything about talk-aloud protocols that should make us a little less nervous about these?
• Do you feel like you could already design a talk-aloud study for the language or tool you’re designing for your final project?
  • If yes, what would you do?
  • If no, what part of that design process is the barrier?
Plan for today

• A quick reminder of the prototype lessons from Tuesday
• How prototypes let us do formative user studies
• An easy-mode skeleton for formative studies, for those who aren’t looking to explore the full space
• Prototyping+formative study in-class activity
Prototyping

“…users can't tell you what they want, but when they see something and get to use it, they soon know what they don't want.”
Prototype Roles

• Make you think harder, plan more thoroughly about what you want to build
• Help you solicit feedback on the thing you plan to build
Low- vs. High-Fidelity Prototypes
Interactable, higher-fidelity

Low-Fidelity Prototypes

• Claims you may hear about low-fi prototypes:
  • People love to give you feedback on font size and if your icons make sense to them
  • If you don’t want that kind of feedback, if you want feedback on elements deeper than aesthetics, consider low-fidelity prototypes
  • Also if it looks like you drew it in crayon and didn’t sink a lot of time into it, people are more willing to criticize, which is what you want

• Personally haven’t found research-backed evidence of the above
  • (Send me your references!)

• But…lots of evidence that you get just as much/just as good feedback from low-fi, and they’re faster and cheaper to make, faster to tweak and change
Low-Fidelity Prototypes

- **But**...lots of evidence that you get just as much/just as good feedback from low-fi, and they’re faster and cheaper to make
- ...with the result that maybe *you’re* more willing to criticize yourself and to throw things away when you realize they’re not right
A nice resource on the case for low-fi prototypes

- With good arguments for the claims mentioned on prior slides
Wizard-of-Oz Prototyping

• Like what we did the very first day of class!
• Lets us get around engineering effort by having a human do the work that our tool will eventually automate
  • Human can be:
    • Compiler, interpreter
    • Program synthesizer
    • Programming environment
    • Program transformation tool
    • ...

We’ve talked about lo-fi...

• ...because for today’s purposes, we’re mostly interested in early-stage formative studies

• But of course we want to be getting feedback from users at all points!

• Calling it low-fidelity naturally suggests the existence of high-fidelity...
<table>
<thead>
<tr>
<th>Type</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Low-fidelity</td>
<td>Lower development cost</td>
<td>Limited error checking</td>
</tr>
<tr>
<td>prototype</td>
<td>Evaluates multiple design concepts</td>
<td>Poor detailed specification to code to</td>
</tr>
<tr>
<td></td>
<td>Useful communication device</td>
<td>Facilitator-driven</td>
</tr>
<tr>
<td></td>
<td>Addresses screen layout issues</td>
<td>Limited utility after requirements established</td>
</tr>
<tr>
<td></td>
<td>Useful for identifying market requirements</td>
<td>Limited usefulness for usability tests</td>
</tr>
<tr>
<td></td>
<td>Proof of concept</td>
<td>Navigational and flow limitations</td>
</tr>
<tr>
<td>High-fidelity</td>
<td>Complete functionality</td>
<td>More resource-intensive to develop</td>
</tr>
<tr>
<td>prototype</td>
<td>Fully interactive</td>
<td>Time-consuming to create</td>
</tr>
<tr>
<td></td>
<td>User-driven</td>
<td>Inefficient for proof-of-concept designs</td>
</tr>
<tr>
<td></td>
<td>Clearly defines navigational scheme</td>
<td>Not effective for requirements gathering</td>
</tr>
<tr>
<td></td>
<td>Use for exploration and test</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Look and feel of final product</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Serves as a living specification</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Marketing and sales tool</td>
<td></td>
</tr>
</tbody>
</table>

Interaction Design: Beyond Human - Computer Interaction by Yvonne Rogers et al.
Formative Research
Pop Quiz:
What’s the very first thing you do when you start designing a user study?
What’s the very first thing you do when you start designing a user study?

Choose your research question!!!!
Three Categories of User Study RQs

Need Finding Study
What are interesting problems to solve?

Formative Study
For a given problem, what are promising solutions?

Evaluative Study
For a given problem, now that we’ve implemented a solution, did it work?
Three Categories of User Study RQs

Need Finding Study
What are interesting problems to solve?

Formative Study
For a given problem, what are promising solutions?

Evaluative Study
For a given problem, now that we’ve implemented a solution, did it work?

Two weeks ago
Shape of a Need Finding RQ

What kinds of problems does <description of audience> face during <description of tasks>?

For this class, usually…

What kinds of problems does <description of audience> face during <description of programming tasks>?
Three Categories of User Study RQs

Need Finding Study
What are interesting problems to solve?

Formative Study
For a given problem, what are promising solutions?

Evaluative Study
For a given problem, now that we’ve implemented a solution, did it work?

This week
Examples of Formative RQ

Now we’re trying to build something, some kind of intervention. So what do we need to know about users in order to build interventions that will actually serve them?

...

Are <description of audience> more <successful/efficient/bug-free/comfortable> doing interaction X or interaction Y?

When <description of audience> describe <description of task> to a human assistant, how do they explain the task spec?

When we ask <description of audience> to do <description of task> with <tool>, what goes wrong?
Why are we treating Formative Studies and Prototyping together?
Why are we treating Formative Studies and Prototyping together?

- Formative Studies
  - What kind of prototype should we make in order to answer our research question?
- Prototyping
  - How should we design the study protocol based on what kind of prototype we can create?
Why are we treating Formative Studies and Prototyping together?

Formative Studies

What kind of prototype should we make in order to answer our research question?

Prototyping

How should we design the study protocol based on what kind of prototype we can create?
Why are we treating Formative Studies and Prototyping together?

1. What kind of prototype should we make in order to answer our research question?

2. How should we design the study protocol based on what kind of prototype we can create?
There’s a spectrum

User-Centered language design on easy mode

User-Centered language design, the full experience
User-Centered PL: Easy Mode

Motivation:
I think programming languages and programming tools are for humans. I want to make PLs that useful and usable. But I don’t care about contributing to generalizable human-centered programming knowledge.

Approach:
• Before implementation, make slides or other documents showing worked examples for multiple approaches and discuss them with users.
• Throughout implementation, regular think-alouds with current prototype.
User-Centered PL: The Full Experience

**Motivation:**
I think programming languages and programming tools are for humans, and every part of my process from deciding what need to tackle to deciding how to tackle it to refining my PL will be driven by understanding users and how my tools interact with user needs.

**Approach:**
- Contextual inquiry and ethnographic studies for need finding
- Formative studies and thorough prototyping
- Usability studies play a role in evaluation
Qualitative vs. Quantitative Formative Studies
Quantitative Formative Usability Studies

If you’re thinking of going quantitative instead of qualitative:

- Can you actually measure the thing you’re trying to measure? (How, specifically?)
- Will the particular experiment you’re planning actually succeed in measuring the thing you’re trying to measure?
- If you measure it successfully, will it actually help you improve your design?

If any “no”s, head back to qualitative.
Qualitative Research Takeaways

...qualitative research helps us understand:
- Behaviors, attitudes, and aptitudes of potential product users
- Technical, business, and environmental contexts — the domain — of the product to be designed
- Vocabulary and other social aspects of the domain in question
- How existing products are used

About Face: The Essentials of Interaction Design, Cooper et al.
Qualitative Research Takeaways

To get the really exciting stuff from qualitative studies:

- Interview where the interaction happens
- Avoid a fixed set of questions
- Focus on goals first, tasks second
- Avoid making the user a designer
- Avoid discussions of technology
- Encourage storytelling
- Ask for a show and tell
- Avoid leading questions
Qualitative Research Takeaways

You’ve already seen what you can learn from qualitative research in the context of need finding

But we’re not limited to need finding activities!
Qualitative Usability Studies

...which brings us back to think-aloud studies

- Minimal planning — pick a task
- Easy and fast to run a session
- Useful at any stage of brainstorming or implementation
- Shockingly informative
- Shockingly persuasive to others
  - Great way to persuade your advisor something actually matters :)

Qualitative Usability Studies
Qualitative Usability Studies

Usability testing is especially effective at determining:

- **Naming** — Do section/button labels make sense? Do certain words resonate better than others do?
- **Organization** — Is information grouped into meaningful categories? Are items located in the places customers might look for them?
- **First-time use and discoverability** — Are common items easy for new users to find? Are instructions clear? Are instructions necessary?
- **Effectiveness** — Can customers efficiently complete specific tasks? Are they making missteps? Where? How often?
Qualitative Usability Studies

Programming languages, to the extent that they require even more time and effort to learn than traditional user interfaces, exacerbate some of the existing problems of usability studies (both qualitative and quantitative).

focused on assessing the first-time use of a product. It is often quite difficult (and always laborious) to measure how effective a solution is on its 50th use — in other words, for the most common target: the perpetual intermediate user. This is quite a conundrum when one is optimizing a design for intermediate or expert users. One technique for accomplishing this is the use of a \textit{diary study}, in which subjects keep diaries detailing their interactions with the product. Again, Mike Kuniavsky provides
Qualitative Usability Studies

Even though qualitative usability may be more flexible, easier for you to adapt in the moment, you don’t completely escape the need to plan the design!

- Am I interacting with the right users?
- Do I know what kinds of information I’m seeking? (So that I can watch the right tasks, design the right tasks, ask the right questions?)
- And remember, always always always start by knowing your research question! Why are you bothering to run this study in the first place? What’s that core question you’re trying to answer?
Analysis of Qualitative Studies

All the stuff we talked about last week isn’t just for need-finding studies!

As always, you don’t get to just pull out a few quotes and call it a day. You need a data analysis approach that’s appropriate for your data.
Final Project Chat!
IRB...another reminder!

If you intend to publish the findings from your interactions with users/potential users, bear in mind that the IRB approval process typically takes 2 weeks.

- Submit at: https://eprotocol.berkeley.edu/userLogin.do
- For support:
  - https://cphs.berkeley.edu/eprotocol_faqs.html
  - Slack—your fellow students are experts!
  - Me

For figuring out if you need IRB approval: https://cphs.berkeley.edu/review.html
  - Scroll to “Activities that Generally Require Review” and “Activities that May Not Require Review”
No High-Risk Studies!

Even if you do not intend to publish the findings from your interactions with users/potential users, make sure you’ve carefully analyzed any risks to your participants. If you identify any risks higher than the risks of day-to-day computer use, please come talk to me.
Formative Study Design Activity

• Pick a need uncovered in last week’s need-finding HW.
• List three research questions you might want to answer in order to design a good tool intervention for addressing that need.
• Pick one! (Doesn’t have to be your favorite, just any RQ.)
• Take 5 minutes to brainstorm 3+ formative studies that would let you answer it.
• Which one do you think is likeliest to get the answer to the RQ?
• Turn to a partner. Share your ideas in turn.
• Do you think your partner’s idea is likely to answer the research question? What risks/threats do you see? Are there ways it might fail to answer the question? Share!
Prototyping + Design Critique Activity

(1) Choose a need that one or more members of your team identified from the need-finding HW.

(2) As a team, develop documents that walk viewers through three possible language, tool, or environment interventions that might address the chosen need.

- **Slides? Google docs? Webpages? Pen and paper drawings?** Up to you! Whatever you choose, you should be able to quickly present it to classmates from other teams.

- **“Walk viewers through?”** What will the interaction look like from the user’s perspective? What will the user see on their screen? What will they see at the start, at every intermediate step, at the end? What actions is the user taking? In the terminology of today’s reading, this will be closest to storyboarding.

- **How different should the three alternatives be?** This is up to you, but I suggest erring on the side of greater variety. You should be showing alternatives that you think are reasonable. But you’ll learn more from showing three very different alternatives than from showing slight variants on a single approach. So I encourage you to get weird!

(3) Pair up with another team. In session 1, Team A presents to Team B, and B gives feedback. In session 2, vice versa!