CSE 291 I: Usability of Programming Languages ("Programmers Are People Too")

Michael Coblenz
Today: Interviews (and Focus Groups)

Following "Research Methods in HCI" ch. 8
Why Interviews?

- For exploratory research
  - What problem should I solve?
  - How might I generate a good hypothesis?
- Depth
Considerations

- Time required
- Recall
- Interviewer-driven bias
- High analysis cost
Situations in Which To Interview

• Initial exploration of a new area
• Requirements elicitation
• Prototypes: evaluation and subjective reactions
Whom To Interview

- Consider all stakeholders
- Who are the stakeholders of a PL?
  - Programmers/software engineers
  - Architects
  - Technical writers
  - Managers
  - End users
  - QA
Interview Structure

- Fully structured interview: rigid script. Like a survey.
  - Can skip questions, but not really a conversation
  - Easier to analyze (answers are always in order)
- Semistructured: free to ask questions in a more conversational way (out of order)
- Unstructured interview: base on list of topics from interview guide
Structure and Skill

- Fully-structured: anyone can be the interviewer
- Unstructured: depends strongly on interviewer skill
- Semi-structured: depends somewhat on interviewer skill
- People almost always choose **semi-structured**.
Focus Groups

- Focus group: gather 5-7 or 8-12 participants for a group interview
- Pro: more participants, less experimenter time
- Discussions reveal similarities and differences
- Con: quiet people might not get heard
- Skill is needed to manage conversation
- Analysis can be tricky (interruptions, changes of speaker)
Demonstration: Writing Questions

- "A taco is a kind of sandwich, right?"
- "A taco isn't a kind of sandwich, is it?"
- Is a taco a sandwich?
- What food categories are the items in the picture in?
- What do you expect to see on the menu at a sandwich restaurant?
- What makes something a sandwich?
Demonstration (2)

• How do you feel about covariant return types?
  • Use terms your participant knows.

• When do you usually decide to start using the debugger?
  • Think of the last bug you fixed. What debugging strategies did you use?
Designing Questions

- Neutral: unbiased, nonjudgmental
- Simple
- Open-ended
- Speak their language
- Ask for demonstrations or recall of concrete events
Simple Questions

- "What were the strengths and weaknesses of the compiler and IDE?"
  - "What did you think of the compiler" & "What did you think of the IDE?"
Netural Questions

• "Did you like the language you used?"
  • -> "What did you think of the language?"

• "Why do you like this design?"
  • What if they didn't like the design?
Recording Data

- Write notes
- Rewrite and summarize after the interview
- Record audio & transcribe
- Screen capture
Rapport

• Be nonjudgmental — develop a poker face!

• Keep people comfortable. Water? Snacks?
Conducting the Interview

- Start with easy questions
- Listen!
- Provide opportunities to continue: "Is there anything else you wanted to tell me?"
- Ask for clarification when needed: "What exactly do you mean when you say...?"
Debriefing

- Give more background on your research (which you didn't reveal earlier to avoid bias)
Practice

• Research question: What strategies do people use to debug non-terminating code?

• Goal: develop debugging tools that might make this easier.

Reminder:

• Neutral: unbiased, nonjudgmental
• Simple
• Open-ended
• Speak their language
• Ask for recall of concrete events
Interview Protocol

• A plan!

• For you AND for replication

• Mostly, just a list of questions, in order, and any other materials needed.
Office Hours

- Come to office hours!
- My office today 4-5 PM