USER-CENTERED PROGRAMMING LANGUAGE DESIGN

Michael Coblenz
WHY?

• Tell me what the following program does…
WHAT DOES THIS DO?
(this is part of Firefox)

```
0000000100000e30 movq %rdi, %rax
0000000100000e33 movq (%rax), %rdi
0000000100000e36 movq $0x0, (%rax)
0000000100000e3d testq %rdi, %rdi
0000000100000e40 je 0x100000e48
0000000100000e42 movq (%rdi), %rax
0000000100000e45 jmpq *0x10(%rax)
0000000100000e48 retq
0000000100000e49 nopl (%rax)
0000000100000e50 pushq %rbp
0000000100000e51 pushq %r15
0000000100000e53 pushq %r14
0000000100000e55 pushq %r13
0000000100000e57 pushq %r12
0000000100000e59 pushq %rbx
0000000100000e5a subq $0x418, %rsp
0000000100000e61 movq %rdx, %r13
0000000100000e64 movq %rsi, %r14
0000000100000e67 movl %edi, %r15d
0000000100000e6a movq 0x119f(%rip), %rax
0000000100000e71 movq (%rax), %rax
0000000100000e74 movq %rax, 0x410(%rsp)
0000000100000e7c movl $0x1, %edi
0000000100000e81 callq 0x100001ad6
0000000100000e86 movq %rax, %r12
0000000100000e89 callq 0x100001b90
0000000100000e8e testq %rax, %rax
0000000100000e91 je 0x10000102a
0000000100000e9a callq 0x100001b8a
0000000100000e9f movq %rax, %rbx
0000000100000ea2 testq %rbx, %rbx
0000000100000eo5 je 0x10000102a
0000000100000eeb leaq 0x10(%rsp), %r15
0000000100000ef2 callq 0x100001b96
0000000100000f34 leaq 0x10(%rsp), %rdi
```

```
0000000100000eb2 movl $0x400, %ecx
0000000100000eb7 movq %rbx, %rdi
0000000100000ee9 callq 0x100001b9c
0000000100000ef3 movq %rax, %rdi
0000000100000ef7 movq %rax, %rbx
0000000100000efb testb %al, %al
0000000100000f00 testb %dl, %dl
0000000100000f04 jmp 0x100001ee0
0000000100000f06 nopl %cs:(%rax,%rax)
0000000100000f10 cmpb $0x2e, %dl
0000000100000f1c jne 0x100001f00
0000000100000f23 movq %rax, 0x1235(%rip)
0000000100000f27 testq %rdi, %rdi
0000000100000f2c callq 0x100001b96
0000000100000f34 leaq 0x10(%rsp), %rdi
0000000100000f39 callq 0x100001ae6
0000000100000f3e movq %rax, %rbx
0000000100000f41 testq %rbx, %rbx
0000000100000f44 je 0x10000102a
0000000100000f4a leaq 0x10(%rsp), %rdi
0000000100000f4f movq %rbx, %rsi
0000000100000f52 callq 0x100001b90
0000000100000f57 movq 0x10(%rsp), %rdi
0000000100000f5c movq $0x0, 0x18(%rsp)
0000000100000f65 movq 0x123c(%rip), %rax
0000000100000f6a movq %rsi, %rdi
0000000100000f6f callq 0x100001680
0000000100000f73 movq 0x10(%rsp), %rdi
0000000100000f7c movq %rax, %rcx
0000000100000f82 callq *0x10(%rcx)
0000000100000f85 movq 0x10(%rsp), %rdi
0000000100000f8a movq $0x8, 0x10(%rsp)
0000000100000f93 testq %rdi, %rdi
0000000100000f96 je 0x100000f9e
0000000100000f99 movq (%rdi), %rax
0000000100000fa5 testq %rdi, %rdi
0000000100000fa8 je 0x10000107b
0000000100000fba movq 0x10000107b
0000000100000fbae movq (%rdi), %rax
0000000100000fb1 callq *0x18(%rax)
0000000100000fb4 movq %rbx, %rdi
```
LANGUAGES ARE INTERFACES

• PLs are interfaces for *humans* to use to write programs.

• Therefore they should be subject to the principles and methods of HCI.

• For next few lectures:

  • What are these principles and methods?

  • How can we apply them to programming languages?
TODAY

• What is usability?

• How might I measure usability?

• (Next time: more ways of assessing usability)
WHAT IS USABILITY?

• "The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use." (ISO 9241-11, Ergonomics of human-system interaction)
USABILITY

https://pixabay.com/vectors/speedometer-kilometers-dashboard-309118/
USABILITY

• Who are the users?
• What are the users trying to do?
• In what context are they doing it?
• How well do they do it?
• How much work does it take them to do it?
• Do they like it?
WHO ARE THE USERS?

• Scope your target audience
  • Education
  • Experience
  • Knowledge
  • Skills
  • Work context
PERSONAS

• Help you think like the user
• Build empathy
Charlie is in her late 20s to mid 30s. She has a Bachelor’s degree but not necessarily in IT. She’s a self-taught developer. Her coding is unconventional and she mixes genius lines with simple errors. She seeks to reinvent her software development career but the how is still unclear. Charlie has a family, which makes financial stability and work-life balance essential. She’s new to the industry and thus looks for a company that offers a supportive, people-oriented environment, where she can learn and improve her skills.

https://www.meistertask.com/blog/there-are-four-archetypal-developer-personas-which-one-are-you/
ROBIN

Robin is in his mid 20s and has completed his formal education, such as a Bachelor's degree in computer science. He is probably on his second or third job but has reached the ceiling in his current job, as in, he has learnt a lot and gained experience but would be keen on taking the next step to further his career. Even though he probably hasn’t taken any steps to find a new job (applied), he is on the lookout for something challenging as well as purposeful. In his current role, he can be found working in a specialized programming area (front-end/back-end/mobile). On a personal level, he is probably in a relationship, he is also quite introverted and self-aware. He enjoys working on complicated tasks and really wants to be involved and feel a part of the company. He values transparency and is happy working with inspiring leaders. He’s keen to know what is going on and where the company is headed. Salary isn’t his top priority (as long as it is not too far below average). Instead, Robin appreciates non-financial rewards, especially those that make him feel valued for his work.
MICROSOFT PERSONAS (CLARKE)

THE SYSTEMATIC DEVELOPER
Writes code defensively. Does everything they can to protect their code from unstable and untrustworthy processes running in parallel with their code. Develops a deep understanding of a technology before using it. Prides themselves on building elegant solutions.

THE PRAGMATIC DEVELOPER
Writes code methodically. Develops a sufficient understanding of a technology to enable them to use it. Prides themselves on building robust applications.

THE OPPORTUNISTIC DEVELOPER
Writes code in an exploratory fashion. Develops a sufficient understanding of a technology to understand how it can solve a business problem. Prides themselves on solving business problems.
GOALS

• What does the user want to achieve?

• Start vague, and then drill down

• What does a C programmer want to achieve?
  • Fix a bug — what kind?
  • Implement something — what?
  • Understand code — for what purpose?
CONTEXT

• Starting from scratch, maintaining a system, or legacy code?

• Large team, or lone developer?

• Beginning of project, or near shipping (risk-averseness)?

• One-off, or repeated task?
HOW WELL DO THEY DO IT?

- Count and describe bugs/errors
- How readable is their code?
  - What does "readable" mean, anyway?
- What fraction of target population can do it?
HOW HARD WAS IT?

• What fraction of participants succeeded?

• How long does it take?

• What obstacles did they encounter along the way?

• Other measures?
  • Galvanic skin response (emotional response)
  • EEG (cognitive load)
DO THEY LIKE IT?

• Of course they do, if it's YOUR system!

• "We find that respondents are about 2.5× more likely to prefer a technological artifact they believe to be developed by the interviewer, even when the alternative is identical." [Dell et al., CHI 2012]

• "When the interviewer is a foreign researcher requiring a translator, the bias towards the interviewer’s artifact increases to 5×."
SO WHAT IF THEY LIKE IT?

- Maybe there are factors you didn't measure
- Adoption
- "Why not?"
- But maybe you don't care!
VARIETIES OF USABILITY

• My "more usable" system might be better because:
  • Learnability: it's easier/faster to learn
  • Task performance: people finish tasks faster or more people finish tasks
  • Audience: a new kind of person can do the task
  • (Not an exhaustive list)
YOUR TURN

• Identify a usability question YOU have about a COMMON PL.

• With a partner.

• Share afterward.

"The extent to which a product can be used by specified users to achieve specified goals with effectiveness, efficiency and satisfaction in a specified context of use."
CATEGORIES OF METHODS

• Qualitative methods
  • Focus is on depth of data
  • Does not imply no quantities

• Quantitative methods
  • Focus is on statistical analysis of data
GENERATING HYPOTHESES
NOT JUST ANY HYPOTHESES…

• Want to only test hypotheses that are probably true.

• You can publish a paper even if all you have is a hypothesis!

  • (if it is well-justified)

• And what if your 💡 is empty?
STAGES

• I don't know what I'm doing.
  • What problems are there to solve?
  • What hypotheses are worth testing?
• I have a tool. Let's make it better.
• I have a tool. Let's try to show that it IS better.

Qualitative studies
QUALITATIVE STUDIES

• Want to understand something we don't understand yet.
  • What problems do factory workers have?
  • What is it like to write code for Indy 500 cars?
• What usability problems do people have when they use my "awesome" system?
USABILITY STUDIES

• Give people tasks and observe what happens.
• NOT experiments
• NOT controlled
• NOT comparative
• Just want to see what problems people encounter.
USABILITY STUDIES CAN SHOW

• X% of my participants completed the task in 30 minutes.
• Participants encountered the following problems…
• Only participants who knew X were able to do the task.
USABILITY STUDIES CANNOT SHOW

• My system is better than an existing system.
If you want to argue your results generalize to X, then ideally you should sample from X.

Plan B: argue X is similar to the population you sampled from.

Examples?
HOW TO RUN A STUDY

• We have submitted a proposal to IRB on your behalf.

• We can't promise IRB will approve it as is!
RECRUITMENT

• Flyers
• Emails
• Social network
• Buy ads
• The street

INCENTIVES

• $$$ (in person, MTurk)
• Desire to contribute to science / help you out
• Food
• Fame (leaderboard)

• Rare experience
• Learning opportunity
• Distraction from work
• Credit
TARGETS

• Programmers
• Architects
• Code reviewers
• Testers
• Security teams

• Designers
• Domain experts
• Tool creators
• Users
• Requirements engineers
ETHICS

• What if incentive is too high?
  • IRB reviews incentives
• What if incentive is too low?
• What if recruitment is misleading?
  • IRB reviews recruitment materials
PARTICIPANT PRE-SCREENING

• Can issue a pre-test to avoid wasting time on unqualified participants.

• IRB likes to review the pre-test.

• Issues:
  • How will you incentivize people to take the test?
  • Can you use the test results in your research?