



building developer tools your coworkers won't hate

adam barber (@necrobuffalo)
amy nguyen (@amyngyn)
devopsdays silicon valley | may 17, 2018

who are we



Amy Nguyen

- observability engineer at stripe
- excited about user experience design and education for developer tools
- @amyngyn
- amynguyen.net

who are we

Adam Barber

- software engineer at nylas
- excited about making usable software
- @necrobuffalo
- blog.taron.rip



agenda

- motivation
- how to do user research
- UX design principles
 - command line tools
 - data analysis tools
 - build tools



why care?

- bad UX causes incidents.
incidents cost money.
- bad UX wastes time.
time costs money.
- bad UX creates hostility.
hostility costs money.





how to do user research

- talk to people
 - one-on-one
 - focus groups

how to do user research

- ask the right questions
 - don't ask questions that confirm what you were expecting
 - dig deep, don't accept the first answer

how to do user research

- communicate as you follow-up
 - build trust
 - confirm your solutions are the right solutions



how to do user research

- talk to people
- ask the right questions
- communicate as you follow-up



command line tools

does anyone even know what that flag does?

be consistent

```
% man -h
```

```
Usage: man [OPTION] [SECTION] PAGE...
```

```
[...]
```



be consistent

```
% less -h
```

Value is required after -h
(--max-back-scroll)



be consistent

- match arguments to your other tools
- match arguments to historical usage



don't break existing tools

- have integration tests for your internal tools
- design with existing tools in mind



preview as the default

- if your tool makes critical changes to infrastructure, don't allow users to make those changes accidentally

```
./run-script --confirm
```

- provide diffs of what will change





data analysis tools

does that data really mean what i think it means?

rapid testing

- users need a REPL or other way of quickly validating ideas



discoverability

- autocomplete
- GUI or other menus for showing all the options



correctness

- syntax errors and highlighting
- explanations for syntax errors



explorability

- back button
- refresh button
- RESTful links
- open in new tabs



data analysis tool design

- rapid testing
- discoverability
- correctness
- explorability



build tools

i don't know who broke the build
because i don't know how to find the build



navigation

- anticipate the most common actions
 - find my latest build
 - locate failing output
 - rebuild



fail fast

- if something is failing, don't wait to finish before showing the result to the user



helpful messages at the right time

- remind users to lint and unit test before kicking off a long build process



helpful messages at the right time

- remind users to lint and unit test before kicking off a long build process
- suggest corrections for bad input



summary

- communicate early and often with users
- imagine the context in which users will use your tools
- user experience and good design matters even for infrastructure tools



thanks!

