

Retour aux sources pour un code de qualité

Frédéric Harper
Senior Developer Advocate
DigitalOcean

@fharper
fred.dev



— People who can't code: Coding is fucking hard!

- People who can't code: Coding is fucking hard!
- People who don't try at all: Coding is fucking hard!

- People who can't code: Coding is fucking hard!
- People who don't try at all: Coding is fucking hard!
 - Teachers: Coding is fucking hard!

- People who can't code: Coding is fucking hard!
- People who don't try at all: Coding is fucking hard!
 - Teachers: Coding is fucking hard!
 - Novice Developers: Coding is fucking hard!

- People who can't code: Coding is fucking hard!
- People who don't try at all: Coding is fucking hard!
 - Teachers: Coding is fucking hard!
 - Novice Developers: Coding is fucking hard!
 - Pro Developers: Coding is fucking hard!

- People who can't code: Coding is fucking hard!
- People who don't try at all: Coding is fucking hard!
 - Teachers: Coding is fucking hard!
 - Novice Developers: Coding is fucking hard!
 - Pro Developers: Coding is fucking hard!
 - Famous Developers: Coding is fucking hard!

- People who can't code: Coding is fucking hard!
- People who don't try at all: Coding is fucking hard!
 - Teachers: Coding is fucking hard!
- Novice Developers: Coding is fucking hard!
- Pro Developers: Coding is fucking hard!
- Famous Developers: Coding is fucking hard!
- Business People: Coding is easy!

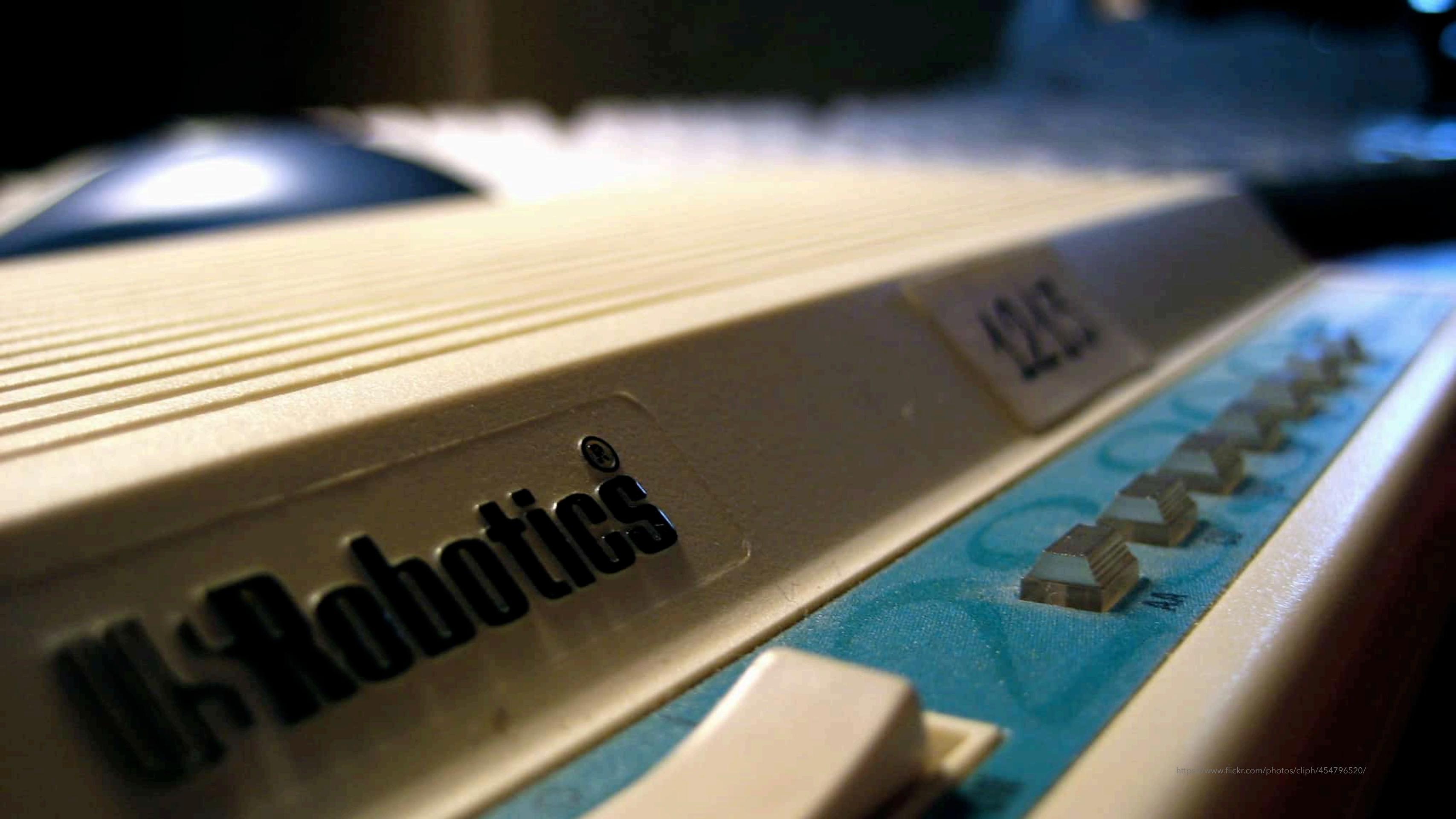
- People who can't code: Coding is fucking hard!
- People who don't try at all: Coding is fucking hard!
 - Teachers: Coding is fucking hard!
 - Novice Developers: Coding is fucking hard!
 - Pro Developers: Coding is fucking hard!
 - Famous Developers: Coding is fucking hard!
 - **Business People: Coding is easy!**



<https://giphy.com/gifs/stress-i-need-a-drink-brain-explode-2rqEdFlkMzXmo/>



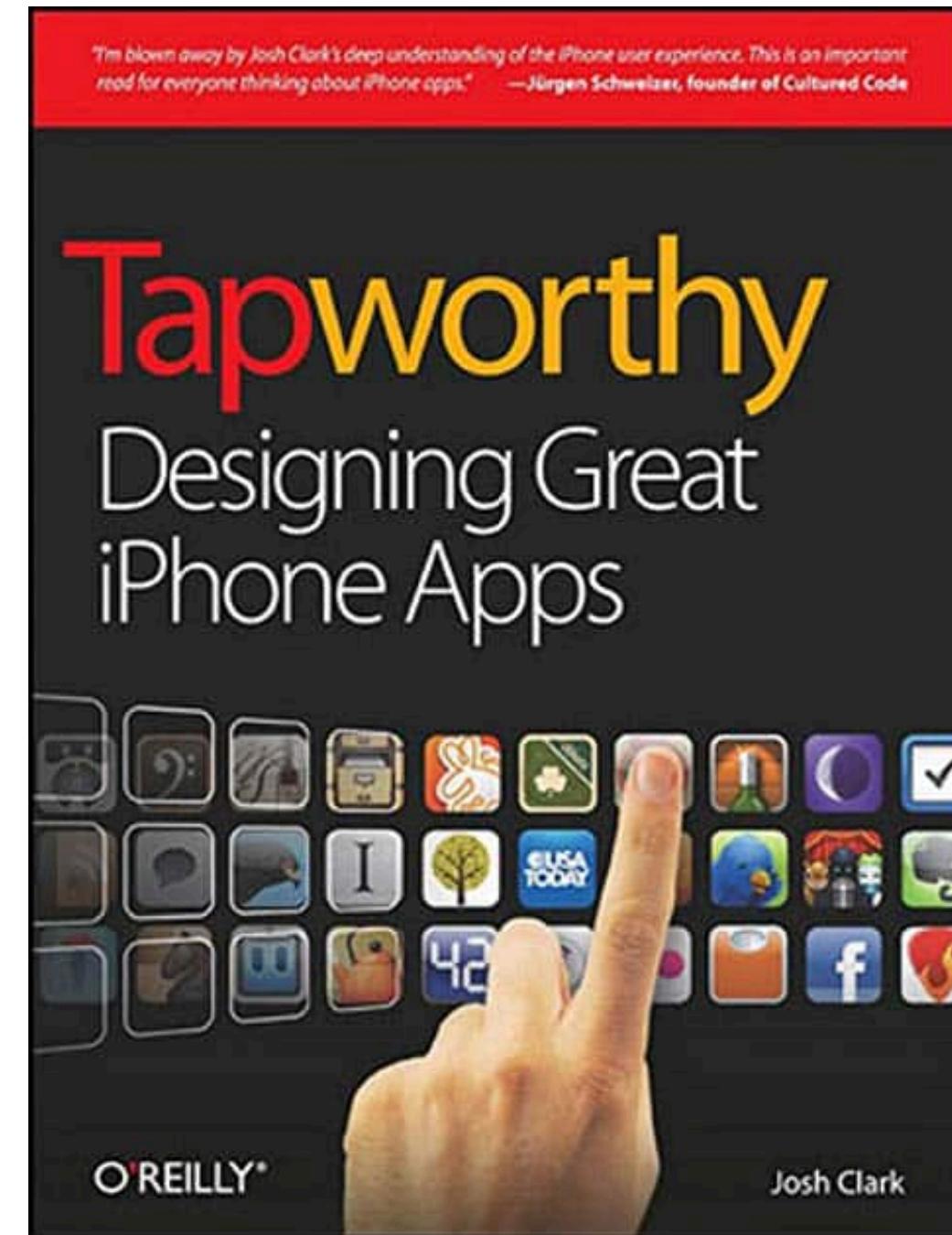






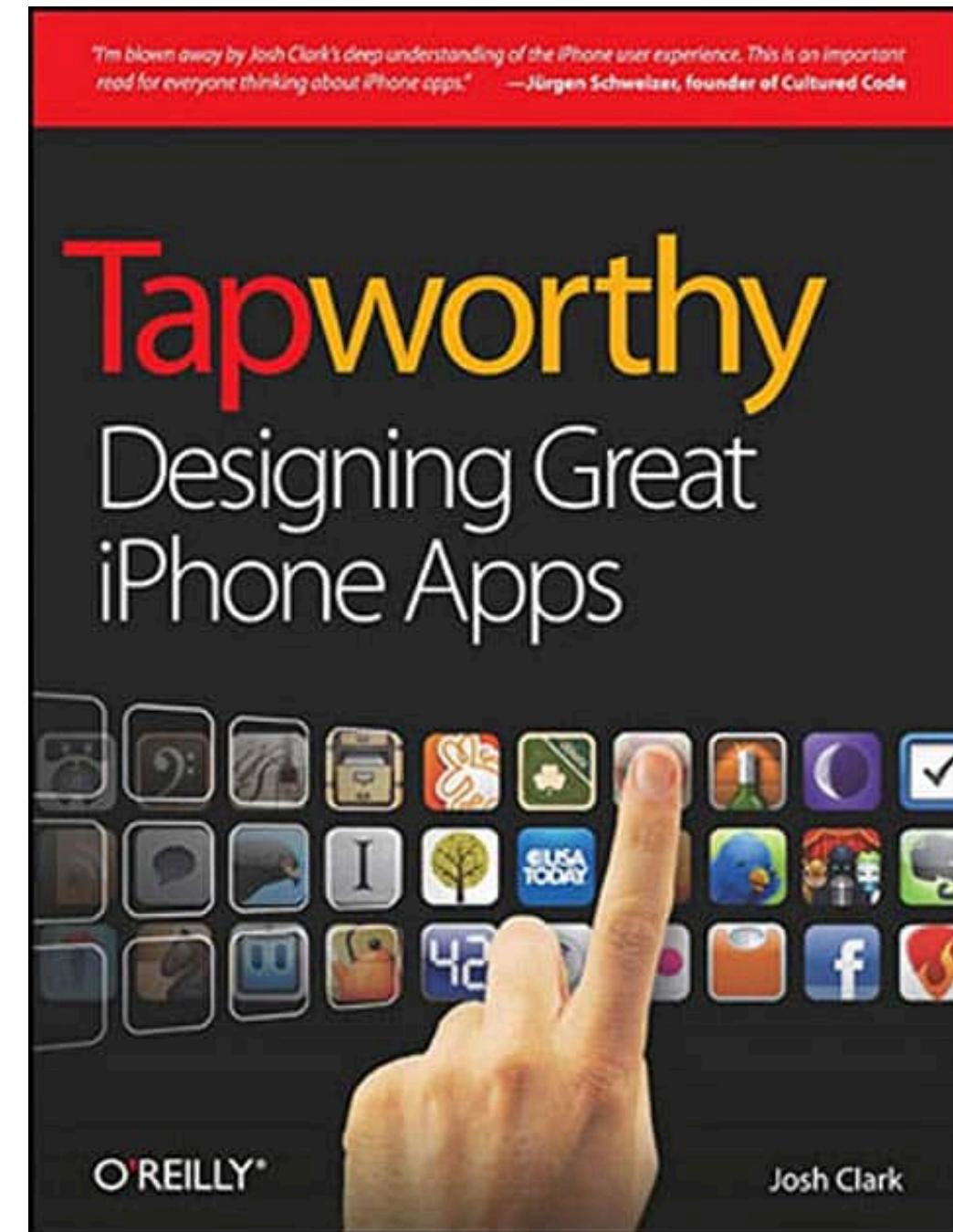
architect

motivations



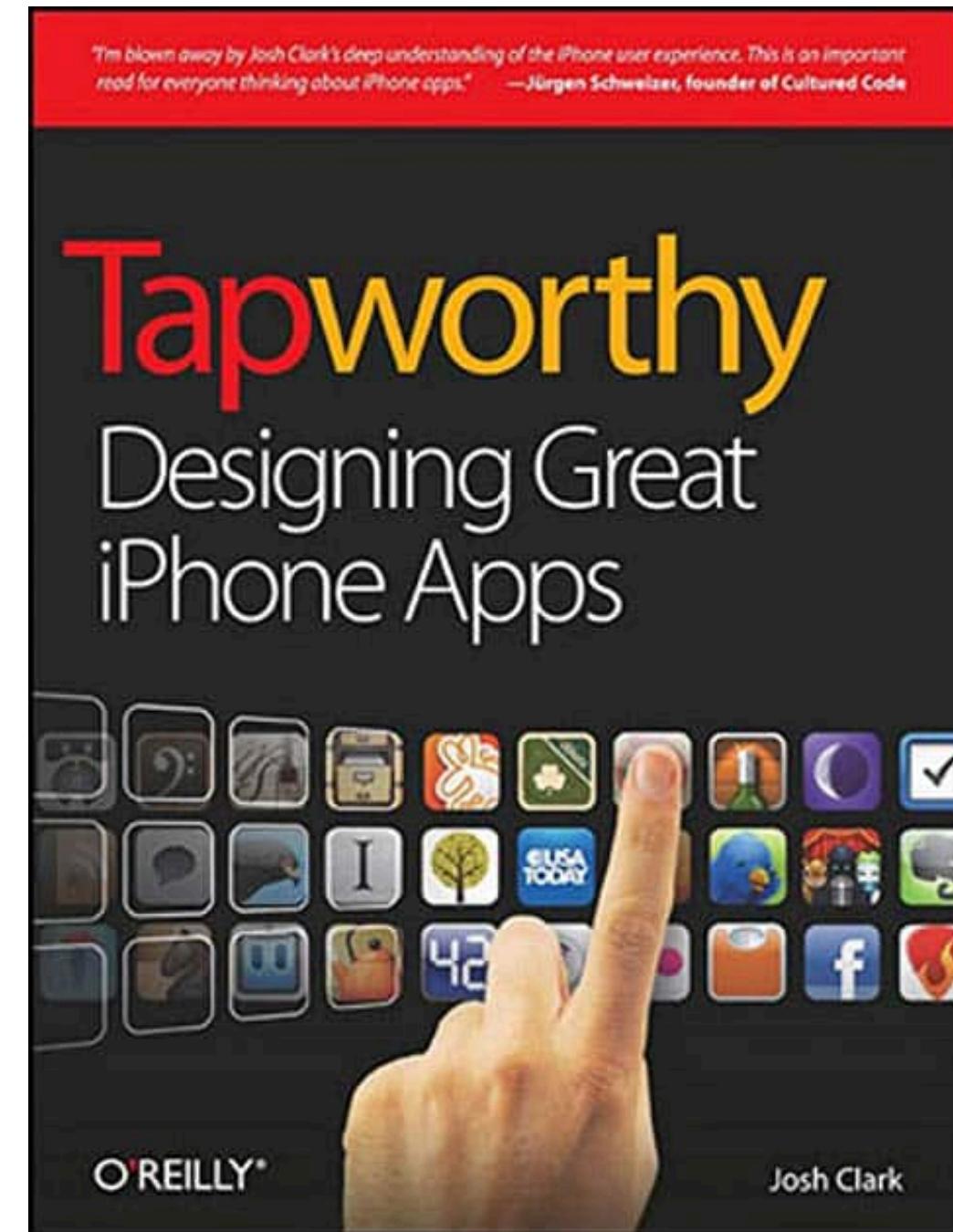
motivations

1. I'm (micro)tasking



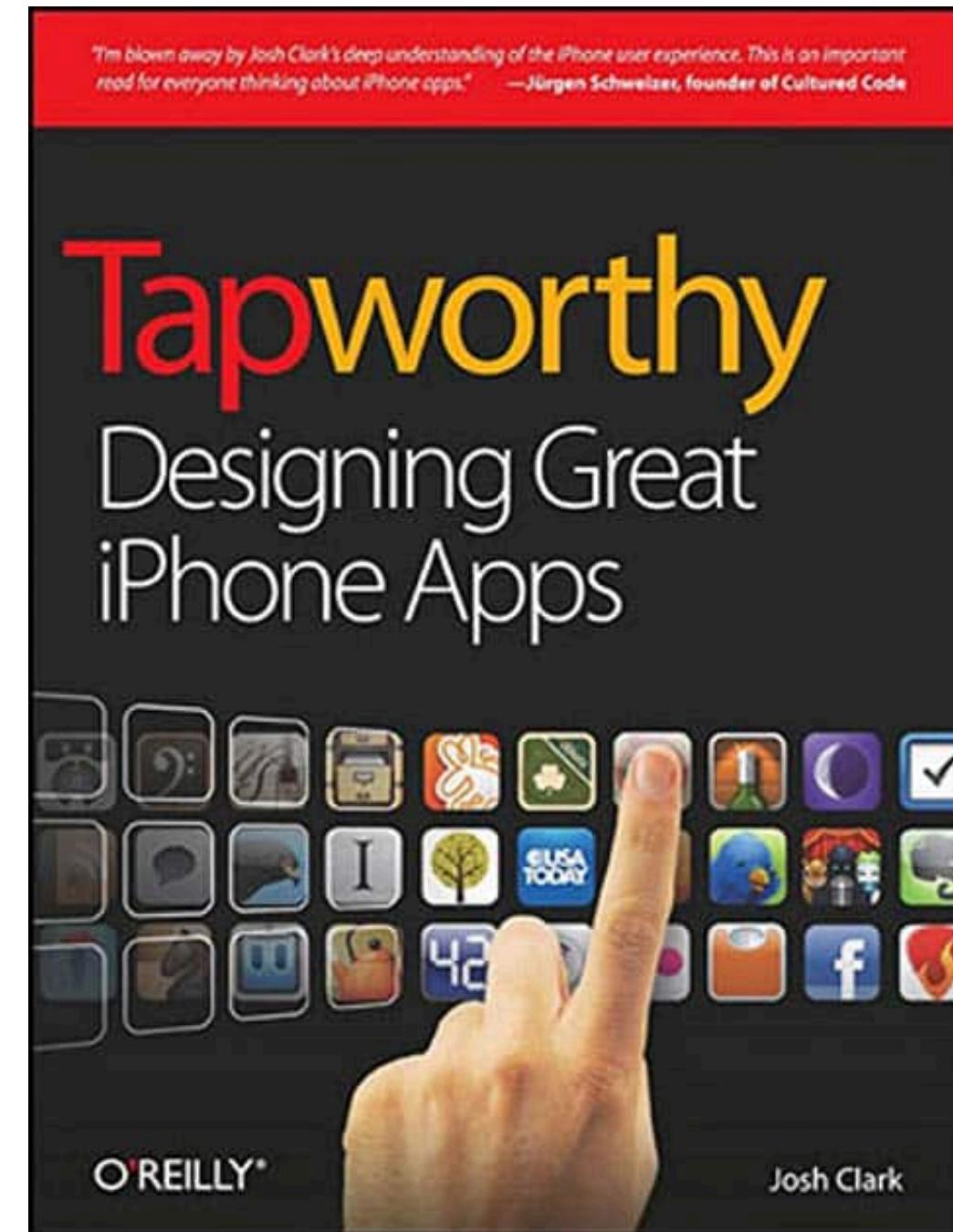
motivations

1. I'm (micro)tasking
2. I'm local

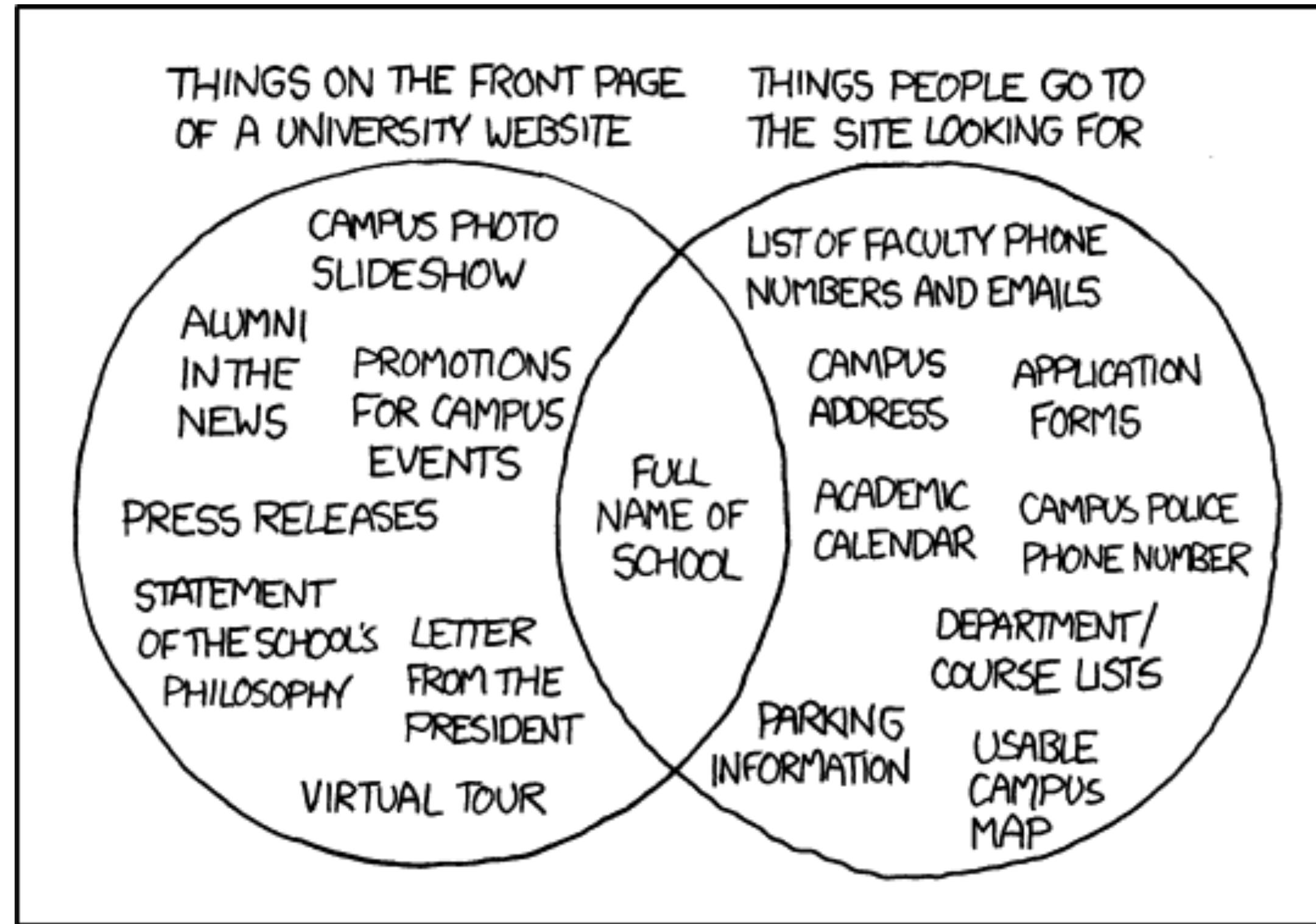


motivations

1. I'm (micro)tasking
2. I'm local
3. I'm bored



design



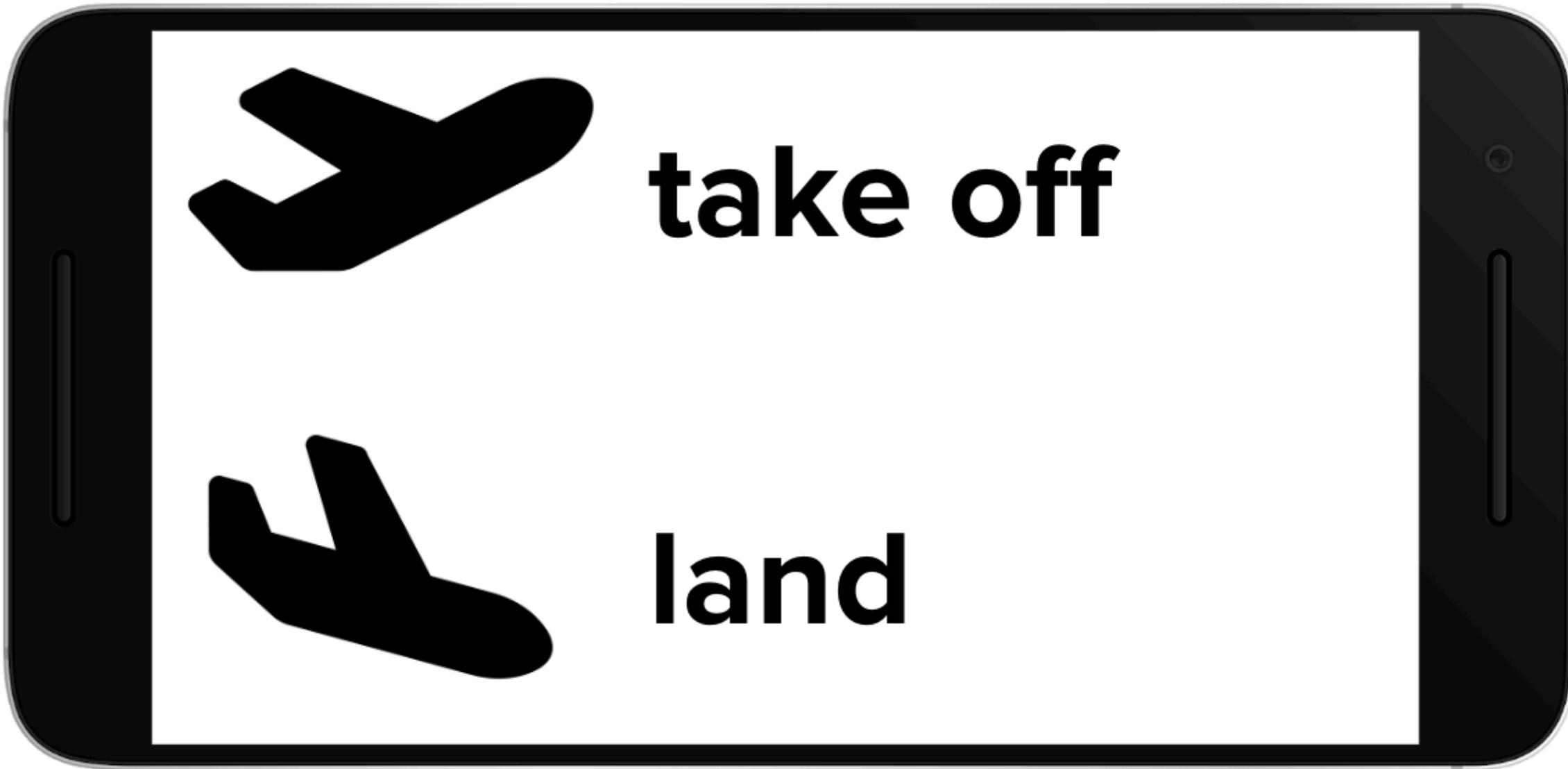
mobile first



mobile first







comfort zone



Fitt's law

$$T = a + b \times \log_2(1 + D/W)$$

T: Average time taken to complete the movement

a: Start/stop time of the device

b: Inherent speed of the device

D: Distance from the starting point to target center

W: Width of target, measured along axis of motion

Fitt's law translated

the **bigger** and
closer a target is,
the **easier** it is to hit.

optimize

JS, CSS & HTML

demo

HTTP requests

HTTP requests

- avoid or minimize 3xx redirections

HTTP requests

- avoid or minimize 3xx redirections
- GZIP encoding (htaccess, web.config)

HTTP requests

- avoid or minimize 3xx redirections
- GZIP encoding (htaccess, web.config)
- image sprites (when it make sense)

HTTP requests

- avoid or minimize 3xx redirections
- GZIP encoding (htaccess, web.config)
 - image sprites (when it make sense)
- use CDN (Content Distribution Network)

HTTP requests

- avoid or minimize 3xx redirections
- GZIP encoding (htaccess, web.config)
 - image sprites (when it make sense)
- use CDN (Content Distribution Network)
- configure the HTTP Cache Headers (Apache)

HTTP requests

- avoid or minimize 3xx redirections
- GZIP encoding (htaccess, web.config)
 - image sprites (when it make sense)
- use CDN (Content Distribution Network)
- configure the HTTP Cache Headers (Apache)
- configure HTTP Expires Response Header (IIS)

HTTP requests

- avoid or minimize 3xx redirections
- GZIP encoding (htaccess, web.config)
 - image sprites (when it make sense)
- use CDN (Content Distribution Network)
- configure the HTTP Cache Headers (Apache)
- configure HTTP Expires Response Header (IIS)
 - cache the content

webhint

demo

Images

Images

- use native image resolution (original width, height)

Images

- use native image resolution (original width, height)
 - use the right image format (PNG, JPEG...)

Images

- use native image resolution (original width, height)
 - use the right image format (PNG, JPEG...)
 - use image preview for videos

Images

- use native image resolution (original width, height)
 - use the right image format (PNG, JPEG...)
 - use image preview for videos
 - compress your images

imagemin
demo

tests

tests

- create tests: unit tests, integration tests...

tests

- create tests: unit tests, integration tests...
- test yourself, be the dummest user you can be

tests

- create tests: unit tests, integration tests...
- test yourself, be the dummest user you can be
- use framework like Mocha or QUnit for JavaScript/Node.js

tests

- create tests: unit tests, integration tests...
- test yourself, be the dummiest user you can be
- use framework like Mocha or QUnit for JavaScript/Node.js
 - test, test and test

misc js

misc js

- use 'strict' mode

misc js

- use 'strict' mode
- avoid creating new object when it's possible

misc js

- use 'strict' mode
- avoid creating new object when it's possible
- load JavaScript files at the end of the page

misc js

- use 'strict' mode
- avoid creating new object when it's possible
- load JavaScript files at the end of the page
- asynchronously load scripts (async) and fetch data

misc js

- use 'strict' mode
- avoid creating new object when it's possible
- load JavaScript files at the end of the page
- asynchronously load scripts (async) and fetch data
 - JSON is faster than XML

misc js

- use 'strict' mode
- avoid creating new object when it's possible
- load JavaScript files at the end of the page
- asynchronously load scripts (async) and fetch data
 - JSON is faster than XML
- don't scale images in drawImage (cache in offscreen canvas)

misc js

- use 'strict' mode
- avoid creating new object when it's possible
- load JavaScript files at the end of the page
- asynchronously load scripts (async) and fetch data
 - JSON is faster than XML
- don't scale images in drawImage (cache in offscreen canvas)
- use WebGL instead of Context 2D (free up CPU for GPU)

misc

misc

- don't fix it if it's not broken (human aversion to change)

misc

- don't fix it if it's not broken (human aversion to change)
 - you don't always need a framework or a library

misc

- don't fix it if it's not broken (human aversion to change)
 - you don't always need a framework or a library
 - put as much logic as you can on the server-side

misc

- don't fix it if it's not broken (human aversion to change)
 - you don't always need a framework or a library
 - put as much logic as you can on the server-side
 - think accessibility (a11y, pa11y, axe tool & hint)

misc

- don't fix it if it's not broken (human aversion to change)
 - you don't always need a framework or a library
 - put as much logic as you can on the server-side
 - think accessibility (a11y, pa11y, axe tool & hint)
 - SSL all the things

misc

- don't fix it if it's not broken (human aversion to change)
 - you don't always need a framework or a library
 - put as much logic as you can on the server-side
 - think accessibility (a11y, pa11y, axe tool & hint)
 - SSL all the things
 - don't fix it if it's not broken

the extra mile

security



OWASP

CHEAT SHEET
SERIES PROJECT

Life is too short • AppSec is tough • Cheat!

snyk & npm

demo

in the end

the philosophy

the philosophy

1. Insulate us from the complexity

the philosophy

1. Insulate us from the complexity
2. Make us accomplish our goals faster & securely

the philosophy

1. Insulate us from the complexity
2. Make us accomplish our goals faster & securely
3. Help us be awesome in the moment!

resources

slides

<https://speaking.fred.dev/>

recording

<https://www.youtube.com/user/fredericharper>



Frédéric Harper

fred@do.co
fred.dev