

Happiness through simplicity

Why simplicity is important.

Ways to make web apps simpler.

What I do

(linkedin pic)

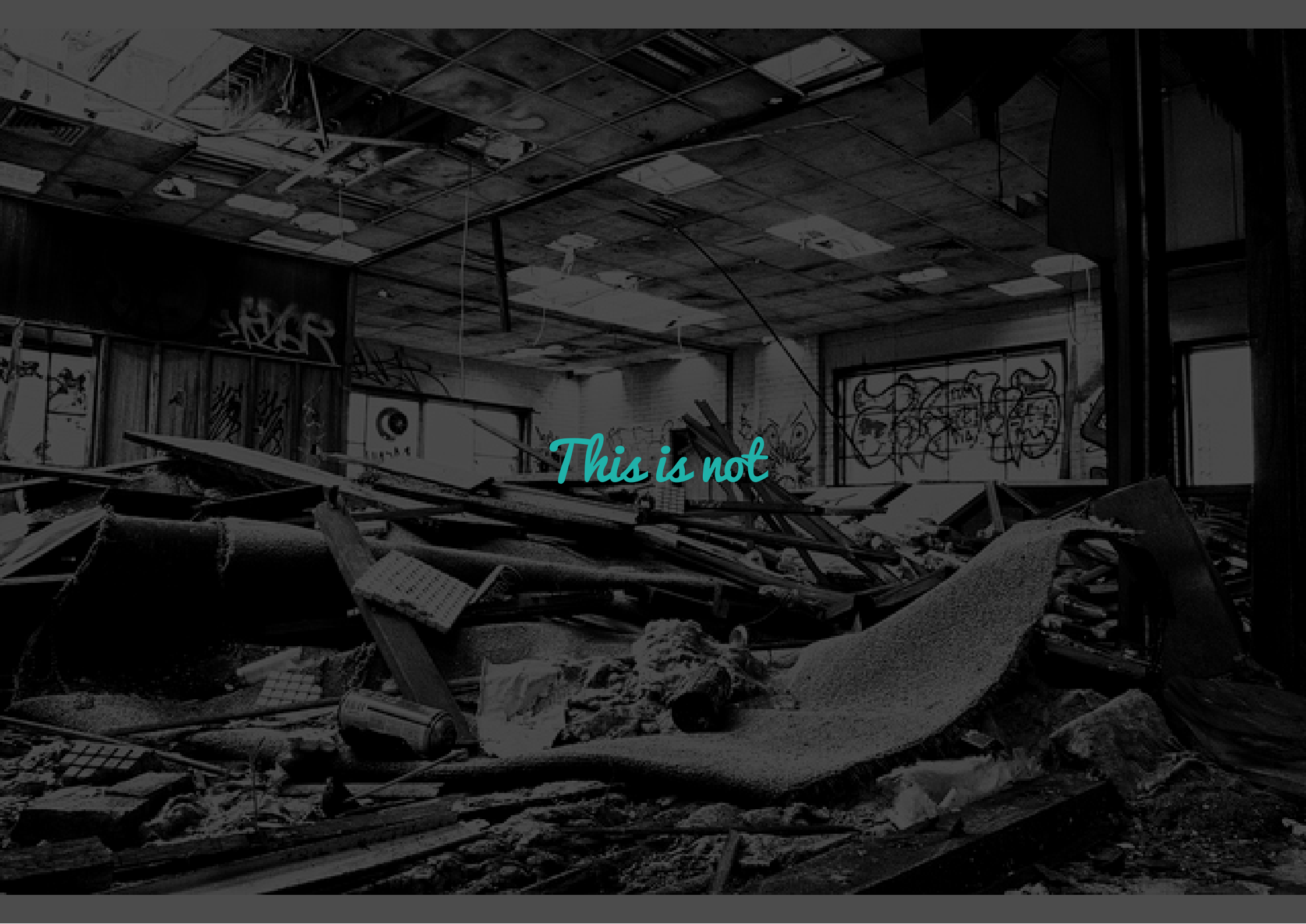
A black and white photograph of two women running towards the camera. The woman on the left has dark, curly hair and is wearing a light-colored short-sleeved top with a Peter Pan collar and a dark skirt. The woman on the right has blonde, curly hair and is wearing a light-colored long-sleeved blouse with a large collar and a dark skirt. Both women are smiling broadly and appear to be in motion. The background is slightly blurred, showing what looks like a building with windows.

Keeping people happy and moving ahead

*A big part of developer happiness is feeling like you're getting
somewhere.*

The image is a grayscale collage consisting of four vertical panels. Each panel shows a different perspective of the Eiffel Tower. The leftmost panel shows the base and lower legs of the tower. The second panel shows the tower rising from the base. The third panel shows the tower from a low angle, looking up. The rightmost panel shows the tower from a side angle, reaching towards the top. Overlaid in the center of the collage is the text "This is nice to work on" in a light blue, cursive font.

This is nice to work on



This is not

Design

"The art or action of conceiving of and producing a plan of something."

OED



*"A designer knows he has achieved perfection not when there is nothing left to add,
but when there is nothing left to take away."*

Antoine de Saint-Exupery



"Refactor on first paste."

Luke Benstead (@kazade)

Good code does what it needs to and not more

Free of **dead code**. If you're not sure, delete it, git revert will forgive you.

Free of **duplication**, so you can change something once.

Free of **unnecessary things**, so they don't need to be updated and maintained or changed.

That includes **unnecessary good things**.

API client

Repeated: credentials, endpoint, content-type etc.

Simplify: export a single function, that takes these options and stores them in a closure, exposes methods that use them.

Benefit: smaller code inside closure, smaller function signatures. Easier testing.

This is a type of Dependency injection

Realtime

Repeated paths - REST API and websocket

Additional complexity to support these unneeded duplex communications

Simplify:

HTML5 SSE

Entire API now in REST/HTTP

Single mechanism to get something from the server

Less code

Rate limiting

Third party API

Resticts connections aross all endpoints to a certain amount per minute

Writing rate limit code and unit tests for this would be doable

Simplify: existing library

Found `node-rate-limit`

Function signature of module was 1:1 match with requirements

Benefit: problem fixed, no significant new code

We use libraries for DOM. Use them for modals, state machines, statistics, and AJAX loaders.

Data Binding

HTML template, some data, somewhere in the document.

Simplify:

ractive.js

Function signature of module was 1:1 match with problem

Benefit: things we didn't consider

DOM complexity

Needed: button, inside a header, inside a nav (header only for vertical centering)

7 levels deep,

one JS global

styling classes in the DOM

things like nav class="nav"

Simplify: hand code

Block element layout was a near* 1:1 match match with design

All elements styled in SASS

All elements styled through a single class name. The button in the nav are styles through `nav button`

* one extra element for vertical centering

JS itself

`console.log()`, `document.querySelectorAll()`, etc.

Solution: in your own code, write all the globals you want and bind them to the objects.

Nobody ever types `console`.

Work practice complexity

Ticket management system (JIRA?) - images attached to ticket were downloaded, 10 images = 20 clicks to see them all.

Simplify: inline every image attachment with a Chrome extension.

In Summary

Unnecessary complexity is not only poor engineering, it's also a drag.

Actively think about simplicity

Keep moving, keep happy.

Thanks!

@mikemaccanaon Twitter

Keyboard Interaction

Powered by [bespoke-keys](#)

Touch Interaction

Powered by [bespoke-touch](#)

Responsive Slides

Powered by [bespoke-scale](#)

Bullet List

Powered by [bespoke-bullets](#)

Bullet 1

Bullet 2

Bullet 3

Syntax Highlighting

Powered by [Lea Verou's Prism](#)

```
bespoke.from('article', [  
  classes(),  
  keys(),  
  touch(),  
  bullets('li, .bullet')  
]);
```

Named Route

Powered by [bespoke-hash](#)

Look up! This route has been named with a '*data-bespoke-hash*' attribute.

Emphatic Slide!

Powered by [bespoke-state](#)

Just the beginning...

Edit [Jade](#) markup in 'src/index.jade'

Edit [Stylus](#) styles in 'src/styles/main.styl'

Edit JavaScript in 'src/scripts/main.js'

Enjoy! :) -@markdalglish