To framework or not?

Peter Fisher BSC MBSC

Freelance web & mobile applications developer

Host of the How To Code Well channel

youtube.com/howtocodewell



Author of Docker In Motion

From Manning Publications



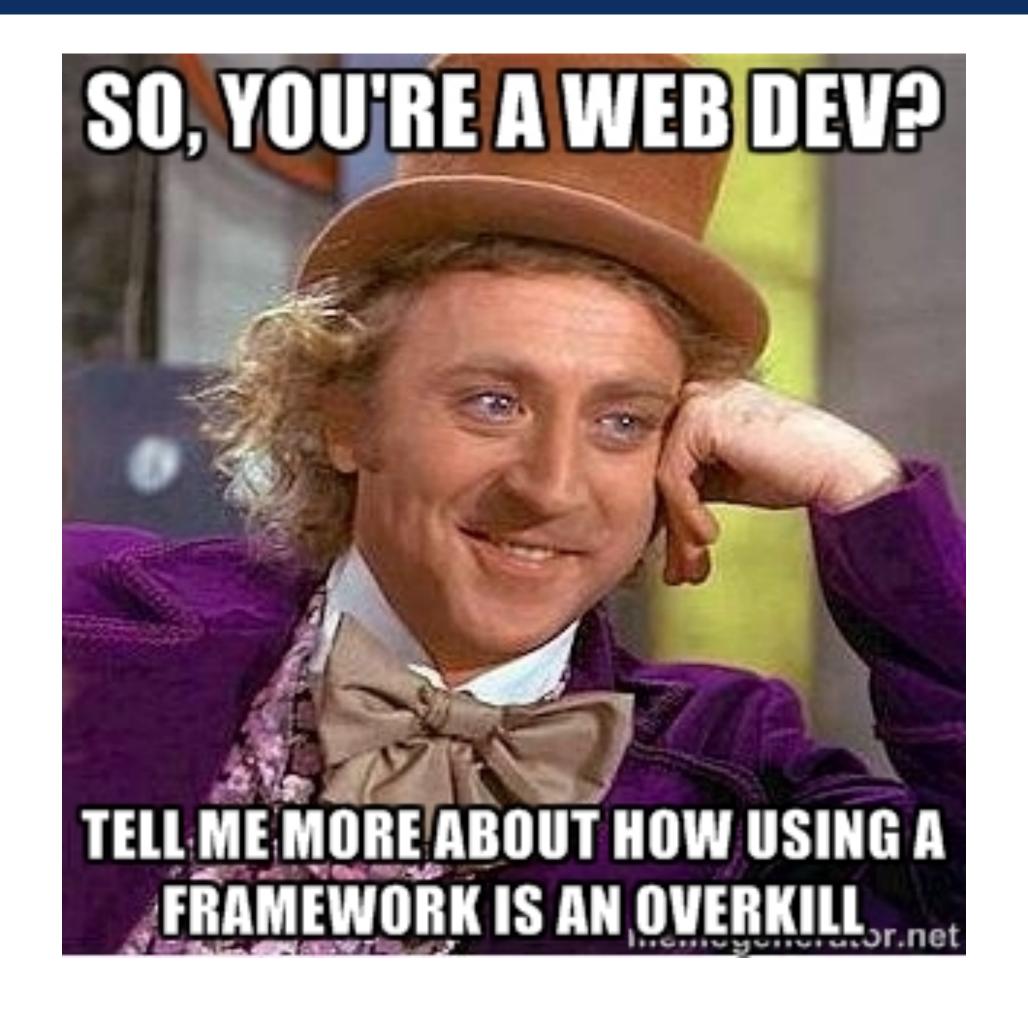








To framework or not?









- · 2 Perspectives
- · Tough questions







Perspectives

- 1 Learning a language via a framework
- 2 Developing a system for production







Both perspectives have different:

- · Responsibilities
- Priorities
- · Requirements







Using a framework when learning a language

Pros

Good practice is used

Lots of help*

Good documentation

Cons

Isolated from the language

A single framework is not a transferable skill







Good practice

Documentation

Community driven

Developing a system for production

Pros

Extendable

Reliable

RAD

Choice

Cons

Bloat

Fighting the framework

Difficult to change once invested







Bloat

Fighting the framework

True of monolithic frameworks







Choice

Compatibility

True of micro frameworks







Rolling your own

Pros

High learning potential

Streamlined and bespoke

You own it

Cons

Lack of support

Not able to develop complex features quickly







Tough questions need to be asked







Is it needed?

Will this benefit me or the project?

How much time is required?

Is it possible?







If you choose to use a framework you should be able to **answer why** you are using it and be able to **identify** what **problems** it is solving

Write this down!







What **features** will the project have in the future and will the framework **allow** for those **changes**?







You shouldn't choose to use a framework The needs of the project should choose for you













