"To designers, HTML is development, to developers, HTML is design"

Why not waterfall?

When we say 'agile'…
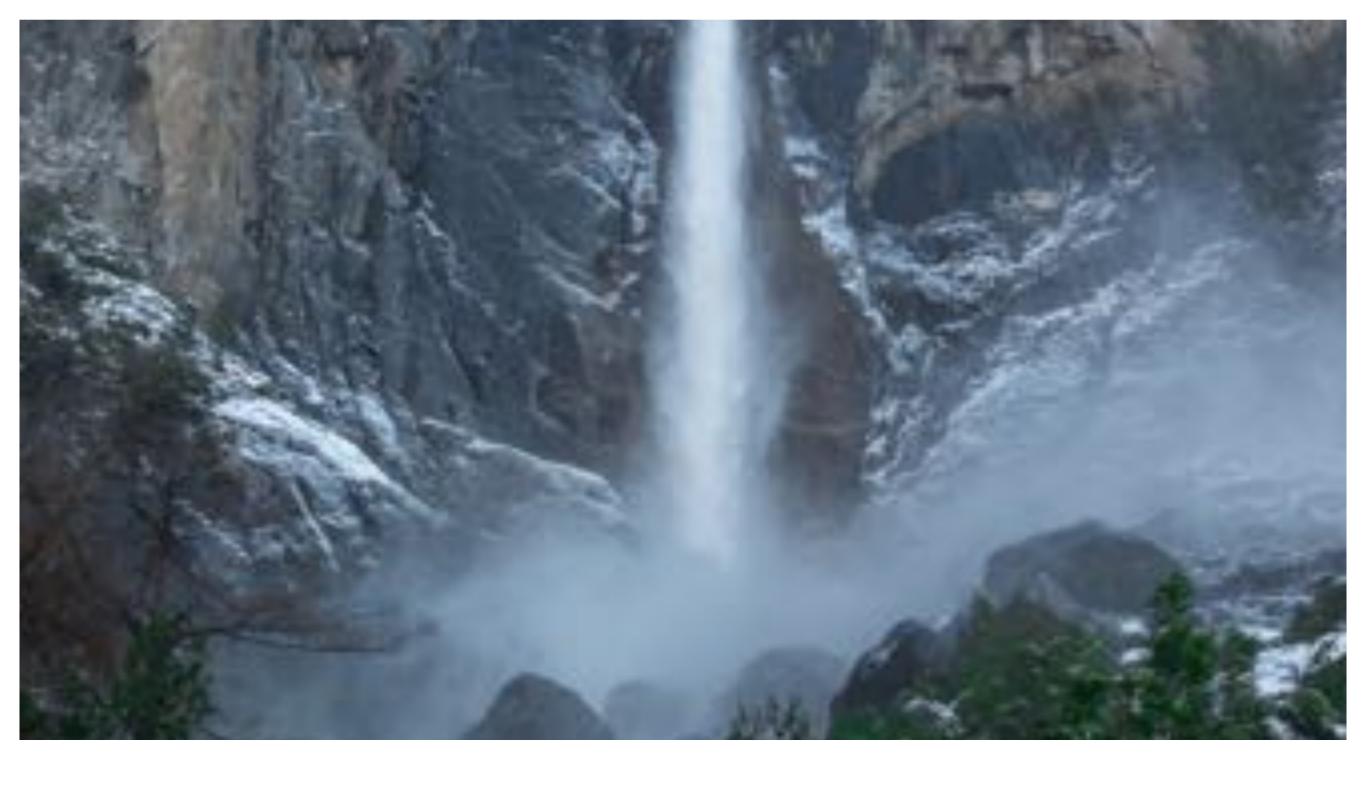
But craft?

Why *not* waterfall?

Individual excellence.

'Design waterfall' encourages unhelpful individual excellence when we need collective achievement.
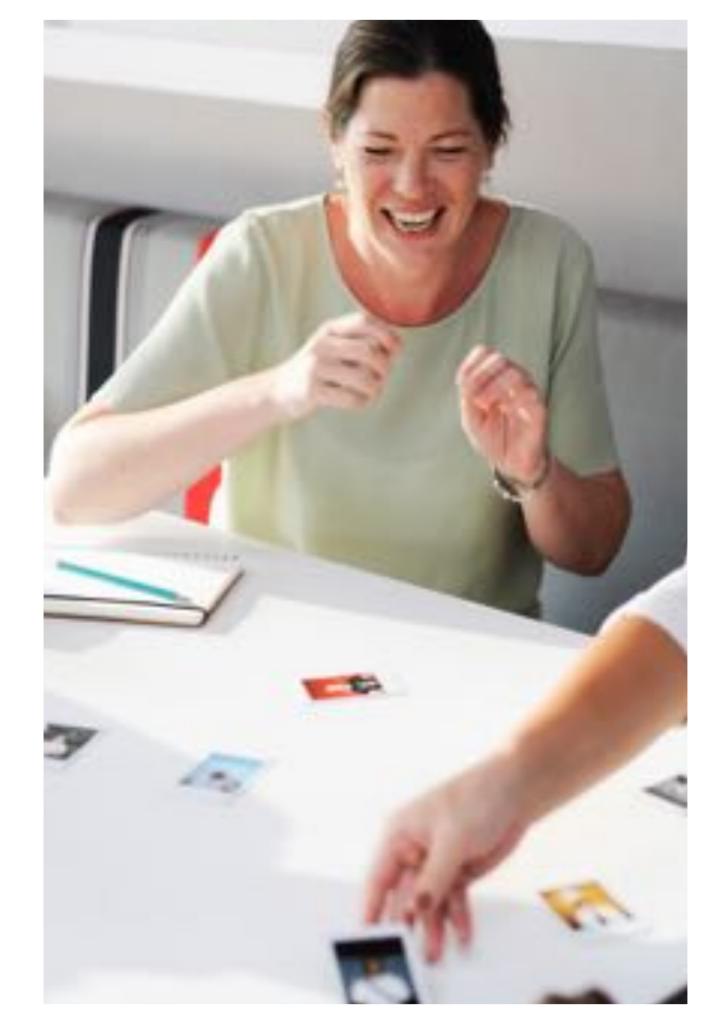
The desire to be perfect.

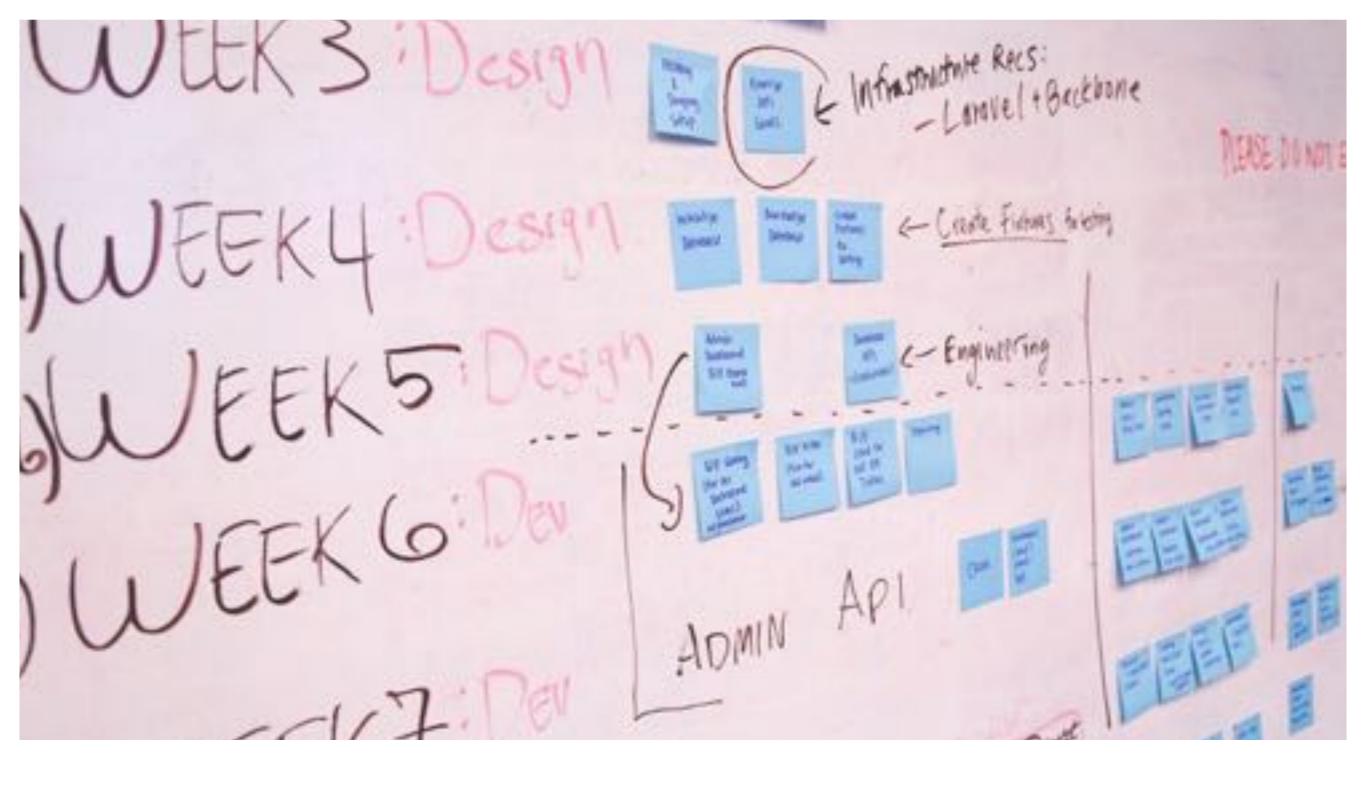Waterfall plays to design's worst instinct: the desire to be perfect.

Deliverables instead of design.

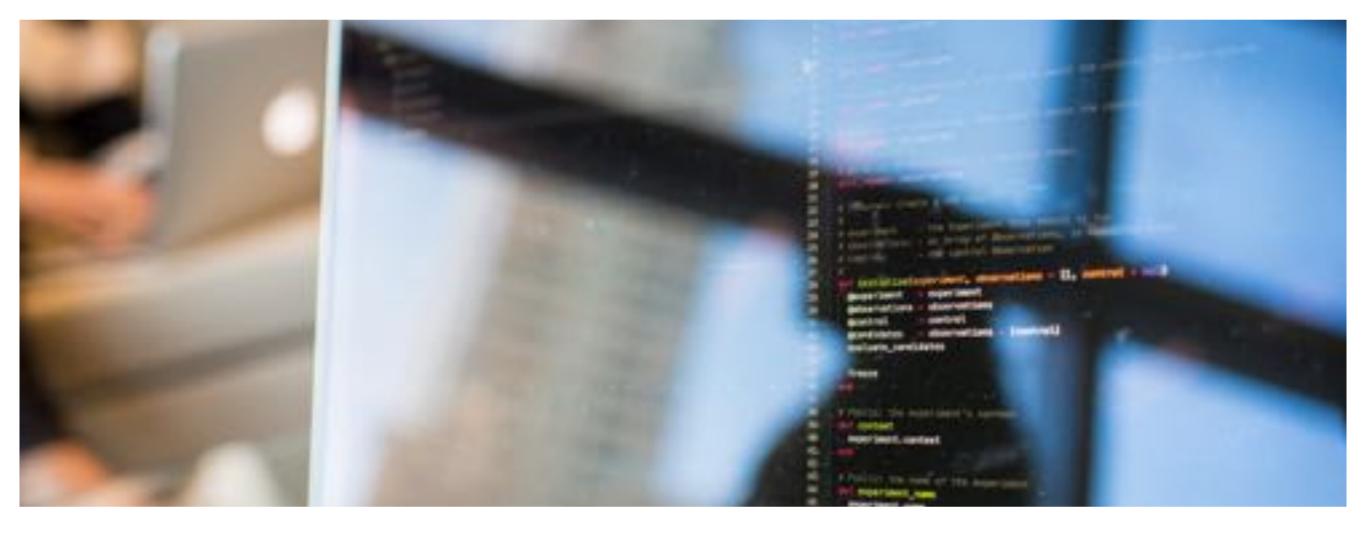We need 'big D' design: intuitive decision-making, product innovation, human-centred practice.

IDEA

Agile encourages visible and valuable, human-centred practice. If we want it to.

Um, what does it mean when we say 'agile'?

Just 'working software'?

The Agile Manifesto promotes 'working software' over documentation, responding to change over following a plan.

Let's iterate the manifesto…

It runs properly and *does what the end user needs it to do.*

Agile + Lean UX + Design Thinking.

We need continuous delivery combined with build-measure-learn loops and innovation through observation.
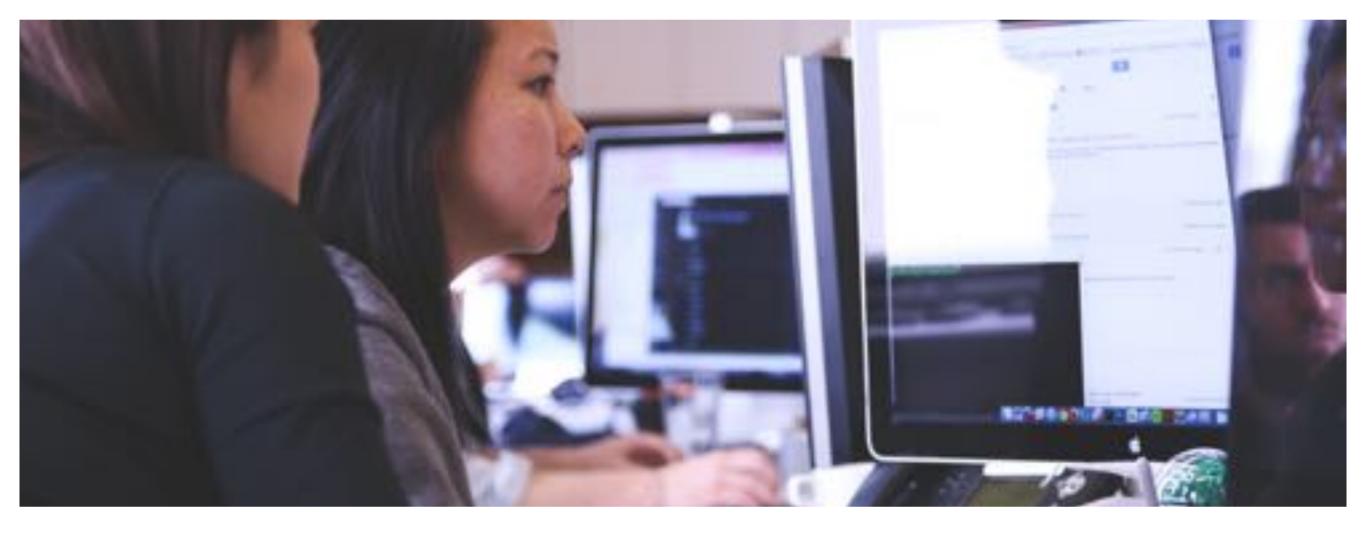
IDEA

We need usability as the definition of done.

But what about craft?

Product design: experience + interface.

Experience (IA, workflows)
Interface (interaction, styling)

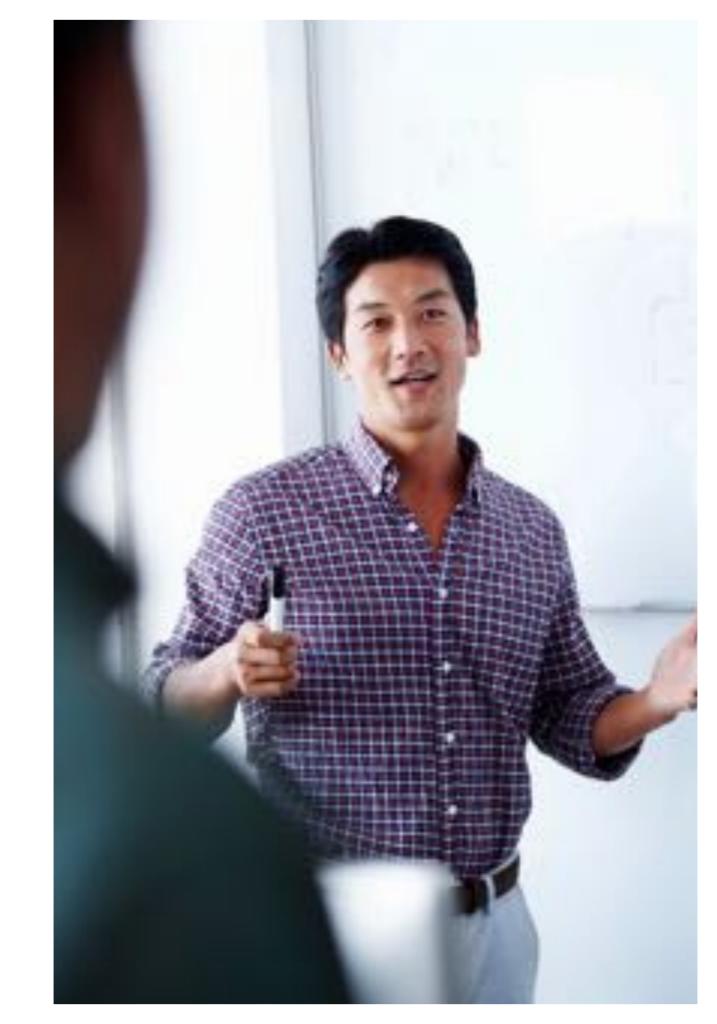## Execution of craft is where the magic happens.

Experience is the deal-breaker, but the visual interface is often where the magic happens, to delight the end user and pleasure of the team.
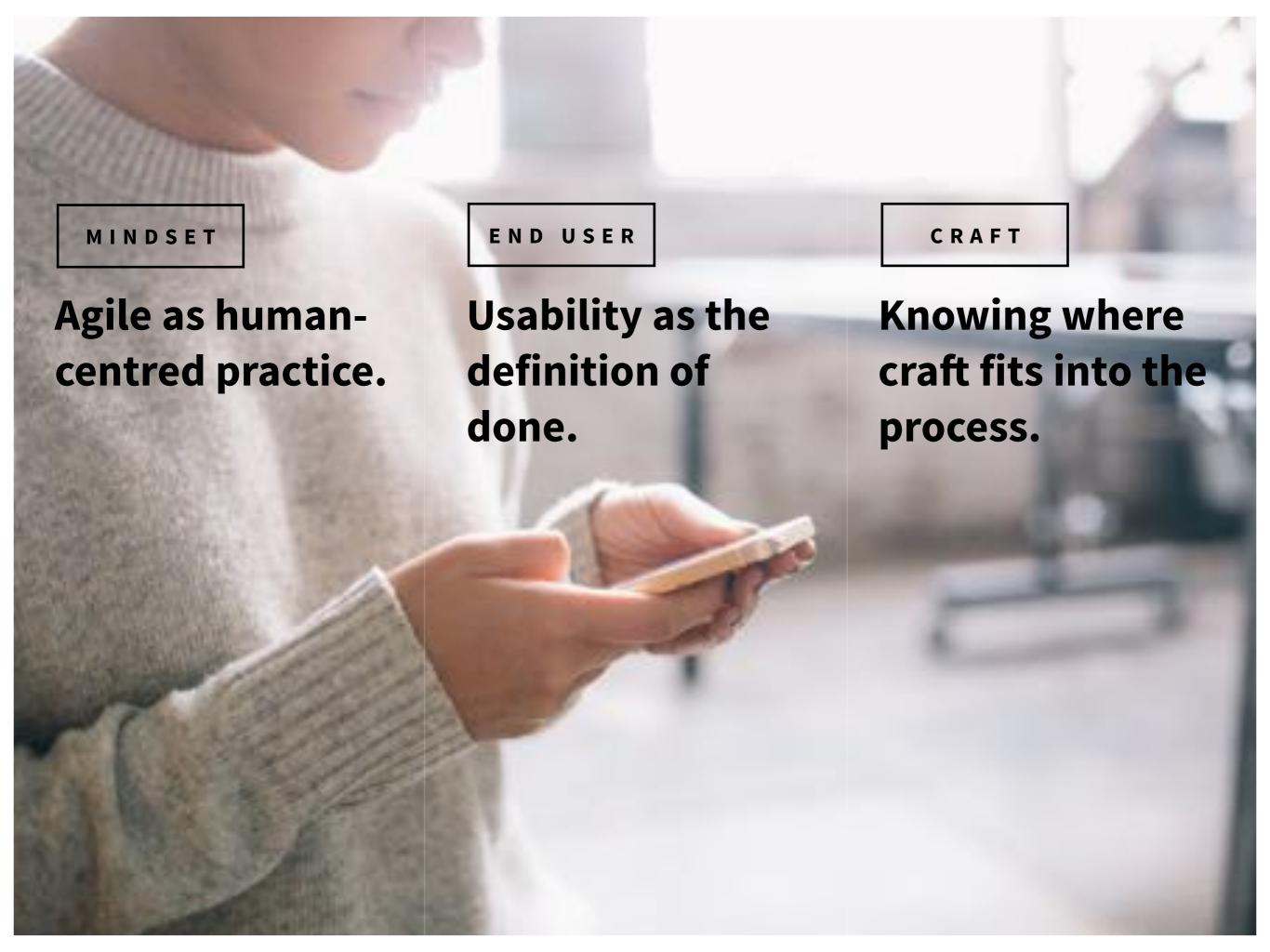
Designers and engineers: craftspeople.

We're both craftspeople on a mission to create and deliver high quality work with the outcome of great experiences.

IDEA

We need to know where craft fits into the process.

MINDSET

Agile as human-centred practice.

END USER

Usability as the definition of done.

CRAFT

Knowing where craft fits into the process.
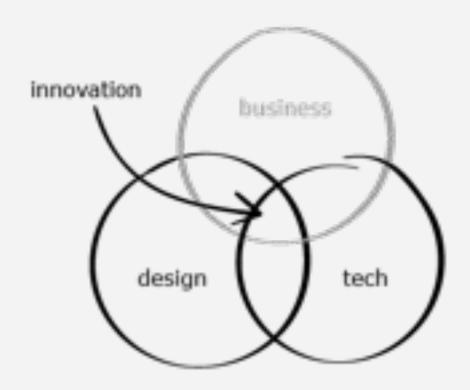
"Agile doesn't have a brain"

Jeff Gothelf (author of Lean UX)

Agile as human-centred practice.

By our powers combined, we are the largest part of any product delivery team.
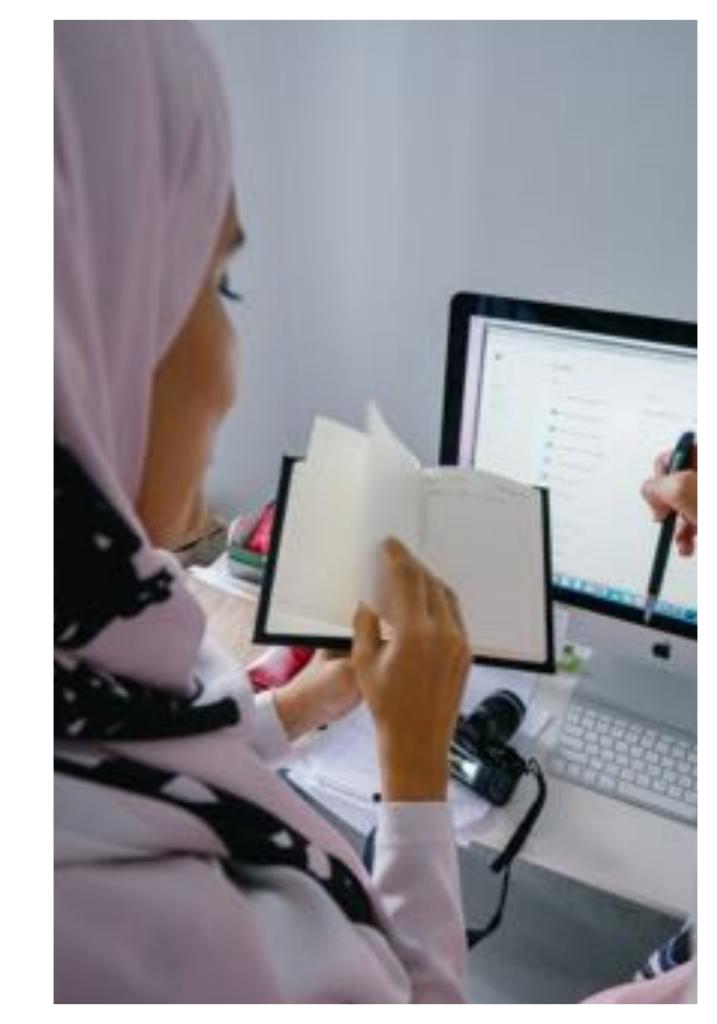
We can influence, even if the business is only 'agile' in name.

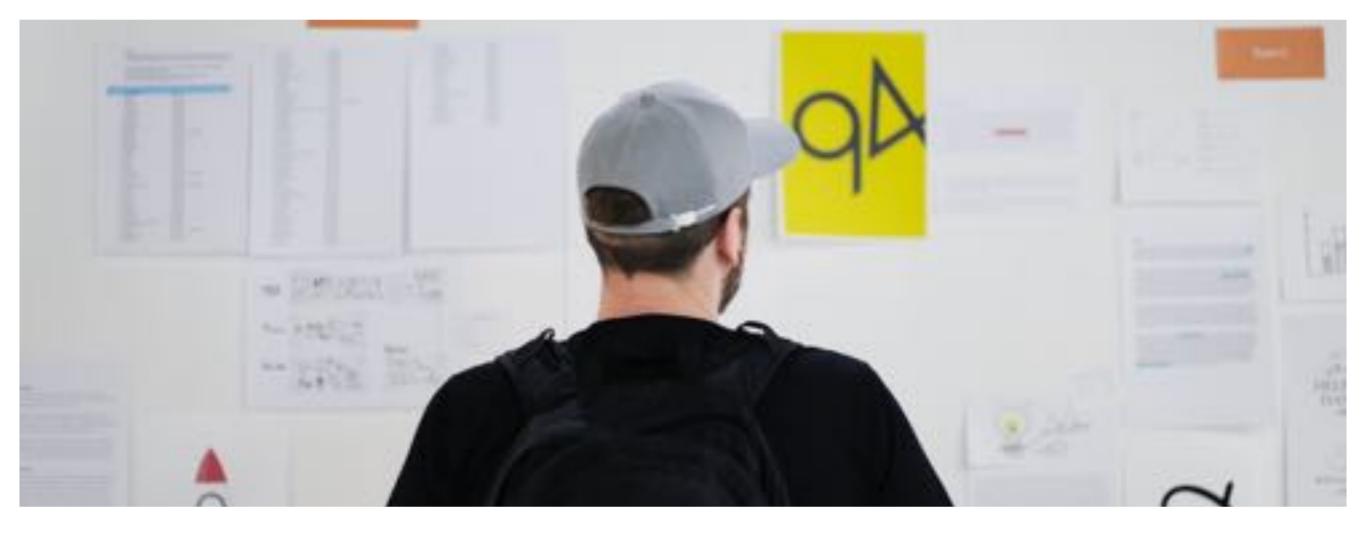A self-determining team is a high performing team.

Agile as human-centred practice.

1. Show early, show often
2. It's iterative, not rework
3. Interaction before interface
4. Learn each other's language
5. Same same, but different
6. Believe in end user value

Show early, show often.

Surface the questions, don't be afraid of the mess (at least within the project safe space). Gather in the most face-to-face way possible.

Try: showcases, design boards

It's iterative, not rework.

Keep the early stages light and focussed on feedback. Design is a process which follows a product for a whole lifetime.

Try: pairing / "Hour of power"

Interaction before interface.

Rapid prototyping or spikes help decision making and later velocity. Get the interaction right and the interface will follow.

Try: prototyping, spikes

Learn each other's language.

Build a shared understanding of design principles, software solutions and blockers - even if it means trying to say it differently.

Try: white boarding sessions

Same same, but different.

We can both be 'agile' with discipline specific methods. Design sprints may be faster, engineering more tracked - it's OK, if it syncs.

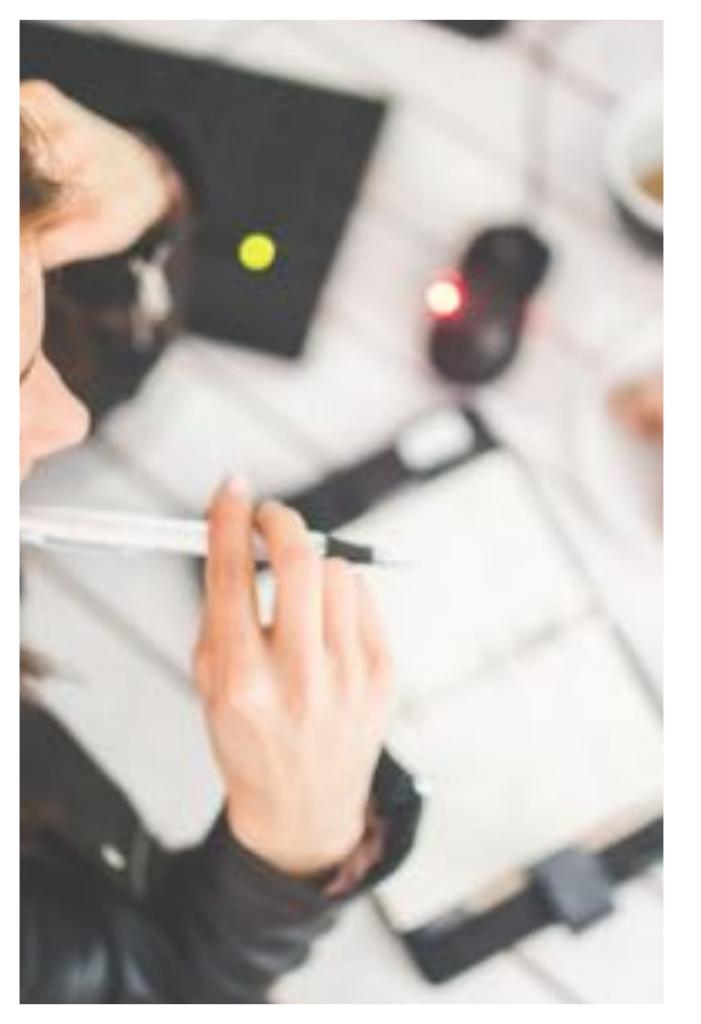**Try: less ceremony, more culture**

Believe in end user value.

Be motivated by user validation. Ask for 'jobs' not tasks. Designers need to share the testing outcomes with the team, not just business.

Try: open testing, JTBD

"If you don't want to be treated as a service, you need to have an opinion - what problems are worth solving and why"

Julie Zhou - VP Product Design, Facebook

Usability as the definition of done.

Let's get obsessed with the problem, not the solution.

'Agile' is just a methodology for getting value, greater and faster, to our end user.
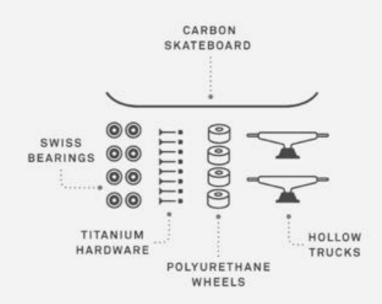
It's about (user) outcome not output.

Usability as the definition of done.

Jobs-to-be-done aka JTBD reframes user stories from a pre-defined solution to surfacing the problem.

When [situation]

I want to [motivation]

So I can [expected outcome]

Homework: Clayton M Christensen

Even though customers buy this...

CARBON SKATEBOARD

SWISS BEARINGS

TITANIUM HARDWARE

POLYURETHANE WHEELS

HOLLOW TRUCKS

...they *really* want this.

"*A dictionary is full of words with fixed definitions, but therein lies every element needed for poetry*"

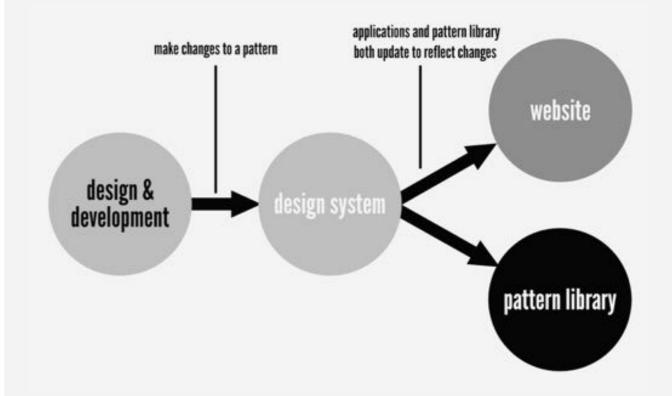Alex Schleifer - Head of Design, Airbnb

Knowing where craft fits into the process.

Design debt is as detrimental as tech debt, and often stems from rushing at the start.

Defining brand / patterns is a seperate track task, ideally in a design system.

Best not to develop a feature still in active 'design' phase.

Homework: Brian Muenzenmeyer

If you have a design system: you win.

Reusable components can build any number of applications. Even better if the living system is cared for by a central DesOps team.

If you don't, you'll need to allow time to define.

Set aside time to develop your building blocks. Otherwise your UI process will be 'iterative chaos' as patterns are defined on the fly.

Start with what you know and ask why.

Sleuth and uncover. Develop patterns together. Share the defining principles early/often. Stay coherent, if not consistent.

MINDSET

Agile as human-centred practice.

We the people create the living agile culture in our organisations.

END USER

Usability as the definition of done.

Let's obsess over the problem, not the solution.

CRAFT

Knowing where craft fits into the process.

We can encourage Design Systems culture to the benefit of all.

@LUCINDA BURTT

Thank you.
Any questions?