

Reading notes on design principles

Most design principles tend to be consensual statements of good intentions, read once and instantly forgotten as they're of no use day-to-day. This document tracks best practices for eliciting and writing useful principles.

Zack

The purpose and consumers of the contents need to be clarified first. How far you would like the principles to impact your internal orgs? Is it for [UX-internal] educating or align stakeholder expectations? Establish a common language in the whole company?

Product Design [can be] a supporting org/subset of Prod Dev, so PM and Dev are involved in signing-off. The principles need to be widely-relatable/educable so that they actually accelerate the buy-in process. Something to watch out for to manage stakeholder expectations.

Fredrik Matheson

<https://twitter.com/movito/status/1028548828931272705>

Bill Bernbach:

A principle isn't a principle unless it costs you something

Jared Spool:

Values worth sticking with are only meaningful when they are hard to execute. If they are always convenient and easy, they aren't values. They are just good practice. Values are the things we do when the wrong thing is easier and more lucrative.

Creating Great Design Principles: 6 Counter-intuitive Tests

<https://articles.uie.com/creating-design-principles/>

It's easy to separate a good design from bad designs. A great principle helps you distinguish a great design from the good designs.

- Test 1: Does It Come Directly From Research?
- Test 2: Does it Help You Say 'No' Most of the Time?
- Test 3: Does it Distinguish Your Design From Your Competitors'?
- Test 4: Is it Something You Might Reverse in a Future Release?
- Test 5: Have You Evaluated it for This Project?

- Test 6: Is its Meaning Constantly Tested?

Easy to use is not something that will ever set a competitor apart. However, *Focus on polish before new features* could set your design apart from a competitor trying to win the hearts and minds of customers through new features, even though their existing platform is buggy and convoluted.

Web App Masters: Design Principles

<http://www.lukew.com/ff/entry.asp?1364>

How do we tell if our designs are getting better? We need way to evaluate them. Design principles embody a design philosophy and help teams get to better design decisions.

Design principles provide abstract guidance about how to make decisions. Dieter Rams was one of the first to apply design principles to his work.

Google and Facebook have very similar high-level design principles (useful, fast, simple, human) but these principles are so broad that they are almost useless.

Companies use generic principles because they are the only things people can agree upon. Who would argue that “fast” and “universal” are not good principles? As a result, these generic principles show up a lot.

The problem with really long lists of principles is that no team can keep them all in mind as they go through the design process.

Good design principles are specific and not overly extensive.

Design principles should not be common. They need to be specific to what you are trying to accomplish.

There are six questions that you can use to evaluate your design principles.

- Does the design principle come directly from research? What kind of research finds principles like “clean” and “universal”? None. Knowing the observations that led to principles is makes them actionable.
- Does this principle allow you to say “no” most of the time? We need to get rid of good designs. We need to get to great designs.
- Does the design principle distinguish your design from your competitor’s designs?
- Have you evaluated the design principle for this project?
- Is this design principle something you might want to reverse in a future release? If we can imagine effective designs that don’t meet our principle. That’s how we get to great deigns from good designs.
- Is the design principle’s meaning constantly being tested?

What makes a good design principle?

<https://matthewstrom.com/writing/principles>

- Good design principles are memorable
- Good design principles help you say no
- Good design principles aren't truisms (a good test of truism is the reversibility test)
- Good design principles are applicable

A shortcut to good principles: "Even Over" statements.

Reminiscent of « pour savoir si un terme est vide de sens, il faut chercher son contraire. Personne ne voudrait par exemple s'appeler A l'arrêt », from [Pourquoi les partis politiques se prennent de passion pour le point d'exclamation](#) (FR).

A Matter of Principle

<https://medium.com/the-year-of-the-looking-glass/a-matter-of-principle-4f5e6ad076bb#.crmh0x42x>

A good set of design principles, on the other hand, does the following:

1. Helps resolve practical and real-world questions around specific design decisions.
2. Applies to an entire class of design decisions, both things we can think of today, as well as questions that will pop up in the future.
3. Imparts a human-oriented sense of "why?" that is easy for everybody—including non-designers—to understand.
4. Has a point of view and a sense of prioritization that a rational person could disagree with.
5. Is generally paired with illustrative examples that show how the principle applies to specific decisions.

Instead of relying on gatekeepers to keep a high quality bar, better instead that everyone gets to agreement on a smaller set of guiding values, so that the best decisions get made in a consistent manner, scaling across many decisions, and even many designers.

Personal Notes

An attempt at consolidating the above into an actionable framework.

Principles are "a fundamental truth that serves as the foundation for a system of belief or a chain of reasoning" ([Julie Zhuo, 2015](#)).

Their point is to support alignment and decision-making, and to a lesser extent creativity. They can come from unmovable facts (business strategy, market constraints, technical limitations) or subjective choices (e.g. brand, personal preferences).

Good design principles are MEMOSPADE:

Memorable

Clear, concise and impactful so that they can be internalized, thus generative, rather than referred to, thus conformative. Use short, unambiguous, and well-illustrated statements. Quantify them when possible.

For example, “discrete inputs over analog movements”.

Evolving

Reasonably iterated upon, to remain relevant as the product grows. Adjust their substance and scope, especially in the earlier phases when things are still in flux.

Meaningful

Impart a human-oriented sense of why, so that they’re memorable and motivating. Make the rationale interesting to the reader (designers or non-designers) rather than true to the system.

Outbound

Expressed in terms of end-user benefits rather than internal benefits. Don’t mention tools, technical capabilities or other similar internal properties.

For example, replace “use push notifications” with “keep the user apprised timely”.

Specific

Not a truism, indistinguishable from competitors, that doesn’t help making decisions. Replace “always true” statements with opinionated visual preferences, focused guidelines or business idiosyncrasies.

For example, change “easy to use” to “clarity before efficiency”.

Prioritized

Ranked against one another, so as to provide a path to resolution for difficult use cases. Write principles as “x over y over z”, or make an explicit different between must have and nice to have.

Applicable

Can be easily connected to designs, and acted upon. If a statement isn’t actionable it’s a goal, not a principle; principles must support design activities. Make sure principles aren’t completely abstract, demanding a lot of thinking to be applied to the product.

For example, replace “make controls obvious” with “use shape to communicate actionability”.

Determinative

Allow to say no, to fairly pick between options. Otherwise the effort devolves into lowest common denominator without style nor conceptual integrity. Break down big statements into smaller, unambiguous ones. Write down explicitly their known consequences, positive or negative. Quantify if possible.

For example, replace “meaningful icons” with “single meaning over several meanings”.

Encompassing

Apply to entire classes of problems, now and in the future. Otherwise they're a pass/fail criteria, not a supporting tool. Don't be too detailed, e.g. principles covering one use case each.

Examples

- [La charte des « Décodeurs »](#) (FR)
- [Gravity](#), with interesting iterations based on experience
- [Linear Method](#)
- [Paste](#)