

RFC 342 Approved: Sourcegraph Design Principles

Editor: Quinn Keast

Status: Review

Requested reviewers (please review by EOD 2021-03-05):

- **Workshop participants:**

Alicja Suska Rob Rhyne Malo Marrec Joel Kwartler María Craig

Christina Forney Felipe Janer Joel Kwartler Tom Ross

- **Other stakeholders:**

Valery Bugakov Tommy Pesavento

Approvals: Jean du Plessis Patrick Dubroy

Team: @design @product @frontend-platform

Background

In February, members of the design and product team [came together to begin defining our design principles](#). This was prompted by the need for high-level guiding principles for our work now and in the future.

This document provides background on design/product principles as a concept, and then presents the [principles](#) that resulted from the workshop and further revision.

Once we're broadly content with the principles, we will update the handbook and begin referencing these in our work.

Feedback requested:

- Do the principles presented in this RFC resonate with you? Do they create a compelling vision for Sourcegraph as a product?
- Do you feel these principles will be able to help us resolve ambiguity in decision-making?
- If we followed these principles very tightly, would Sourcegraph be a better product?

What are design principles?

Design principles are a set of statements that express a shared vision and values for Sourcegraph's design.

Design and product principles are often the same thing.

How will we use design principles?

- To help our team make consistent decisions.
- To provide constraints that lead to better outcomes.
- To resolve ambiguity when faced with options that provide value among different dimensions.
- To build a shared vision across our design and product team.

What informs these principles?

- Our company mission and values
- Our team member's perspectives and values

Who will benefit from these principles?

1. Design team members.
2. Product team members.
3. Engineers, marketing, and other stakeholders that are involved in the design process.

What do principles look like in practice?

A design principle generally follows the structure of a **statement** (the principle, which may reflect experience or process), followed by **elaboration** (a statement of interpretation).

Here are some examples:

Make it feel personal

Our designs reflect that we're connecting real people, not entries in a database.

– Intercom

Unified

Each piece is part of a greater whole and should contribute positively to the system at scale.

There should be no isolated features or outliers.

– AirBnB

A good set of principles is concise and purposeful – about 3–7 principles that work together to create a shared whole. The fewer the principles, the more each principle [brings to bear](#).

Design principles vs. design patterns

Design principles express a shared vision and values. The best principles are simple, broadly applicable, and capture what is most important to both our business and our users. Principles can change, but not often, because changing our principles reflects that the product itself is changing in a fundamental way.

Design patterns are the tangible and repeatable expression and application of our principles in combination with heuristics of good design and usability. Patterns are likely to change over time to meet new needs, technology, and trends.

Design principle:

Empowering

We want people to feel like they can accomplish whatever they're trying to do. Our experiences should give people confidence that they're capable of achieving their goals, no matter their level of experience.

– Shopify

Design pattern:

Provide an escape route

Every flow must have an escape route that will let them return to their previous context, while remaining confident in the integrity of their data and place within the product.

– (Invented)

Proposed Sourcegraph Design Principles

1. **A personal tool within a larger workflow**

Sourcegraph is a powerful yet personal tool that exists within a larger workflow.

Design for familiar patterns with thoughtful defaults, while embracing personalization and adaptability.

2. **Made for everyone**

Our purpose is to make it so everyone can code. This demands we make

Sourcegraph accessible and useful for all developers through universal, inclusive design.

3. **Gracefully manage complexity**

Sourcegraph supports complex product requirements, but also empowers users to manage this complexity for their individual needs.

4. **Code as content**

More time is spent reading than writing code. Elevate the craft of code as content.

5. **Trust is earned**

Sourcegraph is the source of truth, but this trust is earned. Accuracy, transparency, recency, and honesty together work to uphold this source of truth.

6. **Create momentum**

Help developers create and maintain flow. To do this, Sourcegraph must be fast in every way. We design purposefully to help users iterate and build momentum.

The evolution of these principles can be found in the [workshop document](#).

2021-03-09: [Principles are now published in the handbook](#).