

# SKILL SHARE.

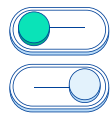
## Making Software Development a Creative Process at Skillshare

### Skillshare: An online creative community

Skillshare is a thriving online learning community that offers more than 30,000 online classes. Members take classes from experts on a wide variety of creative topics from art and graphic design to writing and interior design. A combination of free and subscription-based classes have fostered a creative-leaning community of more than 8 million students.

The Skillshare team is always endeavoring to acquire new users to join the community and to connect existing users to great content and to one another. Their tech stack of PHP, React, Node, and GraphQL is the foundation enabling Skillshare to stream classes and successfully engage their students and teachers.

### Data points



- Skillshare is constantly running 5 or 6 A/B tests, representing 64 (or more!) versions of their app live in production.



- Engineering cycle time sped up 12X, from monthly to weekly releases.

## Setting the development team up for success

A key component of the Skillshare tech stack is A/B testing, allowing for rapid learning about what inspires members to continue their creative journeys. Skillshare had built an internal A/B testing framework with rudimentary capabilities to run tests on 50% of their users. But with such a blunt tool that wasn't able to withstand statistical rigor, the development didn't embrace it.

And it wasn't just A/B testing that wasn't fully developed. According to Mike Sherov, Principle Engineer, "we were doing CI and CD but we didn't have feature flags in place that would allow us to do it safely."

The development team didn't have the permissions and targeting capabilities to enforce the concept of "don't do this in production unless..."

Mike recalls that "we would deploy 10,000 lines of code at once, 3 or 4 months of work, maybe we should QA this in a production environment first." He knew this needed to change. And he had a clear vision for where the development team needed to go.

**“Cycle time is the best predictor of engineering team success. And so I knew we needed feature flags for that.”**

## Speeding up innovation cycle time

Skillshare partnered with Split in 2019 to upgrade their innovation speed. This included both the introduction of feature flags into the software development process and well as an upgraded set of A/B testing capabilities. According to Jean Steiner, Ph.D. and VP of Data Science, "Split makes it possible to really understand when we make changes, how do our users respond. What's the best change for our users and how can we help our users satisfy their creative journey?" This led to a clear business case; "it was the A/B tests that made the case for the investment."

Getting started with Split was fast. At Skillshare, Mike describes, "our engineering team is built on the view that the most important thing you can do is deploy code." Once they implemented Split's SDKs and began using feature flags, "our engineers are developing features in a more cohesive, bold, and innovative way because they aren't afraid of changing

things too much.” This had an enormous set of impacts on engineering efficiency. Mike notices that feature flags “improve our code review, with more frequent and consistent feedback since the team can deploy incomplete features.”

## Moving faster, not breaking things

Feature flags work by separating out deployment from software release. In Mike’s view, “feature flags are table stakes. Feature flags have become a bedrock principle of modern software development with CI/CD.” Skillshare has reduced cycle time by reducing the number of long-lived feature branches. As Mike reflects, “before we had to release it all at once and fix it all at once.” This frees the Skillshare development team to “build continuously.”

In addition to moving faster, Skillshare is able to test in production since in production no longer means “it needs to be live for all users.” Feature flags provide “a sandbox within our production infrastructure.” This ensures a better quality of engineering changes. With feature flags, “when you release it’s much safer. And it gives the team the feeling of safety. Their work can be verified in production by others.”

## **“If you’re not using feature flags, you’re just being dangerous.”**

And Jean notices it across the development team. “the benefits of being able to QA in a real production environment have been huge. We frequently catch stuff with feature flags or A/B tests that would have been a user-facing problem. We could not have done that before.” And the safety benefits aren’t just for users. They directly free up time on the development team. “Without feature flags, you spend so much time fighting fires that it diminishes your capacity to learn.”

## Searching for the first experiment

Within a few months, Skillshare had their first experiment running. And it was an important one! To improve their member engagement, increasing the number of lesson views, Skillshare was considering an infrastructure transition. They were planning to migrate their backend search algorithm from Elasticsearch to Algolia.

With Split, they were able to manage the migration and evaluation process. Mike knows that “if we didn’t have feature flags it would have been a huge change.” The development team was able to compare the two infrastructures side-by-side and declare a winner. “It was a high stakes experiment.”

And it was a huge upgrade on their previous A/B testing system. Mike reflects, “it was the first time we had clear data flowing in. It was a really nice validation of the concepts.” The Skillshare team managed everything through their server-side, back end code to manage the feature flags and A/B tests. “We decentralized it so that it works across multiple microservices. The Split synchronizer architecture was really powerful for us, enabling Split as a container in our datacenter.”

## First, do no harm

Software development, and the change inherent in it, is not just about improving user outcomes. It is also about making sure bad outcomes don’t happen. For Jean, this ‘do no harm’ is a software development version of the hippocratic oath. “If we make a change, we want to know if the drug is better or worse for our patients. Our patients are our users and we want the application to be a great experience for our users and our business.”

**“Split allows us to account for all the other things that are going on in the world and isolate the impact of the change we made on our users and business.”**

Split helps the Skillshare keep their users from experiencing bad outcomes in the controlled chaos that is software development. Jean explains that “phased rollout and monitoring at each stage is really important because there might be certain user behaviors that you don’t observe when you have 5 users, but when you have 2,000 you’ll hit one of those less frequent events. And you want to hit it before you have 100,000s.”

With Split, Skillshare targets specific user segments, rolls out to proportions of the user base, and validates that guardrail metrics aren’t negatively affected. The engineering team knows this all too well. Mike checks in each release “to make sure the metrics that shouldn’t have moved, didn’t. If it’s too good to be true, it is.”

By approaching every engineering change with a focus on impact, the entire Skillshare team, “engineering and data and product, understands which metrics are going to move in which ways.” Mike highlights that this focus on data “allows us to always put the user first in everything we do. It means we have more eyes on what could be going wrong or right.”

## Reaching liftoff

Moving from feature flags to experiments was always the true business reason behind adopting Split.

### **“A/B testing is a side effect of controlled release. That’s what I love most about Split.”**

Things got real 6 months after the first experiment. Skillshare was embarking on a major brand refresh, embracing their creative lean, that would transform all parts of their application and user experience. Months of work across every team was building up to a single change.

Skillshare used Split feature flags to develop the all of components of the brand refresh, deploying them to production and then releasing them when they were ready for the final company launch. Mike saw huge benefits for the engineering team. The final release “was a flip of a button rather than a massive deploy. There was no feature freeze. This wasn’t just swapping out the engines mid-flight, this was like swapping out the plane while you’re flying. It’s hard to imagine how you’d accomplish something like this without feature flags.”

Jean recalls it was “a fast and measured rollout of the site for a few days, where we made sure the core metrics were either improving or comparable, and then we went to 100%.” It was that simple. Since every feature flag in Split tracks core company metrics, Jean saw that “the A/B tests gave us a lot of visibility, protection, and comfort in such a huge change to our site.”

The results from this branding overhaul “was a magical ‘aha’ moment for the entire team” and there were no longer doubters wondering whether the effort of “adding feature flags and taking them out later” was worth it.

Fast forwarding to today, at any given time Skillshare has 5 or 6 A/B tests running. This represents 64 (or more!) versions of the Skillshare app running concurrently.

## Creating a culture of safety and innovation

The journey to making software development a creative experience is rooted in team culture. For Mike, the “psychological safety of feature flags is tantamount to the developer experience.” And pairing these feature flags with data; well that is a cultural revolution. “It transforms the way we talk about our site. We now know how a page works. We can show any member of the team the tests we ran in Split, what we know about the page, and how we think about this page”

Together Jean and Mike have built a whole product development process around Split. Anyone can come forward with a new idea, a new hypothesis. “We figure out which tests we want to run. Which ideas we want to model, including what the potential impact would be. Then we discuss, debate, and prioritize ideas on the roadmap.” This is the new normal. A creative journey that the Skillshare development team is now aligned around.

**“It’s inherently collaborative and transparent, a rigorous and sound process based on the hard data we know about.”**

Jean and Mike look forward to the speed at which software development is taking place at Skillshare. And with Split as a key technology, Jean notes “it has been a real partnership!”

# SKILL Share.