

Skyscanner & GitHub



INDUSTRY

Travel

CUSTOMER SINCE

2018

PRODUCT

GitHub Enterprise Server

NUMBER OF SEATS

800+

No matter the destination, everyone has a different strategy for snagging a deal on flights. Maybe you booked your trip to London a season in advance to get the lowest price. Or, if you're the impromptu type, you may have packed your bags in the morning and bought a cheap ticket to Bali that afternoon. Whether you only purchase flights on Tuesdays or meticulously time your travel, your best bet for a better deal is an algorithm. Most travel sites will spark your wanderlust with ads but to get you where you're going, you need a plane ticket you can afford. That's where Skyscanner comes in.

Since its launch in 2003, Skyscanner has been growing continuously from a modest apartment to 11 offices around the world—in London, Barcelona, Beijing, and more. Their technology evolved from a Microsoft Excel spreadsheet into a site that serves 80 million unique travelers in over 30 languages. Behind the technology are hundreds of committed engineers aiming to build software like the perfect flight: fast, affordable, and friendly. "We want to make travel easier and more accessible, wherever people go," said Engineering Manager Stuart Davidson.





Davidson is an Engineering Manager in Skyscanner's Developer Enablement Tribe—a group that supports developer success across the organization. Responsible for Skyscanner's deployment pipeline, Davidson's team makes sure high-quality code moves efficiently from idea to production.

Before switching to GitHub in 2018, Skyscanner used a different solution. "We found it very difficult to upgrade and grow," explained Davidson. As Skyscanner's codebase grew, so did their infrastructure. When a certain scale was reached, pain points arose especially around the team's backup workflow. Something had to be done.

"Our previous solution was taking up a tremendous amount of our own support time," said Davidson. "The team was working for the tool rather than the other way around."

Why Skyscanner chose GitHub Enterprise

- Shift focus from maintaining infrastructure to faster build times.
- Find a platform that grows with teams and evolves with technology.
- Create a consolidated, low-maintenance home for all of Skyscanner's code.

The search for a new tool started with several goals in mind. "We wanted a best of breed source control system," Davidson explained, "and reducing operational overhead across the organization topped our list." With so many technical issues to address, engineers often spent time trying to fix and fine tune their previous solutions, instead of working to delight their "customers", internal developers.

At first, it was the GitHub team's grasp of Skyscanner's issues that encouraged Davidson to sign up for a trial. He also appreciated the openness and transparency of the Solutions Engineer helping him solve problems above selling a product. But ultimately, he chose GitHub Enterprise for its operability, interface, and support.



Davidson noted, "We were very skeptical because the demo just seemed all a bit too good to be true — but sure enough, GitHub proved to be exactly as it was described. It was a huge confidence boost when we actually got to try it ourselves. There's so little infrastructure to run, everything is inside the box and does it for you."

Now GitHub is used by every engineer building Skyscanner, alongside Drone CI and internal deployment tools that deploy 100 to 200 times daily. But they're branching out. With more than 800 licenses, engineers aren't the only ones using GitHub. Whether they're versioning databases or config as code, the team is proving that a version control system that "just works" goes beyond strong source code.

As a critical piece of infrastructure, source control works best when it stays out of the way. "GitHub just works," said Davidson, "when source control is snappy and fast, it helps our developers focus longer and not have to be reacting to outages or waiting for things to happen."





Davidson's ultimate goal is to run proprietary software like an open source project, where any full-time employee can contribute code to any Skyscanner repository. To that end, their version control system doesn't just consolidate code. It also acts as also a nerve center for teamwork. In particular, Davidson sees GitHub as a foundation for open collaboration and for applying open source development workflows to the team's software—a practice called innersource.

Context switching is a daily battle for teams juggling multiple developer tools. For Skyscanner, consolidation alone has been a big success. Simple tooling allows the team to focus engineers elsewhere. Instead of spending time on source control, Developer Enablement can improve build speeds, try new strategies for artefact management, and more. "The force multiplication of having a source control tool that is as fast and responsive as GitHub is a big win."



In a few short months, the Developer Enablement Team has already lowered operational overhead. "It's an absolute game changer," said Davidson, "we've seen significant savings for our organization and now our engineers have even more time to spend in what matters the most, the business logic."

Best-in-breed productivity and development tools keep teams ahead of their workflows. Adopting applications and adapting toolchains as the organization evolves helps engineers stay nimble and use the latest technologies their industry has to offer. "It's an investment for the future,"

Davidson describes, "I see us still with GitHub in the next five years."

For now, the team is focused on their state-of-the-art DevOps pipeline and making the most of GitHub APIs. So far Skyscanner has integrated with multiple GitHub Apps and other third-party tools like Snyk—an open source security platform—to scan tokens and help keep their code safe. They've also used the GitHub Checks API to run checks instantly and shift code validation earlier in the process.



Beyond automation, Davidson also values GitHub's built-in code review tools. Easy and organic reviews—whether manual or automated—are a sure route to better code. "Pull requests and peer review are core to what we do in Skyscanner," said Davidson, "no code makes it to production unverified." This is part of Skyscanner's Engineering Principles which are public via [GitHub](https://github.com/skyscanner/engineering-principles).

Manual code reviews—and air travel—prove that no matter what a technology has to offer, people have the power to push a good experience into an exceptional one. Davidson revealed this was equally true of GitHub. "GitHub's Support makes us want to work with and build on the platform, rather than just live with it," he said, "We feel listened to and supported — which is not that common in the industry. Being easy to work with can't be undersold."

Visit us at github.com/enterprise/