Introducing GitHub Enterprise

How to transform your business in a digital world

WRITTEN BY GITHUB WITH ♥

Contents

Introduction

No matter your industry, technology plays a central role in your ability to stay ahead of the competition. That's why just about every company is investing in and building software to power their business: From Silicon Valley start-ups to established real estate firms, business leaders are looking inward at their tech stack and their developers, and are starting to think like a software company.

Think of how the banking industry has evolved from in-person to ATMs to online to mobile. Now, people can deposit and withdraw money in seconds. And when it comes to paying: we've gone from swiping a card to inserting a chip, tapping to pay, and then being able to pay with your phone or watch. This is how fast technology is accelerating.

To keep up, thousands of development tools have been created in the past decade. But we've learned some important lessons along the way: More tools don't always correlate with productivity. In fact, bringing in more tools risks creating a fragmented, inefficient developer experience and slow time to market.

Developers are almost 60% more likely to feel equipped to do their job when they can easily find what they need.[1] An end-to-end, centralized platform like GitHub Enterprise empowers developers with tools they already know and love, accelerates high-quality software development and secure delivery, and enhances the speed and power of innovation.

90% of Fortune 100 companies use GitHub Enterprise

Get to know GitHub

GitHub accelerates progress by helping you build faster, more collaboratively, and more securely.

83+	4+	200+
million developers	million organizations	million repositories

1: "Writing and shipping code faster | The State of the Octoverse." 2021. GitHub. https://octoverse.github.com/writing-code-faster/#search-improves-software-development

Today's industry challenges

In order to become more competitive in an already competitive market, companies are under increased pressure to deliver secure software at scale and manage increasingly large development teams. All of that while getting the best product to market quickly and creating a good customer experience. There are scores of challenges to overcome, which we'll outline below. The good thing: None of them are insurmountable.

Insufficient supply of developers: The demand for developers is growing: According to the U.S. Bureau of Labor Statistics, the demand for software developers, including software engineers, will increase 22% between 2022 and 2030. Even if you do manage to attract and hire more developers, you still need to onboard them—another process that requires time and resources, and impacts time to deployment in its beginning phases.

Tech stack complexities: To keep up with the ever-evolving tech landscape, DevOps leaders are forced to add more and more tools to their tech stack. This means that defining, maintaining, and provisioning development environments becomes harder, and developers have to spend more time context switching between different tools and less time writing code.

Security: According to the 2022 Open Source Risk Assessment, out of the 17 industries represented, four contained open source in 100% of their code bases. The remaining verticals had open source in 93% to 99% of their code bases.[2] While it's incredibly beneficial to reuse code, companies struggle with doing this securely. In fact, 77% of flaws in third-party libraries remain unfixed after three months.[3] With attackers on the rise, companies need a security solution that surfaces vulnerabilities before they're introduced and makes remediation simple.

Remote work: In GitHub's 2021 State of the Octoverse report, we found that only about 11% of our respondents expect to go back to working in person, a 30% drop from the 41% working in an office before.[4] While there are many benefits to remote work—from saving costs on office space to being able to hire the best talent, no matter their location—the need to stay in sync and collaborate is that much greater. Whether they're in New York or Shanghai, to run at full speed, teams need a solid process in place, a workflow that'll seamlessly take them through merging pull requests to deployment without missing a beat.

2: "Open Source Security and Risk Analysis Report." 2022. Synopsys. https://www.synopsys.com/content/dam/synopsys/sig-assets/reports/rep-ossra-2022.pdf

3: "State of Software Security v12 (SOSS)." 2022. Veracode. https://info.veracode.com/report-state-of-software-security-volume-12.html

4: "Writing and shipping code faster | The State of the Octoverse." 2021. GitHub. https://octoverse.github.com/writing-code-faster/#search-improves-software-development

Lack of documentation: Good quality documentation is essential for onboarding new contributors and creating a shared understanding of processes and goals.[5] Unfortunately, with teams moving so quickly, this critical step is often skipped, leading to missed opportunities and attrition. Even when documentation does exist, if it isn't kept up to date, it can cause confusion and erode trust.

Developer burnout: With so many challenges, developers are exhausted: they're constantly context-switching to address multiple pain points, getting bogged down in what should be simple tasks, like code deployments and running security tests, and struggling to collaborate with siloed teams. Caught up in these day-to-day frustrations, it's impossible for them to jump on new projects and focus on what's important: delivering great, secure software at scale. If left unchecked, this can also affect developer retention down the road.

All these challenges reverberate throughout the business: missed deadlines and unmet goals, increased time to market, increased costs from service disruptions and missed SLAs. This can lead to lost revenue, attrition, low ROI from tooling investments, and a negative brand reputation.

> *"Improving business performance through software development comes down to empowering developers, creating the right environment for them to innovate, and removing points of friction."*
>
> Developer Velocity: How software excellence fuels business performance // McKinsey[6]

There is simply too much risk if you're not paying attention to the developer experience.

This is where GitHub Enterprise (GHE) can help developers, companies, and business leaders. As an end-to-end centralized platform, it encourages collaboration, empowers developers to build productively (while prioritizing their happiness), and accelerates high-quality software development and delivery without sacrificing security.

All of this leads to improved time-to-market, revenue growth with reduced cost of hardware and admin time, improved customer satisfaction, and a much better developer experience.

5: Theunissen, Theo, Uwe van Heesch, and Paris Avgeriou. 2021. "A mapping study on documentation in Continuous Software Development." Science Direct. https://www.sciencedirect.com/science/article/pii/S095058492100183X

6: Srivastava, Shivam, Kartik Trehan, Dilip Wagle, and Jane Wang. 2020. "Developer Velocity: How software excellence fuels business performance." McKinsey. https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/developer-velocity-how-software-excellence-fuels-business-performance

Why the developer experience matters

As development teams have become more integral to the day-to-day success of businesses, providing an excellent developer experience is on everyone's mind. By placing developer experience at the core of your organization's efforts, you can attract new employees and improve retention, enhance security and quality of work, and increase productivity.[7]

When the developer experience is fragmented, innovation is slow. At work, developers need processes to be fast, delightful, and easy; in open source, project leaders and maintainers need ways to make communities welcoming and sustainable.

> *"In order to make the life of a developer easier, rather than harder, a quality developer experience is focused on allowing developers to use the tools they're familiar with..."*
>
> The Developer Experience Gap – tecosystems // RedMonk[8]

At GitHub, we know that recruiting and retaining the best developers starts with prioritizing the developer experience, including an efficient onboarding process, a singular platform that solves for developer needs, and fostering a culture that encourages collaboration. No longer should developers have to forgo coding time to grapple with complex, non-integrated tools and all the issues that come with them.

> *"When was the last time you asked your engineering teams about what slows them down and stops them from shipping awesome software?"*
>
> Martin Woodward // Director of Developer Relations, GitHub

To keep developers focused on learning and long-term growth, GitHub Enterprise simplifies your tech stack, provides tools for automation, and creates workflow efficiencies with an interface that developers know. By reducing complexity, increasing transparency, and anchoring on a central platform, GitHub gives

7: Boehm, Jim, Lucy Shenton, and Daniel Wallance. 2022. "Why your IT organization should prioritize developer experience." McKinsey. https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/tech-forward/why-your-it-organization-should-prioritize-developer-experience

8: O'Grady, Stephen. 2020. "The Developer Experience Gap – tecosystems." RedMonk. https://redmonk.com/sogrady/2020/10/06/developer-experience-gap/

them the ability to focus on what you hired them to do. Perhaps that's why the majority of developers prefer to collaborate on GitHub.[9]

> *"When developers open up their first PR, chances are they already know the workflow. Because of GitHub, they've already done it before; they don't need to learn something brand new. GitHub helps us scale and allows us to concentrate on building amazing features for our clients."*
>
> Mark Côté // Senior Manager of Developer Infrastructure, Shopify

The value of having a centralized platform

Nothing detracts from innovation and creativity more than siloed teams. That's why having a centralized platform is so important: it's one place that empowers developers to do their best work, together.

GitHub Enterprise provides a holistic environment that addresses the development lifecycle journey end to end—where teams can work together to get projects off the ground and delivered, and move on to the next. What's more, GHE combines all the tools that developers love and is built to integrate with all their existing workflows, which reduces context shifting and lost time. From shared best practices to issue tracking and project pages, developers have everything they need to do work smarter and faster.

Let's take a look at how having one centralized platform impacted Telus, a communications and information technology provider in Canada that powers wireless connectivity, accessible healthcare, and safe food supply chains for more than 35 million people. Before switching to GitHub Enterprise, Telus had a massive and fragmented tech stack with different teams (including 5,000 developers) using different tools—many of which were not compatible with each other. This led to long build and testing times, not to mention a lack of communication and coordination among teams.

By harnessing GitHub as its central location for code and communications, Telus could streamline the development process. Now, end-to-end workflows ensure developers know exactly which action to take next, while GitHub Issues and

9: "Where do computer practitioners prefer to collaborate? GitHub – The new stack." 2022. Aumag. https://aumag.net/where-do-computer-practitioners-prefer-to-collaborate-github-the-new-stack/

Projects provide much-needed visibility into every pull request. It's an environment where Telus developers can thrive and do their best work, together.

> *"The bet we made on GitHub was that it would be a unifying technology that would drive innovation and collaboration throughout the organization. As we keep building features, that bet keeps paying off."*
>
> Justin Watts // Director of Engineering and Productivity, Telus

Promoting collaboration at scale

When you can build as one team, you can deliver a strong, unified vision for your business. But growing teams with a variety of different tools to manage, a need to scale quickly, and asynchronous work schedules means coordination can prove nearly impossible without the right solution.

In our State of the Octoverse Report, you'll see that developers work faster when they share work. But too many contributors can make coordination difficult and slow down the workflow.[10]

> *"With GitHub, we can scale and build projects on a new level now. It's not about how good you are alone. It's about the greatness we can achieve through sharing and collaboration."*
>
> Charline Grenet // Head of Digital Communities and Communications, ENGIE

In a real-world example, 3M demonstrates how breaking down silos between teams and increasing collaboration can lead to impactful business growth.

Before switching to GitHub Enterprise, 3M's code was scattered across different repositories and in shared drives. With so much historical knowledge and limited documentation, it was difficult to onboard new members and innersource among teams. This led to lost time and duplicative work.

With GitHub Enterprise's cloud offering, along with GitHub Actions and GitHub Packages, 3M was able to increase their deployments from 400 to 6,000 in just 60 days. With one spot for centralized code, the teams now collaborate on a global

10: "Writing and shipping code faster: The State of the Octoverse." 2021. GitHub. https://octoverse.github.com/writing-code-faster/#search-improves-software-development

scale easily. By building as one team, they're far more agile, whether they want to onboard new members or experiment with a new feature.

> *"With GitHub, we can collaborate better across our various environments. You don't have to go out to a separate project management tool. You don't have to go to a spreadsheet, or a Microsoft project, or into Jira. It's all on GitHub. It's made us twice as productive."*
>
> Tina Beamer // IT Manager of Operations and Quality, 3M

And developing as one team doesn't mean everyone has to be in the same office. With GitHub Enterprise's mobile access, Codespaces, and Web platform, collaboration can happen from anywhere. With GitHub Mobile, for example, you can quickly access your issues and pull requests dashboard wherever you are. That means you can review and merge pull requests on-the-go, whether you're at the office or in a taxi.

> *"Codespaces is a great enabler at a large enterprise with many different requirements because it lets developers skip the tedious, error-prone stuff that normally stands between them and actually getting started on real work."*
>
> Keith Annette // Cloud Capability Lead, KPMG

Here are a few core GitHub Enterprise features that promote collaboration, speed up the developer lifecycle, and help teams go to market faster.

GitHub Issues: For simple project management and tracking, use GitHub Issues. This feature allows you to create an issue to suggest a new idea or track a bug, then assign tasks to your team. By crosslinking between issues and pull requests anywhere in the organization, GitHub Issues allows teams to track updates to projects without overcomplicating the developer workflow.

GitHub Projects: GitHub Projects helps developers project manage and plan, from idea to production. With project boards, it offers a quick glimpse of all tasks that are planned or in progress, either in a repository or across the organization.

> *"GitHub Projects gives us a high-level overview so we can look at all our projects across the board, and triage priorities quickly. Plus the Kanban flow is easy to follow and is already familiar to most developers."*
>
> Lisa Vanderschuit // Engineering Program Manager, Shopify

Innersource: Another way to encourage collaboration with GitHub Enterprise is through innersource, which is like an open source community behind your own firewall. Using innersource methodology harnesses those open source best practices—which can boost productivity by 87%[11]—to securely promote code and knowledge sharing.

All of these tools help developers not only collaborate better together, but give them the ability to bring their ideas to production.

Shell's one-stop shop: GitHub helped Shell embrace Innersource and the team hasn't looked back. With Shell's developer portal, which uses GitHub as a backend, they now have a one-stop shop for code, documentation, training, and guidelines. Anyone who wants to contribute to documentation or guidelines for a project—such as the custom distribution of React used at Shell—can open a pull request and, once accepted, see their changes appear on the portal instantly. Thanks to shared innersource libraries, developers can spend more time on new products and features, keeping Shell on the cusp of what's new in the ever-changing energy industry.

> *"Shareability is a big part of why we've been able to drive adoption so quickly. Developers are really excited about GitHub Actions and eager to share what they've learned. It's a great application of the innersource ethos."*
>
> Sherin Mirza // DevOps Transformation Lead, Shell

11: Srivastava, Shivam, Kartik Trehan, Dilip Wagle, and Jane Wang. 2020. "How software developers can drive business growth." McKinsey. https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/developer-velocity-how-software-excellence-fuels-business-performance

Automation, CI/CD, and development tools

"Organizations with strong tools—for planning, development (for example, integrated development environments), collaboration, and continuous integration and delivery—are 65% more innovative than bottom-quartile companies."[12]

At GitHub, we know that developer happiness is one of the keys to a thriving business. But with the complexity of technology and systems increasing, it's that much harder to both scale your business and keep teams happy in the process. GitHub's built-in automation tools help teams adapt to the changing landscape without having to implement anything new.

Strong automation also helps teams communicate more clearly, which in turn improves company culture. With better tools, like those for automation, developers feel empowered to do their best work and feel fulfilled because they're not stuck working on repetitive tasks.

This is where GitHub Actions comes in. **Using Actions for CI/CD can reduce build time from 80 minutes to 10**. This time back gives developers the ability to focus on code contributions without the distraction of repetitive manual tasks. There are more than 13,000 free Actions in the GitHub Marketplace that your teams can use for everything from building out a CI/CD workflow to closing stale issues and greeting new members to a repository.

Once large repositories start using Actions, teams merge almost 2x more pull requests per day than before (61% increase) and they merge 31% faster.[13]

How P&G utilizes GitHub Actions

With such a huge roster of brands—including Bounty, Crest, Dawn, Gillette, and Tide—P&G needs to be on the cusp of innovation, especially when it comes to the supply chain, in order to stay competitive. Their teams use GitHub Actions to build

12: Srivastava, Shivam, Kartik Trehan, Dilip Wagle, and Jane Wang. 2020. "How software developers can drive business growth." McKinsey. https://www.mckinsey.com/industries/technology-media-and-telecommunications/our-insights/developer-velocity-how-software-excellence-fuels-business-performance

13: "Writing and shipping code faster | The State of the Octoverse." 2021. GitHub. https://octoverse.github.com/writing-code-faster/#search-improves-software-development

automation, cloud deployments, and infrastructure as code. Not only does this save their developers time managing configurations and tooling, but it helps newer, less experienced developers automate tasks more easily. This means teams have more time to innovate and keep P&G on top of efficient e-commerce sales.

> *"GitHub is an accelerator. It reduces the amount of time it takes for us to bring value to our customers and consumers."*
>
> Danilo Suntal // Agile & DevOps lead, P&G

Making security and DevSecOps part of your workflow

Application security is a challenge for every business, at all levels. Despite significant investment into security tools, most organizations can't keep up with or remediate their security risk, leaving them vulnerable. Seventy-five percent of attacks in 2020 used vulnerabilities that were at least two years old.[14]

Adopting the right approach to DevSecOps reduces risk and allows teams to deploy quickly and securely. Right now, 32% of developers spend up to 10 hours a week fixing bugs instead of writing code.[15] Think of what they could be accomplishing if they had that time back.

With GitHub Enterprise's inclusive approach to application security, you can natively integrate security into every step of the developer workflow, no matter your role at the organization. The result: end-to-end security without sacrificing speed.

> *"GitHub helps us ensure that we have our security controls baked into our pipelines all the way from the first line of code we're writing."*
>
> Miguel El Lakkis // Chief Information Security Officer, Dow Jones

Here are two GitHub Enterprise features that'll help secure your business:

Security Overview: Security overview provides a centralized view of risk for application security teams, development leaders, and developers who work across hundreds—or even thousands—of repositories. In the security overview dashboards, you can see code scanning, Dependabot, and secret scanning

14: "2021 Cyber Security Report." 2021. Check Point Software. https://www.checkpoint.com/pages/cyber-security-report-2021/
15: "2021 Report | The State of Software Code." 2021. https://try.rollbar.com/report/the-state-of-software-code-report/

alerts across every repository you have access to. It also shows unknown risks where security features aren't yet enabled.

Dependabot: Integrated directly into the developer workflow, Dependabot keeps your projects secure and up to date by monitoring for vulnerable and out-of-date third-party components. If a suggested update is found, Dependabot will automatically open a pull request with recommended fixes.

From customizable branch protection rules and auditable reviews, to status checks, two-factor authentication, and role-based access control, GitHub Enterprise is secure by design and helps your business stay secure both internally and externally.

The security upgrade: GitHub Advanced Security

GHAS natively embeds security into the developer workflow—enabling you to secure your software supply chain and proprietary code across the software lifecycle.

With GHAS, there are automated security checks at every pull request, and identified security issues are shared immediately within the familiar GitHub workflow. This empowers teams to fix vulnerabilities in minutes, not months. Having these native security capabilities integrated into the GitHub platform gives you:

- Detailed changelogs across security issues and fixes
- Visibility into your security posture across code, secrets, and supply chain
- Crowd-sourced security intelligence from millions of developers and security researchers around the globe

This complete, native, and automated approach enhances productivity, reduces risk, and improves time to market. And because most developers are already familiar with GitHub, GHAS also eliminates the need to learn new security tools—increasing speed, collaboration, and developer satisfaction.

Mercari is an e-commerce platform that connects millions of people across the U.S. to shop and sell everything from fashion to toys to sporting goods

and electronics. To keep their business running efficiently and securely, their teams use GitHub Enterprise and GitHub Advanced Security. Before switching to GitHub, developers had to manage multiple tools for finding secrets and vulnerabilities. But GitHub Advanced Security has all the tools all in one place. With secret scanning as part of the developer workflow, they can spot problems right away, instead of finding them months later when they might not remember the context of that particular bug.

> *"Adopting code scanning across the org is definitely something that GitHub makes much easier. It would have been a struggle otherwise because integrating tools into everyone's workflow is complicated and time consuming. Having secret scanning and CodeQL baked into the GitHub platform makes things much, much easier for us."*
>
> Nikolay Elenkov // Director of Product Security, Mercari

Conclusion

There's a reason 90% of Fortune 100 companies use GitHub Enterprise. Perhaps the most important is that the end-to-end platform makes the workday seamless and efficient for developers, which is necessary for businesses in any industry that want to keep up with the increasing pace of the market. From minimizing context switching and simplifying the tech stack to increasing collaboration and saving time through code reuse and automation, GitHub Enterprise leads with the developer experience in mind—and helps them stay secure throughout every step of their workflow.

At GitHub, we are able to meet developers where they are: on a platform they already know and love, which is imperative when it comes to attracting and retaining top talent. As you build your team to take your technology and business to the next level, choose GitHub Enterprise as your long-term partner and let the experimentation and innovation begin.

Got questions about GitHub Enterprise?

We can help.

Visit our GitHub Enterprise page
or connect with our sales team.

WRITTEN BY GITHUB WITH ♥