Mattermost

Building a Developer-Centric Culture

Great technical teams are built, retained, and scaled with strong developer-centric cultures. Ensuring that people feel safe, respected, and empowered to do their best work is essential to building any team, and developers are no exception!

Culture can be a tricky thing, though. It's not something that can be measured, and while it can't truly be enforced from the top down, a healthy culture requires the right environment to grow and scale. Before you start adding headcount, lay the foundation of a developer-centric team culture that devs will love to be part of.

What do great developer cultures have in common?

While many teams struggle to build great developer cultures, some organizations do seem to have things figured out. Let's take a look at some of the characteristics that developer-centric teams have in common.

They empower developers to work autonomously together

Enabling your team to work autonomously helps them feel ownership over their work. It also helps keep their work interesting and engaging over time. Give your team the agency they need to move quickly, collaborate effectively, and adopt new tools freely.

Keep in mind that "autonomy" doesn't mean that every developer is a lone wolf, acting on

their own. Rather, it means that the team has the flexibility to build and collaborate however they need to to meet the project's needs.

Spotify: Organizational structure designed for autonomy

As the engineering organization at Spotify grew, its early Agile processes and practices began to slow down development. Spotify shifted to a model of "Autonomous Squads," each of which has its own mission and end-to-end responsibility for whatever they're building.

The team has control over how they organize themselves and achieve their goals. This model helps developers at Spotify stay motivated, keeps decision-making moving fast, and minimizes handoff and waiting time. Learn more about Spotify's engineering culture »

Put it into practice

- ☑ Prioritize building a functional codebase
- ☑ Reduce technical debt
- ☑ Normalize documentation and code commenting
- ☑ Create clear handoff and well-documented processes
- ☑ Reduce friction with your developers' preferred tool stack

Diversity and inclusion are more than just metrics

Diversity is a buzzy word right now, and it can sometimes feel like organizations talk the diversity talk, but don't walk the walk. But hiring a more diverse set of candidates is only the first step toward building a diverse and inclusive organization.

Developer-centric organizations must give explicit, loud support to underrepresented folks within the company and user base. Developers aren't always eager or skilled at voicing their opinions. If the quietest person in the room feels adequately understood and represented, the rest of the team is much more likely to feel the same way.

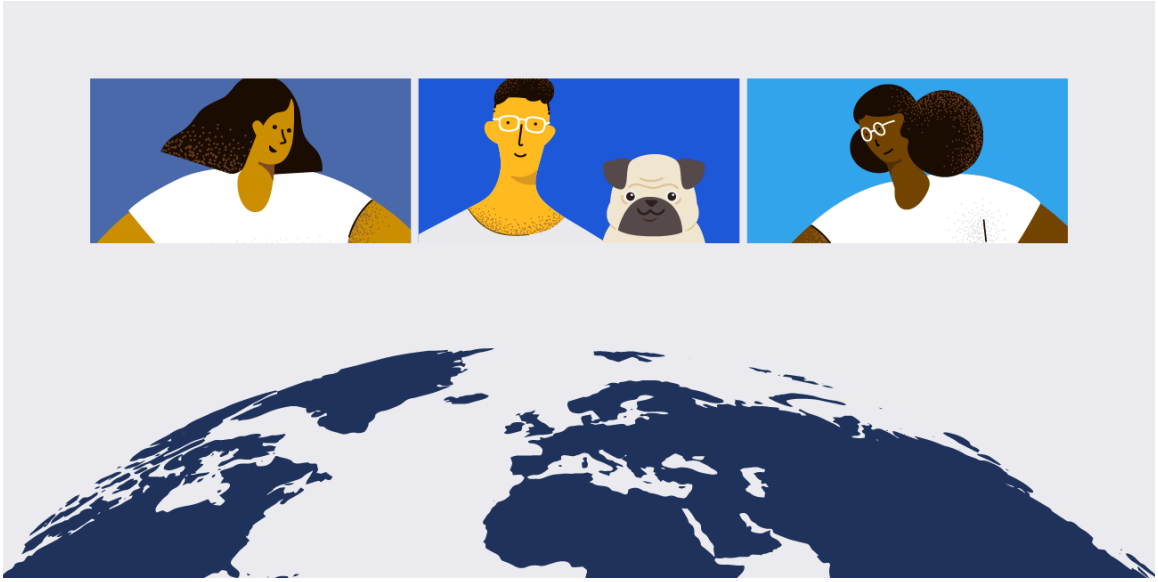
Asana: Programmatic diversity and inclusion initiatives that scale

From the early days, Asana committed to cultivating a diverse team. But they didn't simply set a hiring quota and call it "done." Instead, they built a fully-fledged, action-oriented program that starts with their hiring pipeline and processes, and continues throughout the entire lifecycle of employees at the organization.

Asana also hired a Head of Diversity and Inclusion to focus full-time on maintaining and extending their values as they grow the team. Now, the role includes partnerships with organizations that work with underrepresented demographics, benefits programs designed to bridge the gap for groups that sometimes get lost in the shuffle (e.g., mental health and parental leave), and employee support groups to help team members connect. Learn more about Asana's diversity and inclusion program »

Put it into practice

- ☑ Create spaces for team members to connect with each other
- ☑ Hire team leads that champion all team members
- ☑ Document your processes and measure your impact



They encourage career growth and provide a roadmap to success

Development work can be a grind. Far too often, developers get so bogged down in their day-to-day work that true career growth is overlooked. Great developer cultures counteract this in a few important ways:

1. **They provide clear pathways to growth.** They document what's needed to get to the "next level" and make that documentation available to every team member.
2. **They create space for growth.** They work to ensure the team has the right tools and processes in place to work efficiently and prioritize projects. They also hire appropriately to ensure the team isn't overloaded.
3. **They give every member of the team an advocate.** Their leaders speak up for folks who might not feel they have a voice and regularly check in with the team members to ensure they're moving forward and upward.
4. **They support non-traditional pathways.** They recognize that not everyone has the same background or career goals and create support systems for hiring and promoting people who don't fit the mold, but who have great skills and experience to bring to the table.

Patreon: Honoring a commitment to professional growth

Like many organizations in their early days, the Patreon team lacked clear criteria for growth within their engineering organizations. After realizing this, Patreon developed a comprehensive guide to leveling up engineers to provide a clear, actionable roadmap for team members looking to grow. This document helps support early-stage developers navigate the organization, and provides guidance for what they need to do to climb the ladder internally. Learn more about Patreon's engineer leveling »

Put it into practice

- ☑ Create opportunities for professional development — engagement with open source tools, training sessions for new technologies, funding for continued education, etc.

- ☑ Support regular knowledge-sharing activities, such as Lunch and Learns, pair coding, technical blogging, and process documentation, etc.
- ☑ Increase visibility into organization charts and promotional paths for all team members
- ☑ Encourage 1-on-1s with managers that include career growth and goals, not just day-to-day responsibilities

They leave nothing unwritten

Tribal knowledge can be very helpful — as long as you're a fully entrenched member of the tribe. For fast-growing teams — and especially for distributed ones — those things that everyone "just knows" don't always filter through the ranks as thoroughly as they might in a shared office. Even when that knowledge does make its way through the team, it doesn't always translate across roles, levels, and departments consistently.

Communication is essential to any team. For technical organizations, it's even more critical. Great documentation of everything from daily workflows and processes to career pathways and review cycles is essential to high-performing technical teams.

Keep in mind that this documentation doesn't appear out of thin air — everyone on your team should be empowered and encouraged to create, contribute to, and iterate on documentation of processes and best practices as part of their day-to-day work. Incorporating a new tool into your workflow? Add it to the docs. Found a way to streamline a repeatable process? Update the playbook. After a while, documentation will become second nature, and the whole team will benefit from it.

Great teams don't leave the team's work "unwritten" either. Public recognition for hard work, contributions, and growth can increase visibility into what everyone is doing and bolster morale before, during, and after working on the toughest problems.

GitLab: Increasing information accessibility and transparency for all

GitLab has built a wildly successful DevOps platform in just a few years — all with a fully remote, globally distributed team. At the core of their organizational operations is a documentation-first culture. Their single source of truth is an encyclopedic public handbook that would clock in at over 8,000 pages if printed in its entirety.

The goal of this approach is simple: to provide every member of the team with all the information they need to get their work done, no matter what time zone they're working from. Learn more about GitLab's handbook-centric documentation »

Put it into practice

- ☑ Consolidate all your how-tos, FAQs, and guides in one place
- ☑ Make sure your messaging platform is searchable
- ☑ Default to public channels for conversations — not direct messages
- ☑ Create playbooks or runbooks for repeated processes
- ☑ Have retrospectives as a regular part of your processes and workflows
- ☑ Include formal recognition into the rhythm of launches, work cycles, etc.

They use cutting-edge and open source technologies

A great developer can make the most of whatever technology you throw at them, right? Guess again. A StackOverflow survey of over 100,000 developers found that when it comes to deciding where to work, technology is a major deciding factor for developers, trailing only compensation and benefits as the top priority.

So, how do you know what tools are the right ones for your team? Part of the equation is letting them tell you! According to Google's DevOps Research and Assessment (DORA) program, DevOps teams that are empowered to choose their own tools perform better than those that don't have any say:

> *"Allowing teams to choose tools doesn't mean each team is given free rein to select any tool they want. Introducing technologies without any constraints can increase technical debt and fragility. However, when you combine tool choice with other capabilities—for example, a full view of the system, fast feedback, and the understanding that they are responsible for the code that they write—it helps your technologists make wise decisions about tools they will use and need to support."*

In fact, DORA's research has found that one of the most common indicators of job satisfaction is whether people have the tools they need to be successful.

Open source technology in particular is an essential part of many developers' preferred tech stack because of its flexibility and customizability. This preference points to an important aspect of empowering developer teams: They want and need tools that can adapt to their specific workflows and help them work as effectively as possible.

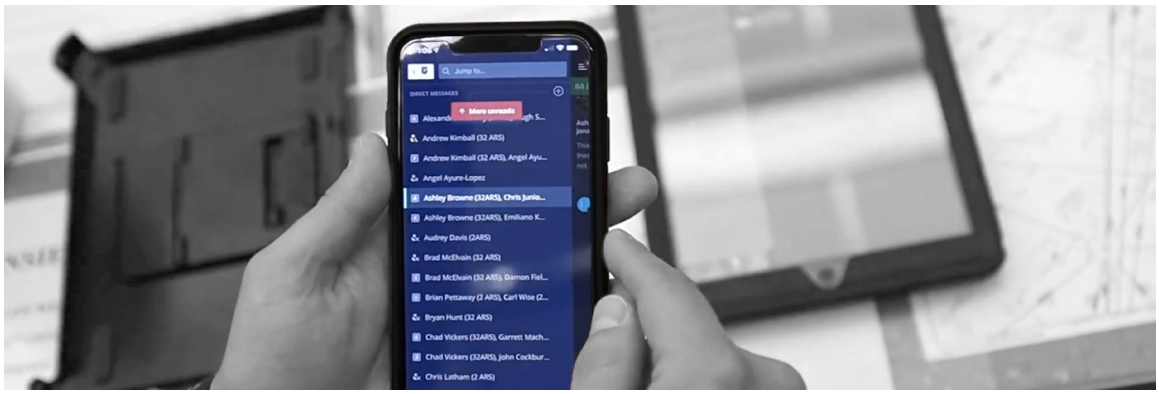
Did You Know?

The highest in-demand skill for software engineers in 2021 was Redux.js, landing engineers nearly 3x more interview requests than average. Experience with Google Cloud, AWS, React.js, Go, and Scala were also highly sought-after — all of these skills garnered at least 2x more interview requests for developers.

United States Department of Defense: Enabling open source tool adoption

The Department of Defense (DOD) might not be the first organization you think of when you think of high-performing tech organizations. But in the past few years, the DOD has gone through a serious digital transformation that has changed the face of software engineering for the entire organization.

Platform One, the U.S. Department of Defense's DevSecOps Enterprise Services team, adopted a new approach to streamline the approval and procurement of software solutions. Now, teams within the DOD are able to spin up new tools much more rapidly and cost-effectively than before, allowing them to leverage cutting-edge, cloud-based and open source tools that were previously unavailable to them. As a result, they've cut their software spending by millions of dollars and enabled their team to move many times faster than before by revolutionizing their approach to software development. Learn more about the Department of Defense's digital transformation »



Put it into practice

- ☑ Work with your team to understand the technologies they need and want
- ☑ Establish clear, quick procurement guidelines and processes
- ☑ Enable your team to experiment with open source tools and other cutting-edge technologies

THE BOTTOM LINE

Great developer cultures are cultures of innovation

There's no one "right" way to build a great developer culture, and it's impossible to completely overhaul your culture overnight. But even small changes can add up to a big difference over time. Former StackOverflow COO Jeff Szczepanski shared his thoughts on hiring great developers:

> *"[W]e do our best to give our programmers what they need, whether that's a high-end keyboard, three monitors, or the ability to work remotely. We also have an open-door policy for new ideas and feedback across the organization. We've found that this combination of tools and transparency makes all the difference.* ***The bottom line is that if you can't treat developers like critical stakeholders in your company's strategy, they'll find another company that will."***

At the end of the day, great developer cultures are cultures that encourage innovation. They give teams opportunities to think outside the box, be creative with their work, collaborate in unique ways, and give back to the community.

Adopt robust, flexible workflows, and tools that reduce toil. Eliminate timesinks and give the team the bandwidth to innovate.

Make "safe" spaces for people to bring up new, unusual, or "weird" ideas. Encourage projects like Open Source Fridays and hackathons that give developers space to think creatively.

Prioritize work-life balance and provide opportunities for growth. Keep the team engaged, refreshed, and thinking about the future.

Now that you have a better picture of the type of culture you want to foster at your organization, it's time to start bringing in new people to help you realize your vision and get your technical team to the next level.

Mattermost

Learn more about building your technical team

Want to learn more about building and retaining a high-performing technical team? Now that you have a better picture of the type of culture you want to foster at your organization, it's time to start bringing in new people to help you realize your vision and get your technical team to the next level. Check out the next installment in our series of team-building resources: **Hiring Technical Talent: Finding the Right People for Your Developer Team.**

Read More