Gartner

Gartner Research

How to Lead High-Performing Software Engineering Teams

Akis Sklavounakis

How to Lead High-Performing Software Engineering Teams

2 May 2024 - ID G00773076 - 11 min read
By: Akis Sklavounakis
Initiatives: Software Engineering Practices; Build and Deliver Software to Drive Business Results; Build a World-Class Software Engineering Organization

Software engineering leaders cannot simply form teams with talented people and hope that they deliver. Rather, they must drive high performance within their teams by fostering psychological safety, improving developer experience and connecting people and teams.

Overview

Key Findings

- Software engineers are unwilling to take interpersonal risks that drive learning, collaboration and innovation, because software engineering leaders often focus only on developing technical skills within their teams, while overlooking human factors such as psychological safety.
- Software engineers often experience friction in their delivery workflows due to distractions in their work environment, nondevelopment work tasks and inadequate tools.
- Software engineering teams do not deliver valuable digital products in isolation. Complex product delivery is a “team of teams” sport that requires organizationwide alignment and support.

Recommendations

To lead high-performing teams, software engineering leaders should:

- Foster psychological safety by framing mistakes as opportunities to learn, acknowledging their own mistakes, modeling curiosity and cultivating belonging. They must create an agile learning environment which leads to high performance.

- Optimize developer experience by improving developer journeys, optimizing for creative work and providing opportunities to make a meaningful impact.
- Connect people and teams by adopting the Connector manager approach.

Introduction

Software engineering leaders are at the nexus of business and technology, between strategy and implementation. Your software engineering organization is delivering critical digital technology initiatives. Gartner's annual Board of Directors survey found that digital technology initiatives are the top strategic business priority for the second consecutive year. [1]

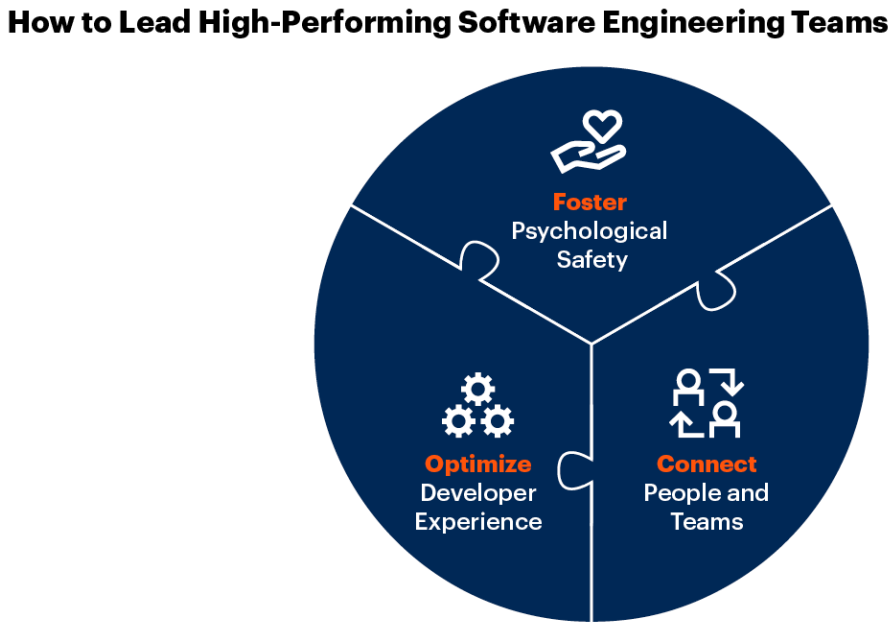
The board of directors depends on your leadership, your practices and your empathy to deliver customer value via innovative digital products.

Software engineering leaders often struggle to deliver valuable digital products due to talent and skills shortages. Finding talented software engineers is a great challenge. Our surveys show that demand for software engineering positions has increased by 10% in the previous 12 months. On average, there are fewer than eight job applicants for software engineering roles, compared to 96 for all other roles, showing that competition for software engineering talent is fierce. [2]

However, even when you find people with the skills you need, you cannot simply form teams with talented people and hope that they deliver. Instead, you must create the right environment and implement the right processes and practices to lead teams to high productivity, creativity and resilience. This research will help your organization meet its ambitious goals for digital initiatives.

Which practices should you adopt to lead high-performing teams? Software engineering leaders must foster psychological safety, optimize developer experience and connect people and teams to position their organization for success in delivering valuable, innovative digital products (see Figure 1).

Figure 1: How to Lead High-Performing Software Engineering Teams



Source: Gartner
773076_C

Gartner

Analysis

Foster Psychological Safety

We often focus on technical skills and work experience when building teams, but overlook psychological safety — the single most important factor for high team performance. [3]

A Gartner survey found that when employees work in a psychologically safe environment, discretionary effort can improve by up to 24%. [4] In other words, employees who feel psychologically safe are more willing to go above and beyond in their daily efforts by helping others with heavy workloads, volunteering for additional duties or looking for ways to perform the job more effectively.

> **Team psychological safety is "a shared belief held by members of a team that the team is safe for interpersonal risk taking."**
>
> *— Amy Edmondson, Harvard University professor and author*

Psychological safety creates a virtuous cycle where software engineers are more willing to speak up and share their viewpoints. Their increased authenticity and participation drives high team performance, as engineers feel more comfortable with collaborating, learning and innovating as a team.

To foster a psychologically safe environment, software engineering leaders should:

Frame failure as an opportunity to learn. Failure can be essential for learning and growth. You should distinguish between blameworthy failures — that is, failures that result from lack of competency, a policy violation or negligence — and praiseworthy failures. Share these praiseworthy failures with the team, and focus on the key lessons that your team members learned from the failure.

To determine if a failure is praiseworthy, ask yourself:

- Did the failure result from trying something new, testing a hypothesis or exploring a new opportunity?
- Did it occur because you or the team did not anticipate changing circumstances?
- Did it stem from tackling a particularly complex task or collaborating with many or new stakeholders?

Acknowledge their own mistakes. Demonstrate vulnerability by sharing one of your recent praiseworthy failures, and outline how the lessons you learned led to positive impact. Setting an example for employees will encourage them to admit their own mistakes and take greater interpersonal risks without fear of retribution.

Model curiosity. Ask lots of questions in both one-on-one and team discussions.

When discussing performance or career-related topics with individual team members, ask questions such as:

- What do you want to accomplish? What impact will you have?
- How will you accomplish your goals? What obstacles would prevent you from accomplishing this?
- What does success look like?
- How can I help you achieve your goals?

When leading team discussions about the organization's goals, ask questions such as:

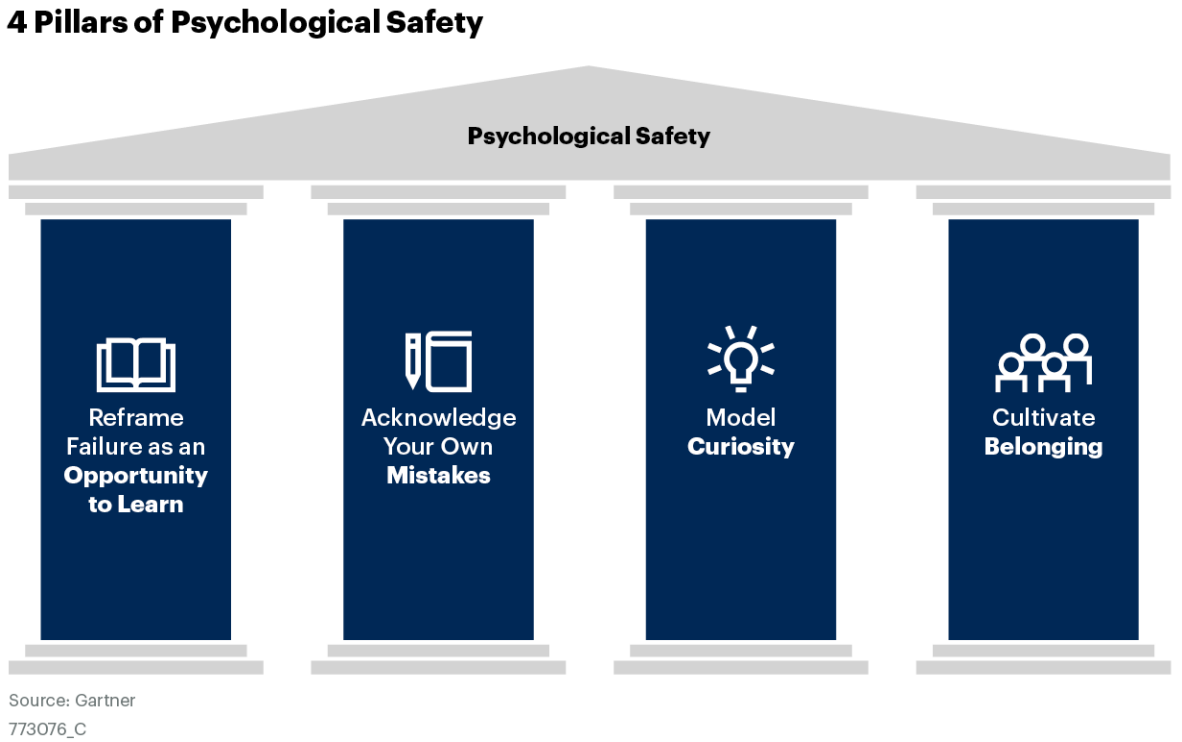
- How might our roles and responsibilities need to change to best support organizational goals?
- What potential ideas and strategies can help us achieve these goals?
- As a team, how can we overcome possible barriers to supporting these goals?
- What are the unique strengths and weaknesses of our team? How might these qualities affect our ability to support organizational goals in our daily work?

Cultivate a sense of belonging. Incorporate these practices within your teams:

- **Create shared goals:** Lead a workshop with the team to craft shared, team-level goals. These shared objectives will provide a common sense of purpose and encourage team members to collaborate and hold each other accountable.
- **Appreciate and understand cultural differences:** Encourage team members to share personal stories that promote a better understanding of their culture. Acknowledge all public and religious holidays that team members observe. When scheduling meetings, be considerate of the time zones in which all team members are based.
- **Celebrate success:** Provide public recognition when the team performs well. This includes both celebrating individual accolades from managers and peers, and praising the team for its collective performance.

Figure 2 illustrates the four pillars of psychological safety.

Figure 2: 4 Pillars of Psychological Safety



See Quick Answer: How Do We Build Psychological Safety in Our Software Engineering Teams? for further advice.

Psychological safety is a key component of a continuous learning culture. You should create learning opportunities by starting an agile learning program. Agile learning offers an effective way for employees and enterprises to acquire new skills, thrive amid future-of-work shifts and deliver successful business outcomes. Learning faster than other organizations is the only sustainable advantage in today's talent market. To foster a culture of agile learning, start by shifting the mindset of employees from waiting for learning to happen to proactively seeking learning opportunities by creating an environment that supports learning. Make learning part of the job, not time away from or in addition to it. Carve out some common time for learning activities each week. It shows management's commitment to learning and willingness to invest productive capacity to improve engineering skills and competencies. See How to Establish a Reskilling/Upskilling Talent Development Program for Software Engineering for further details.

Optimize Developer Experience

Developer experience refers to all aspects of interactions between developers and the tools, platforms, processes and people they work with, to develop and deliver software products and services. A frustrating developer experience is a common challenge, and it often undermines your team member's productivity and job satisfaction.

A Stack Overflow developer survey found that a few of the greatest challenges to developer productivity are a distracting work environment, being tasked with nondevelopment work and inadequate access to necessary tools[5]. To overcome these obstacles and optimize the experience for your developers, software engineering leaders must focus on three core pillars:

- **Improving developer journeys** by:
  - Streamlining the onboarding experience
  - Implementing self-service DevOps workflows
  - Establishing accelerated feedback loops
- **Optimizing for creative work** by:
  - Safeguarding focus time for engineers to do deep work
  - Building a collaborative work environment
  - Automating repetitive tasks to reduce toil
- **Providing opportunities to make a meaningful impact** by:
  - Contributing to the wider community
  - Fostering a safe environment for experimentation and learning
  - Eliciting and sharing direct feedback from end users

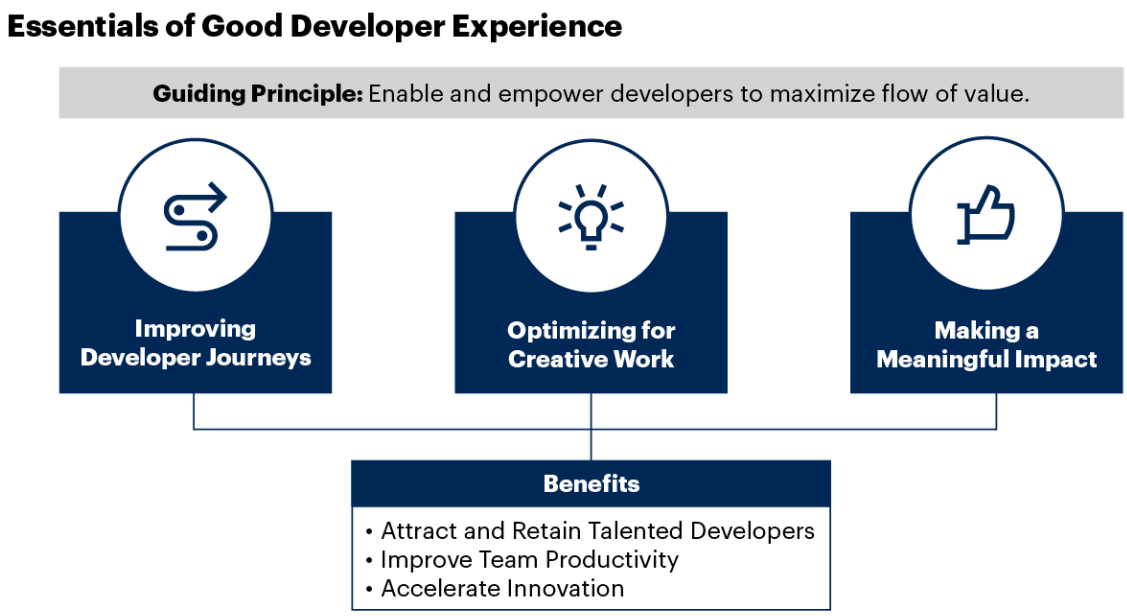
Speak with your teams to understand their needs and discover the points of friction and frustration in the development process. Establish regular developer surveys to capture objective and subjective metrics of happiness and productivity within teams, and take actions to improve them. Treat software engineers like customers and perform customer journey mapping with an aim to provide an environment where they can do their best work. Friction makes it difficult for software engineering teams to meet business expectations for speed and responsiveness. The most significant source of software engineering friction is dependencies. Dependencies are caused when software engineering teams are required to hand off work, rely on other roles and teams to get work done, or depend on shared technologies to achieve objectives.

To minimize dependencies, software engineering leaders must:

- Structure teams with people who have all the skills the team needs, including roles that are dedicated to managing and coordinating dependencies.
- Provide teams with access to all systems, tools and resources they need to get their jobs done.
- Implement platform engineering to streamline the complete software development life cycle — from ideation to realization of customer value.
- Work with peers and other leaders in the organization to align goals, incentives and objectives across functions and practices.

Figure 3 highlights the three essential elements of good developer experience.

Figure 3: Essentials of Good Developer Experience



See A Software Engineering Leader's Guide to Improving Developer Experience for detailed advice.

Connect People and Teams

Software engineering teams do not deliver valuable digital products in isolation. Complex product delivery is not only a team sport, but an organizationwide game. It is also a long-term game. Short-term increases in productivity may not always be sustainable, as team members can overexert themselves — leading to burnout and attrition.

To achieve sustainable high performance in their teams, software engineering leaders must provide team members with high-quality connections that enhance both their performance and their mental well-being.

The Connector Manager approach boosts sustainable performance by as much as 45%. [6]

Software engineering leaders and their team leaders should take the Connector manager approach by creating three types of connections:

Employee connection. Get to know each of your team members on a personal level and partner with them to create individualized development goals that align with their needs and interests. Ask questions such as:

- What are your long-term career aspirations?
- What support do you need from me to achieve your goals?
- What specific skills do you want to learn, either within your role or in future roles?

Team connection. Promote reciprocal and real-time development within the team by:

- Assessing each team member's strengths and skills proficiency. Identify areas that could improve with greater collaboration between team members.
- Using skill profiles to pair employees with opposing strengths and development needs.
- Encouraging collaborative practices such as pair programming for fast and high-impact learning.

Organization connection. Make quality development connections outside of the team. Prioritize the quality of these connections, rather than focusing on quantity. Build this ecosystem of connections by:

- Establishing informal cross-function learning programs, such as communities of practice.
- Matching employees across teams with complementary strengths to develop the needed skills and the right behaviors.
- Inviting individuals from all functions to volunteer as a coach or mentor to increase the likelihood that employees will find relevant help.

Evidence

[1] **2023 Gartner Board of Directors Survey on Business Strategy in an Uncertain World:** This survey was conducted to understand the new approaches adopted by nonexecutive boards of directors (BoDs) to drive growth in a rapidly changing business environment. The survey also sought to understand the BoDs' focus on investments in digital acceleration; sustainability; and diversity, equity and inclusion. The survey was conducted online from June through July 2022 among 281 respondents from North America, Latin America, Europe and Asia/Pacific. Respondents came from all industries, except governments, nonprofits, charities and NGOs, and from organizations with $50 million or more in annual revenue. Respondents were required to be a board director or a member of a corporate board of directors. If respondents served on multiple boards, they answered for the largest company, defined by its annual revenue, for which they are a board member. Disclaimer: The results of this survey do not represent global findings or the market as a whole, but reflect the sentiments of the respondents and companies surveyed.

[2] Gartner TalentNeuron is a labor market intelligence solution combining real-time data from job postings, censuses and government statistical bureaus with expert research and advice.

[3] Guide: Understand Team Effectiveness, Google re:Work.

[4] 2017 Gartner Culture Workforce Survey: This survey collected data from 7,502 global employees and HR leaders across 25 industries and 23 countries.

[5] Stack Overflow Developer Survey Results 2019, Stack Overflow.

[6] The 2020 Manager of the Future survey is a continuation of the 2017 Connector Manager research study. The survey simulates the work environment of the future and evaluates what the best managers are doing to prepare employees for it. It was conducted from 1 December 2020 to 18 December 2020. A total of 4,787 employees and managers were surveyed across 20 industries and 14 countries. The survey was designed and developed by Gartner's HR Practice research team.

Recommended by the Author

Some documents may not be available as part of your current Gartner subscription.
How to Turn a Team of Engineers Into a High-Performing Application Development Organization

Quick Answer: How Do We Build Psychological Safety in Our Software Engineering Teams?

How to Attract, Develop and Retain Great Software Engineering Talent

Software Engineering Teams Must Learn to Deliver More Value

Overcome Talent Shortages by Building Diverse, Equitable and Inclusive Software Engineering Teams

Quick Answer: How to Create a Frictionless Onboarding Experience for Software Engineers

Quick Answer: Create Compelling Job Descriptions to Attract Software Engineering Talent

Infographic: Reduce Software Engineering Friction by Minimizing Dependencies

Case Study: Developer-Centric Approach to Building a High-Value DevOps Toolchain (LexisNexis)

5 Tactics to Drive Greater Business Value From Software Delivery

---

Actionable, objective insight

Position your organization for success. Explore these additional complimentary resources and tools for IT and software engineering leaders:

eBook

Building and Implementing Modern Application Architecture and Technologies

Create a compelling business case to obtain funding and develop a migration plan.

Download Now

eBook

Reduce Friction Between Developer Productivity and Application Security

Uncover wasted developer time and reduce long wait and cycle times with secure-by-design platforms.

Download Now

eBook

How Software and D&A Leaders Can Drive AI Innovation

Boost collaboration, link technical execution to data and align staffing practices to your IT operating model.

Download Now

Tool

Gartner Software Engineering Leader Effectiveness Diagnostic

Use this diagnostic tool to benchmark your performance against your peers.

Learn more

Already a client?
Get access to even more resources in your client portal. Log In

Connect With Us

Get actionable, objective insight that drives smarter decisions and stronger performance on your mission-critical priorities. Contact us to become a client:

U.S.: 1 855 322 5484

International: +44 (0) 3300 296 946

Become a Client

Learn more about Gartner for IT Leaders
gartner.com/en/information-technology

Stay connected to the latest insight

Attend a Gartner conference
View Conference

Gartner®