STATE OF
ENGINEERING
MANAGEMENT

2022

CODING SANS

Contents

Introduction

Welcome to the State of Engineering Management 2022 report!

This year's report is a completely reworked version of the previously published State of Software Development series.

The title tells it all: it focuses on engineering management and goes deep into it to deliver valuable insights.

The report has five main sections.

The first focuses on self-improvement: uncovering fresh insights into how engineering leaders spend their limited time learning and improving themselves.

The following chapters cover company-related topics such as hiring and retaining talent and the ways tech companies build their engineering brand.

The report has a dedicated section for remote work, sharing new information about the current state and a sneak peek into the future plans.

The last chapter focuses on people management, incorporating every important aspect of leading teams.

There are highlights of differences between the top-performing teams and others, so you can learn and level up your engineering team.

I hope you will enjoy scrolling through the report and find some exciting insight you can apply right away.

Thank you,
Matt Wohlmuth
CEO, Coding Sans

Our Partners

LEVEL→UP
engineering
PODCAST

MEETUPFEED

LEVEL-UP ENGINEERING PODCAST

Level-up Engineering covers leadership topics bi-weekly, interviewing accomplished leaders from notable tech companies. They bring you the hottest engineering management stories out there!

MEETUPFEED.IO

MeetupFeed collects the latest talks about a wide range of software technologies, engineering management, and career development. The best source to keep up with everything software engineering.

CHECK THEM OUT ⟶

CHECK THEM OUT ⟶

CODING SANS
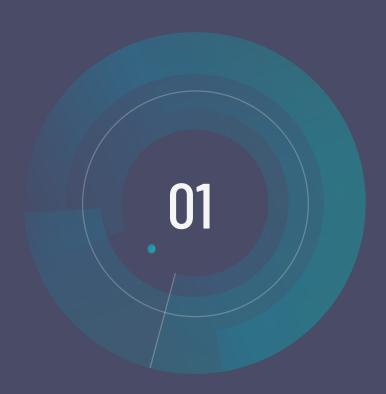
Leadership Self-improvement

"The journey of climbing the leadership ladder is about self-discovery." - Mike Seavers, VP of Online Development at Epic Games, Level-up Engineering episode #56.

It's so true. You must keep improving yourself as a person to become a better leader.

We bring you insight into how other leaders do this. What channels they use, what books they read, and what they do to share the knowledge they've acquired.

01

CODING SANS

What are the most important soft skills in leadership?

Communication	59.53%
Problem-solving	44.65%
Interpersonal skills	36.74%
Time management	31.63%
Teamwork	31.63%
Adaptability	22.33%
Leadership demeanor	20.93%
Creativity	11.16%
Attention to detail	11.16%
Work ethic	5.58%
Other	0.47%

Interview with **Gergely Hodicska "Felhő"** / ex-VP of Engineering at Bitrise

"Creating mental frames and looking at the world through these lenses is a natural part of human thinking."

LEVEL→UP
engineering

Podcast with Gergely

CHECK IT OUT ⟶

Control and impact

Earlier in my engineering leadership career, I was sensitive about other leaders crossing into my domain. I wanted to control everything, which made me less collaborative. This behavior limited the overall impact of my team, so it was a bad strategy.

At the time, I wasn't aware of my presence in these dynamics. In my mind, my impact was connected to my controlling function.

Even my boss perceived me as someone who wasn't great at collaborating on a broader level. As I started to understand what was behind this behavior, I started to give up this need for control and became better at cooperation. This led me to obtain a bigger responsibility as I became VP of Engineering at Ustream. Giving up control made my impact bigger.

Improve communication

As I've learned about behavior and soft skills, I realized that I don't act consciously in many situations. Gaining control over my instincts was a big challenge for me.
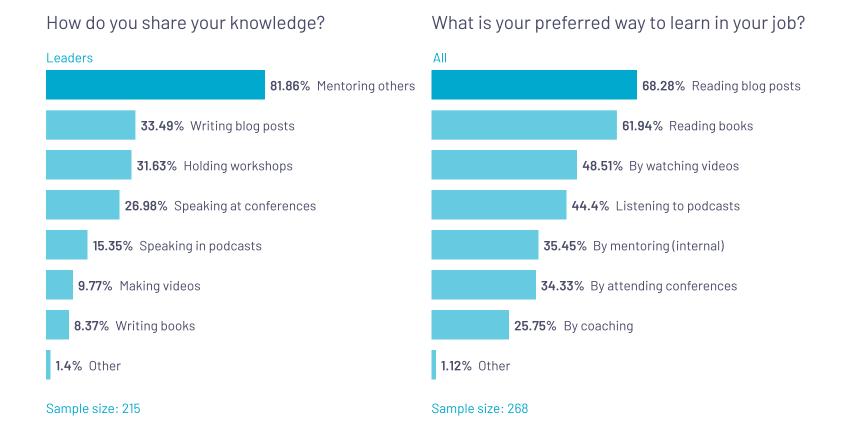
Giving and receiving feedback is closely related to this. When I receive constructive feedback, I tend to feel attacked. I get defensive and often start attacking myself, and since I'm good with words, I manage to win many of these arguments. It turns out, by winning these battles, I'm losing the war.

I realized that when giving feedback, I often aim to control others, show off my intellect, or dominate the discussion. Investing in self-awareness allowed me to use my autopilot less. These instinctive reactions often hurt more than they help.

Learning and Knowledge Sharing

How do you share your knowledge?

Leaders

81.86%	Mentoring others
33.49%	Writing blog posts
31.63%	Holding workshops
26.98%	Speaking at conferences
15.35%	Speaking in podcasts
9.77%	Making videos
8.37%	Writing books
1.4%	Other

Sample size: 215

What is your preferred way to learn in your job?

All

68.28%	Reading blog posts
61.94%	Reading books
48.51%	By watching videos
44.4%	Listening to podcasts
35.45%	By mentoring (internal)
34.33%	By attending conferences
25.75%	By coaching
1.12%	Other

Sample size: 268

CODING SANS

Book Recommendations

#1 Book for Leaders

We asked our participants what was the most influential book they'd read in their careers. The overwhelming favorite turned out to be:

Camille Fournier - **The Manager's Path**
This book requires no introduction; you can't go wrong with it. It's specifically aimed at engineering leaders, so if you haven't read it yet, it's time to pick it up.

Top Performers' Favorite

Only leaders who have a high performing team recommended this book, and it got the second most recommendations from this group only behind Camille's classic:

Dale Carnegie - **How to Win Friends and Influence People**
It's a true evergreen; getting close to 100 years old and it's still relevant. It was written with sales in mind, but it's worth taking a look at from a leadership point of view as well.

Honorable Mentions

We simply have to highlight a few more top recommendations from our participants. These books were all very close to the top spot on the leaders' recommendation list.

Stephen Covey - **The 7 Habits of Highly Effective People**
A widely known self-help book; deservedly so. This one isn't about building up your leadership skills, but it can help you fine-tune your habits and attitude that translate into making you a better leader.

L. David Marquet - **Turn the Ship Around!**
A true story of turning around the morale of a ship's crew of 100. It may not be about software engineering, but many of the leadership principles you can pick up definitely apply.

Kim Scott - **Radical Candor**
Another modern classic directly aimed at people in management. A must read about building and maintaining relationships in a professional environment.
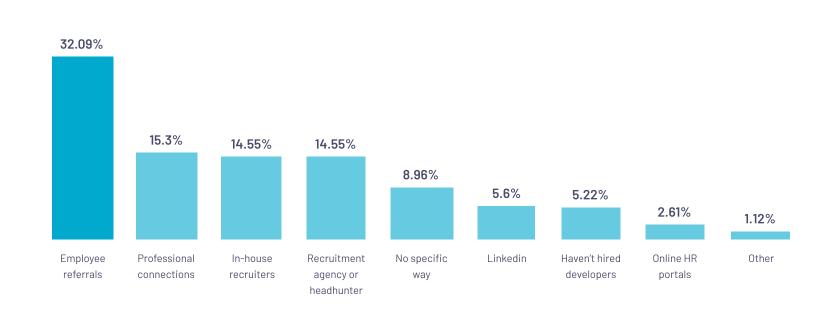
Hiring and Retention

Hiring is always on every leader's mind, and retention has become a more serious issue in a remote environment.

We bring you data on how other tech companies combat this challenge when it comes to hiring and retaining both developers and managers.
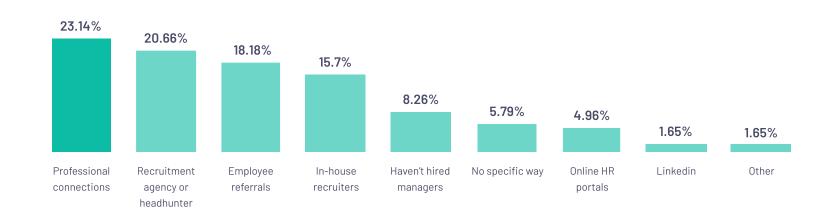
02

What's your most efficient method to hire **developers**?
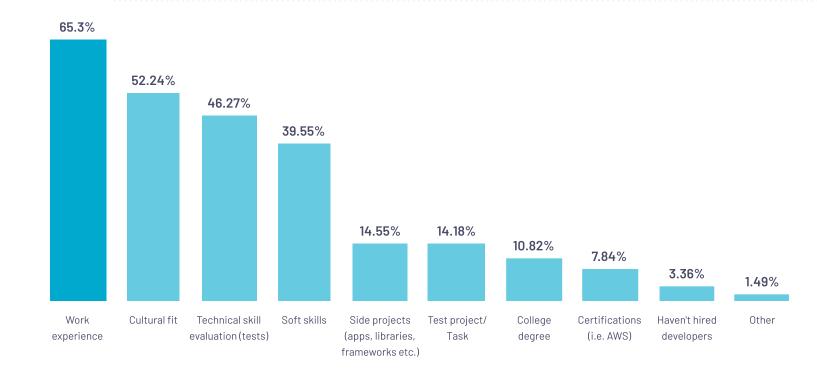
32.09% — Employee referrals
15.3% — Professional connections
14.55% — In-house recruiters
14.55% — Recruitment agency or headhunter
8.96% — No specific way
5.6% — Linkedin
5.22% — Haven't hired developers
2.61% — Online HR portals
1.12% — Other

What's your most efficient method to hire **engineering managers**?

23.14%	Professional connections
20.66%	Recruitment agency or headhunter
18.18%	Employee referrals
15.7%	In-house recruiters
8.26%	Haven't hired managers
5.79%	No specific way
4.96%	Online HR portals
1.65%	Linkedin
1.65%	Other

What are your most important hiring criteria for **developers**?

Bar chart showing hiring criteria percentages:

- Work experience: 65.3%
- Cultural fit: 52.24%
- Technical skill evaluation (tests): 46.27%
- Soft skills: 39.55%
- Side projects (apps, libraries, frameworks etc.): 14.55%
- Test project/Task: 14.18%
- College degree: 10.82%
- Certifications (i.e. AWS): 7.84%
- Haven't hired developers: 3.36%
- Other: 1.49%
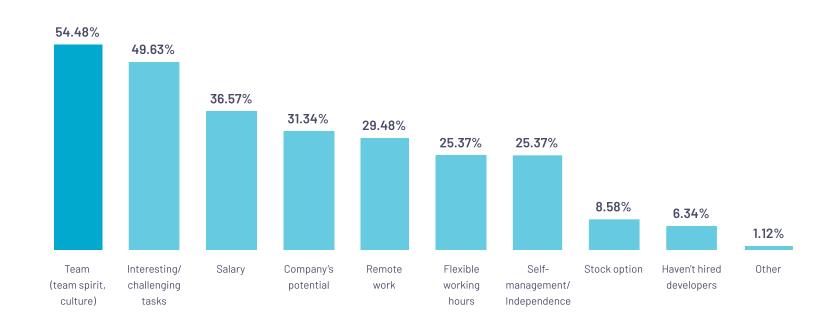
What are your most important hiring criteria for **engineering managers**?

31.78% Work experience
28.04% Soft skills
26.17% Cultural fit
11.68% Technical skill evaluation (tests)
11.21% Willingness to learn
7.48% Side projects
6.07% Certifications
5.61% College degree
3.74% Test project/ Task
1.87% Haven't hired managers
0.47% Other
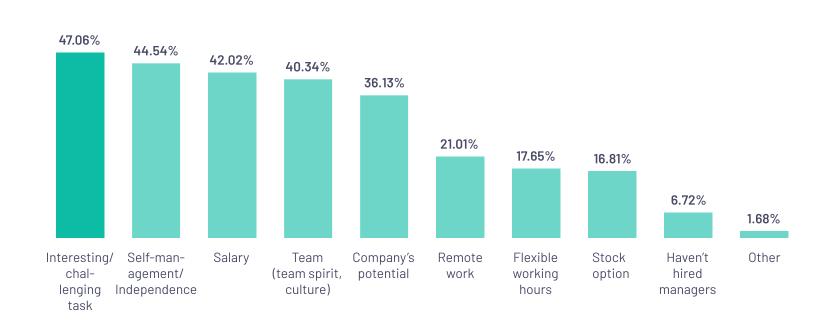
How do you attract new **developers** to your company?

- **54.48%** — Team (team spirit, culture)
- **49.63%** — Interesting/challenging tasks
- **36.57%** — Salary
- **31.34%** — Company's potential
- **29.48%** — Remote work
- **25.37%** — Flexible working hours
- **25.37%** — Self-management/Independence
- **8.58%** — Stock option
- **6.34%** — Haven't hired developers
- **1.12%** — Other
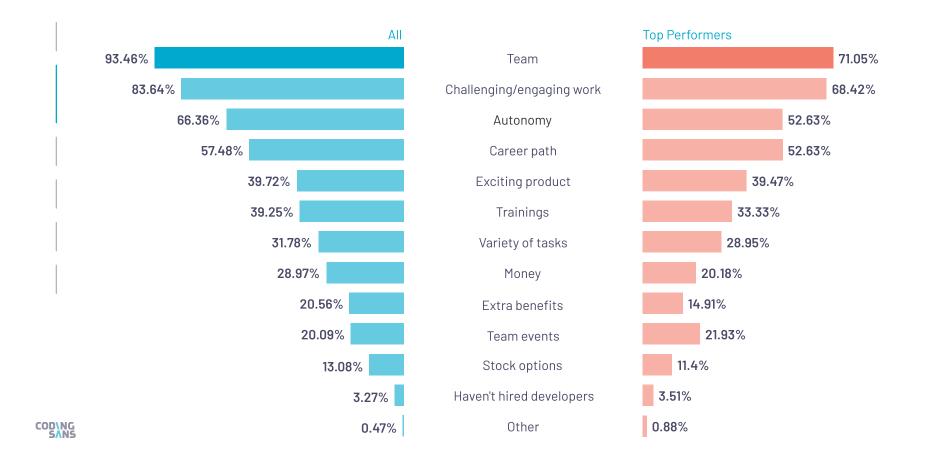
How do you attract new **engineering managers**?

- Interesting/challenging task — 47.06%
- Self-management/Independence — 44.54%
- Salary — 42.02%
- Team (team spirit, culture) — 40.34%
- Company's potential — 36.13%
- Remote work — 21.01%
- Flexible working hours — 17.65%
- Stock option — 16.81%
- Haven't hired managers — 6.72%
- Other — 1.68%
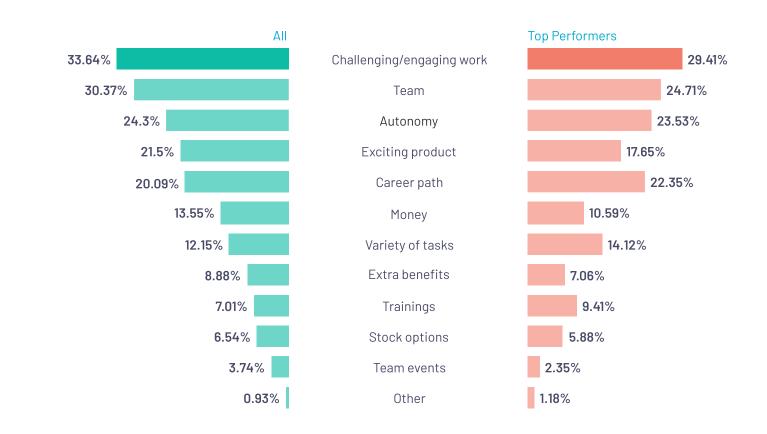
How do you keep **developers** motivated?

	All		Top Performers
93.46%		Team	71.05%
83.64%		Challenging/engaging work	68.42%
66.36%		Autonomy	52.63%
57.48%		Career path	52.63%
39.72%		Exciting product	39.47%
39.25%		Trainings	33.33%
31.78%		Variety of tasks	28.95%
28.97%		Money	20.18%
20.56%		Extra benefits	14.91%
20.09%		Team events	21.93%
13.08%		Stock options	11.4%
3.27%		Haven't hired developers	3.51%
0.47%		Other	0.88%

How do you keep **engineering managers** motivated?

	All		Top Performers
Challenging/engaging work	33.64%		29.41%
Team	30.37%		24.71%
Autonomy	24.3%		23.53%
Exciting product	21.5%		17.65%
Career path	20.09%		22.35%
Money	13.55%		10.59%
Variety of tasks	12.15%		14.12%
Extra benefits	8.88%		7.06%
Trainings	7.01%		9.41%
Stock options	6.54%		5.88%
Team events	3.74%		2.35%
Other	0.93%		1.18%

CODING SANS

Interview with **John Ford** / VP of Digital Workplace Technology at GoTo

"The sky's the limit with motivated, talented, engaged, capable people working well together. A team that isn't engaged, lacks expertise, or has lots of infighting will unavoidably underperform."

LEVEL→UP
engineering

Podcast with John

CHECK IT OUT ⟶

Engagement

Engagement is an essential aspect where engineering leadership plays a huge role. You need open, honest, transparent communication. Even if you can't tell everything about a specific topic, share as much information as you can.

On the other side, your people need to know that they can share confidential information with you, and you're not going to pass it on. This type of trust gives them the necessary psychological safety. You don't have to take it to a personal level, but your team needs to trust you on a professional level.

Building trust helps you improve engagement.

Empowerment

Delegate your engineers and empower them to do the work. You need to avoid creating a bottleneck by taking on work you don't have the time to get done.

Empowerment means that you let people make decisions in their area of expertise. You give them clear guidelines and frameworks, so they know what their scope is. Once this is all in place, you let them work.

In my experience, Dan Pink's theory about motivation is dead right. Software engineers are motivated by finding a purpose, having a degree of autonomy, and trying to do their best. Most people want to get better, and they want to do a good job.

Engineering leaders need to support this.

GoTo has found its purpose as the COVID-19 pandemic forced many companies into working remotely, and our product helps make this possible. Working for a cigarette company can be a valid choice, but a large-scale purpose is the icing on the cake when it comes to motivation.
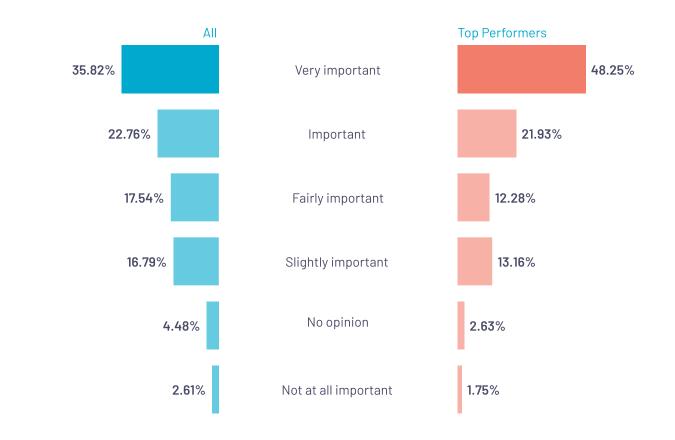
Engineering Brand and Diversity

Engineering brand is key to recruitment, but it's not at all obvious what you could or should put into building it. We bring you insight from other engineering leaders to highlight what they're doing about their brands.

Diversity and inclusion can be key components of your brand, and it's an important topic for many in tech.
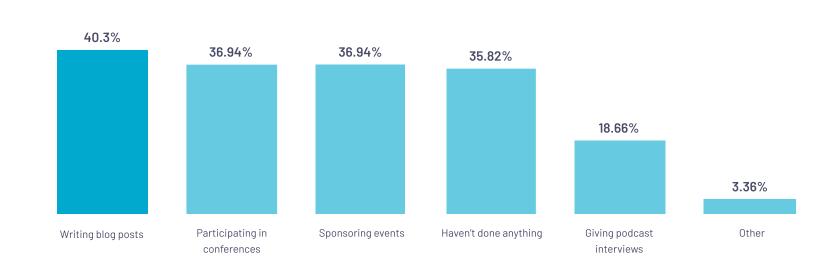
03

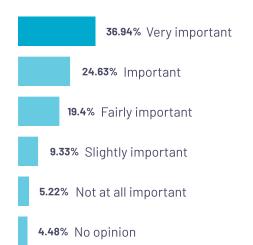CODING SANS

How important is the engineering brand for your company?

All

Top Performers

	All	Top Performers
Very important	35.82%	48.25%
Important	22.76%	21.93%
Fairly important	17.54%	12.28%
Slightly important	16.79%	13.16%
No opinion	4.48%	2.63%
Not at all important	2.61%	1.75%

What are you doing to build your engineering brand?

40.3%
Writing blog posts

36.94%
Participating in conferences

36.94%
Sponsoring events

35.82%
Haven't done anything

18.66%
Giving podcast interviews

3.36%
Other

Diversity and Inclusion

How important is diversity for your company?

- **36.94%** Very important
- **24.63%** Important
- **19.4%** Fairly important
- **9.33%** Slightly important
- **5.22%** Not at all important
- **4.48%** No opinion

What have you done to improve diversity and inclusion?

- **53.36%** Eliminating bias in the hiring process
- **44.03%** Promoting pay equality
- **34.7%** Eliminating bias in the promotion process
- **34.33%** Training programs
- **33.58%** Hiring remote employees
- **31.72%** Mixing up teams
- **27.99%** Strengthen anti-discriminatory policies
- **0.75%** Other

Interview with **Ola Sitarska** / Senior Development Manager at Shopify

"Once you're aware how your unconscious bias can affect you, you become more perceptive of it and you instantly get better at keeping it in check."

LEVEL→UP
engineering
PODCAST

Podcast with Ola

CHECK IT OUT ⟶

When you're running a business, you want to hire the best candidates. Increasing diversity and inclusion broadens the pool of qualified candidates applying to work at your company.

If the pool is limited, then your ability to hire the best is limited as well. You always want to work towards broadening the pool of exceptional people who may have gone through a non-traditional path in their career.

My story is an example of this: I've never studied computer science, but I've been programming since I was 12 years old. Broadening your perspective on who you are open to hire can get diverse talent on your team who you would have missed out on otherwise.

Bias in engineering

Many job ads for software engineers require six years of experience or even more with a programming language. This has little to do with what they will be doing on your team.

You hire developers to build a product, build a strong team and make an impact on your customers. Whether they've used a language for three years or five years won't make a big difference in the results.

Some engineers have only been working professionally for three years, but they have ten years of experience coding in that language. Others have only been coding for two years, but spend all their evenings learning about it, so their relevant skills are sharper than their CV suggests.
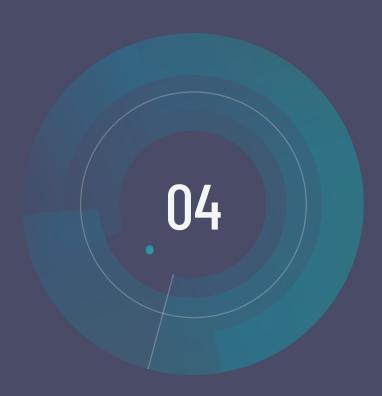
Many think if someone has only been a developer for two years, they can't be great. You need to measure their skills and objectively assess what they can bring to your team, instead of focusing on what they look like on paper.

Remote Work

Remote work has become the norm for most people in tech during the pandemic.

We bring you a sneak peek into the heads of engineering leaders, and see how many teams work remotely at the moment, whether they plan to change it, and what challenges they're facing.
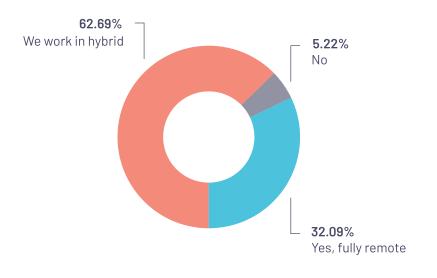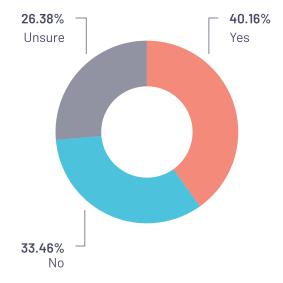
04

Remote Work and Future Plans
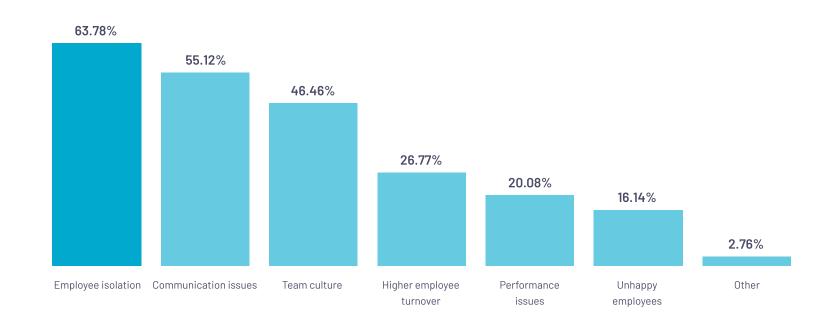
Does your team/company work remotely?

62.69%
We work in hybrid

5.22%
No

32.09%
Yes, fully remote

Do you have plans to return to the office?

26.38%
Unsure

40.16%
Yes

33.46%
No

What challenges have emerged due to sustained remote work?

63.78%	55.12%	46.46%	26.77%	20.08%	16.14%	2.76%
Employee isolation	Communication issues	Team culture	Higher employee turnover	Performance issues	Unhappy employees	Other

Interview with **Greg Dick** / Engineering Director at Hudl

"The sneakiest challenge is making the remote employees feel included in everything we do."

LEVEL→UP
engineering

Podcast with Greg

CHECK IT OUT ⟶

We've been making an effort to move all our conversations to Slack. It feels weird at times that I'm sitting three feet away from a colleague and I want to ask them a question or just tell a joke, but I post it on Slack instead. However, this helps me include the remote members of my team.

Story from Hudl

At one point, I joined a team after gaining some experience managing remote engineers, and the team was used to using direct messaging on Slack and one-on-one conversations a lot.

The team had one remote engineer.

I brought up the topic, "I think we should move most of our technical conversations to a Slack channel. Even if you think nobody cares about it, there's value in increasing the visibility of these." The remote engineer appreciated it, because sometimes days went by with only a couple of messages showing up in the team's channel, which made him feel isolated from the team.

So the team started posting everything into the team channel.

Even if it was a question specifically to one team member, we put it in the team channel. They often saw that the things they thought no one else cared about sparked interest in others.

It doesn't matter if team members don't have a strong opinion about a topic, knowing about it is valuable.

It started with work conversations, and naturally evolved to include non-work conversations. When someone told a joke, or saw a funny video on the internet, it became natural to share it in the Slack channel. Teams often create a separate work and a fun Slack channel for themselves, so neither one interrupts the other.

It slowly became natural for people to move verbal conversations to Slack. Nowadays, you often see people talk in the office about what's happening on Slack, and often check what's happening in the team's channels.

People Management

Management is all about people, and despite what some might think, software engineers are people too.
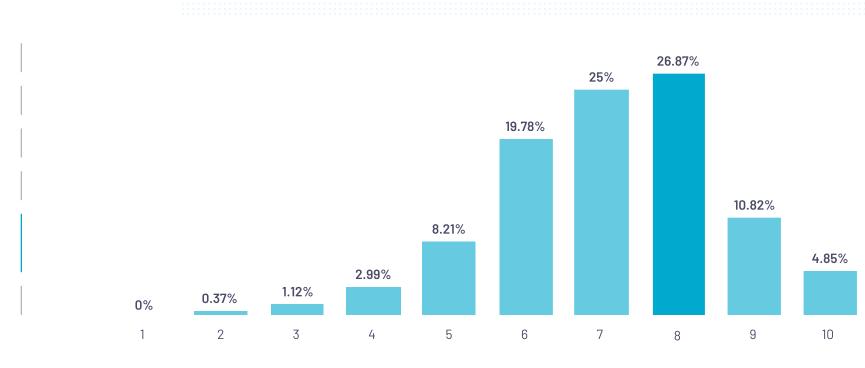
What can you do to tease the best performance out of them? What are common issues?

We're bringing you insight into how other leaders see it.

05
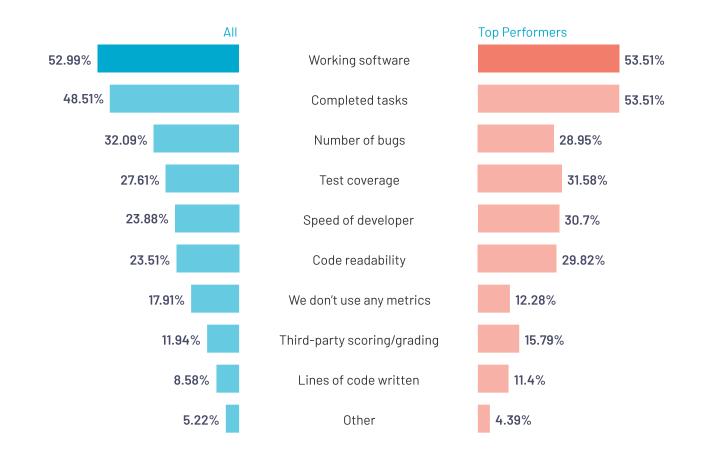
How effective is software development at your company?
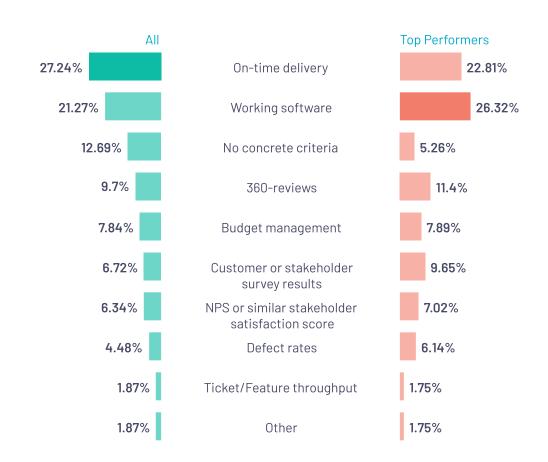
What metrics do you use to measure **developer** performance?

All

Top Performers

Metric	All	Top Performers
Working software	52.99%	53.51%
Completed tasks	48.51%	53.51%
Number of bugs	32.09%	28.95%
Test coverage	27.61%	31.58%
Speed of developer	23.88%	30.7%
Code readability	23.51%	29.82%
We don't use any metrics	17.91%	12.28%
Third-party scoring/grading	11.94%	15.79%
Lines of code written	8.58%	11.4%
Other	5.22%	4.39%
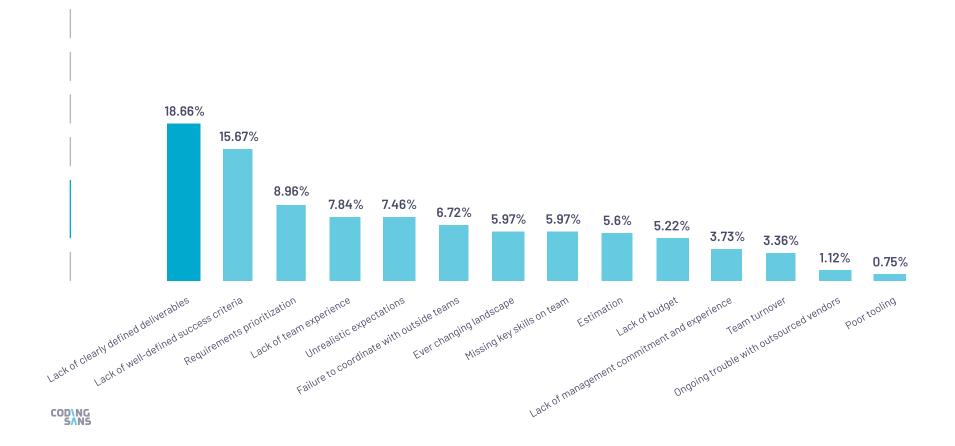
#1 measured criteria for success of **engineering managers**?

	All		Top Performers
27.24%		On-time delivery	22.81%
21.27%		Working software	26.32%
12.69%		No concrete criteria	5.26%
9.7%		360-reviews	11.4%
7.84%		Budget management	7.89%
6.72%		Customer or stakeholder survey results	9.65%
6.34%		NPS or similar stakeholder satisfaction score	7.02%
4.48%		Defect rates	6.14%
1.87%		Ticket/Feature throughput	1.75%
1.87%		Other	1.75%

05
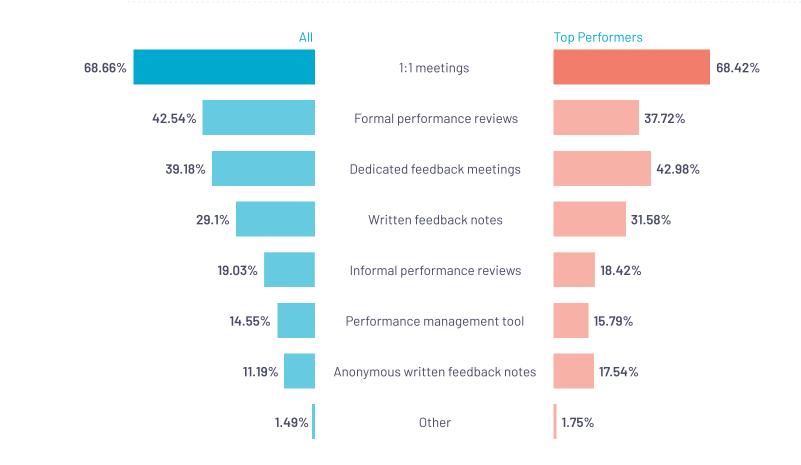
What is the #1 cause of delivery problems for your team?

18.66% — Lack of clearly defined deliverables
15.67% — Lack of well-defined success criteria
8.96% — Requirements prioritization
7.84% — Lack of team experience
7.46% — Unrealistic expectations
6.72% — Failure to coordinate with outside teams
5.97% — Ever changing landscape
5.97% — Missing key skills on team
5.6% — Estimation
5.22% — Lack of budget
3.73% — Lack of management commitment and experience
3.36% — Team turnover
1.12% — Ongoing trouble with outsourced vendors
0.75% — Poor tooling

What are your primary channels for giving feedback?

	All		Top Performers
1:1 meetings	68.66%		68.42%
Formal performance reviews	42.54%		37.72%
Dedicated feedback meetings	39.18%		42.98%
Written feedback notes	29.1%		31.58%
Informal performance reviews	19.03%		18.42%
Performance management tool	14.55%		15.79%
Anonymous written feedback notes	11.19%		17.54%
Other	1.49%		1.75%

CODING SANS

How often do you have 1:1 meetings with your team?

	All		Top Performers
Weekly	37.31%		35.96%
Biweekly	25.75%		30.7%
Monthly	13.06%		14.04%
More than once a week	11.94%		8.77%
Quarterly	7.09%		7.89%
Not using 1:1 meetings	4.1%		2.63%
Other	0.75%		0%

Interview with **Lara Hogan** / Leadership Coach

"The worst kind of feedback isn't clear on what you can do about it."

LEVEL→UP
engineering
PODCAST

Podcast with Lara

CHECK IT OUT ⟶

No one should get surprising feedback in a performance review. In a perfect world, you receive feedback from everyone you're working with between performance reviews, so you should know what they have to say by the time a review cycle comes around. This doesn't always happen in reality.

Emphasize the reason

People often give feedback without saying why it's important. That's a mistake.

Make it specific

Make everything included in a performance review specific. Give feedback based on specific events or actions in a specific environment. In a worst-case scenario, at least specify the outcome you're looking for.

Focus on behaviors. Saying, "You're so smart," is too broad. Clarify the specific behaviors you've seen them demonstrate before, or behaviors

you'd like to observe in that person going forward.

It can be specific and actionable, but you also need your team members to understand what you're trying to achieve, and why it matters to their colleagues or the company. Make it clear why a specific behavior is great or why it's a problem, so they have a clear goal by repeating or changing that behavior.

Make it fair

Fairness is one of our core needs at work. When we perceive something as unfair, the amygdala takes over and we want to avoid that. When you deliver feedback, make sure to make it as fair as possible to your report.

When you deliver feedback about behaviors that you haven't witnessed, you need to triple check whether it's valid; otherwise, it can't be fair. You also need to understand the context to make it actionable and to answer any questions your report may have to help them improve.
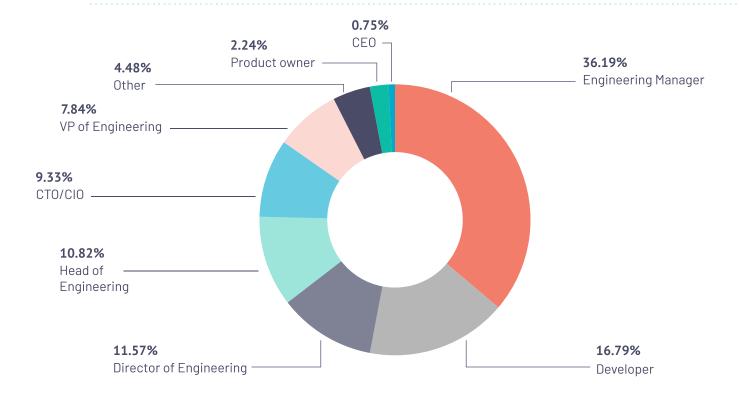
CODING SANS

About this Survey

The State of Engineering Management 2022 survey had 268 respondents, all of them are professionals from the software industry. This chapter provides more info about who they are, what kind of company they work for and where they're based.

Bylines and Methodologies

Coding Sans fielded an online survey in cooperation with two partners from March 2022 until the end of April 2022. The survey was available in English and consisted of 29 questions. The responses were sourced via email invitations, social media, and newsletter promotion. No personal information was collected within the survey. eGift cards were offered as incentives to increase survey participation and completion rate.
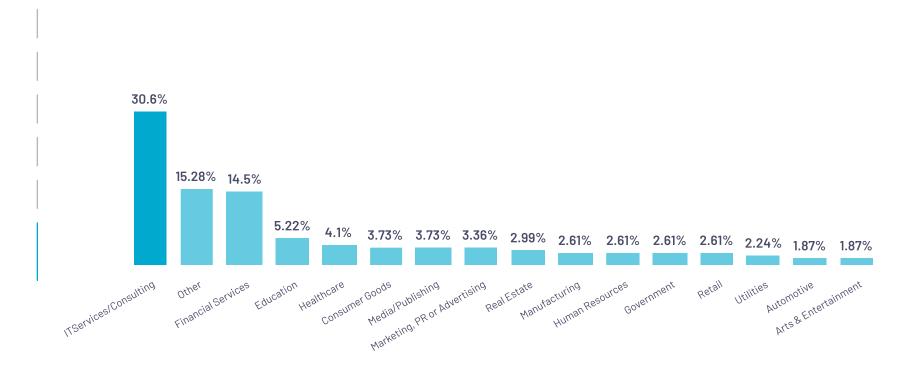
06

CODING SANS

What's your role in your company?

0.75%
CEO

2.24%
Product owner

4.48%
Other

7.84%
VP of Engineering

9.33%
CTO/CIO

10.82%
Head of
Engineering

11.57%
Director of Engineering

36.19%
Engineering Manager

16.79%
Developer

What industry are you in?

- ITServices/Consulting — 30.6%
- Other — 15.28%
- Financial Services — 14.5%
- Education — 5.22%
- Healthcare — 4.1%
- Consumer Goods — 3.73%
- Media/Publishing — 3.73%
- Marketing, PR or Advertising — 3.36%
- Real Estate — 2.99%
- Manufacturing — 2.61%
- Human Resources — 2.61%
- Government — 2.61%
- Retail — 2.61%
- Utilities — 2.24%
- Automotive — 1.87%
- Arts & Entertainment — 1.87%

Location and Company Size

Where is your company based?

North America **38.43%**

30.97% European Union

Eastern Europe **4.1%**

Oceania, Africa, Middle East **6.72%**

10.44% Central and South America

9.33% Asia

How many employees does your company have?

101–200

16%

0–20

12%

201–500

15%

30% 500+

13%

21–50

15%

51–100

Nice to meet you!

Coding Sans is a software development agency, building web applications from design to delivery with Node.js, React and Angular.

We're based in Hungary working for companies all around the world.

@ Say hi here:
info@codingsans.com

💬 Questions & feedback:
gabor.zold@codingsans.com

📍 Snail mail:
1051 Budapest, Arany János utca 10., Hungary

🏠 Website:
www.codingsans.com

Let's connect!

in LINKEDIN ⟶

🐦 TWITTER ⟶

f FACEBOOK ⟶

📷 INSTAGRAM ⟶

From design to delivery.

CODING SANS