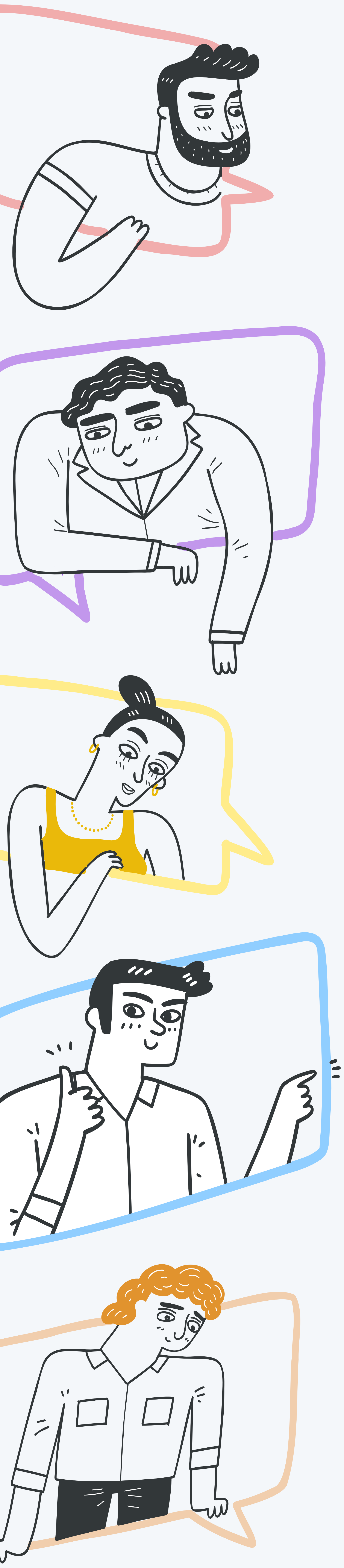
The Ultimate Guide to Running 1:1 Meetings in Software Development

Tips and insights on leading your developers
with empathy, context, and a sense of direction.

hay



1-on-1 meetings are one of the most important tools for a software engineer's growth, development, and wellbeing. Yet, tech leads and engineering managers often struggle with deciphering emotions, identifying needs, and interpreting responses.

Most importantly, leaders fail to align both their personal and the company's objectives with the satisfaction, performance, and productivity of their engineers.

We expect engineering managers to be great problem solvers when building technology. But how can we help them solve complex human problems during the software development cycle?

That's what this guide is all about.

What Is a 1:1 Meeting?

One-on-ones are regularly held meetings between an engineering manager and a software developer. Beyond updates surrounding PRs and commits, they aim to improve alignment, motivation, productivity, and personal growth.

The key here is for the engineering manager to offer personalized attention. A wide range of topics are discussed like performance, goals, challenges, upskilling, or any other issue that may affect the engineer's Developer Experience (DX).

It's vital to recognize that 1:1s aren't just ordinary conversations, "touchpoints," or mere formalities. The confusion surrounding 1:1s are exactly what makes them a challenging task for engineering leaders.

Seeing this problem persist in tech companies, we'd like to introduce our 1-on-1 meeting toolkit to foster constructive, ongoing dialogue between engineering managers and software developers.

The *Benefits* of Running 1-on-1 Meetings

Before we take a deep dive into the world of 1-on-1s, let's first deconstruct their meaning and essence.

Consider 1-on-1 meetings not merely as casual, albeit candid, conversations but as crucial links within the organization. By reevaluating your perspectives on 1:1s, you'll uncover benefits that may have previously gone unnoticed.

Benefits for Organizations:

We know that effective 1-on-1s improve rapport between engineering managers and developers. But there's solid evidence to show that they serve as a strategic conduit for productivity, morale, engagement, and organizational cohesion.

With the ability to unearth insights on an interpersonal level, managers can then cultivate a feeling of belonging and teamwork, and that can be a determining factor for company growth and success. Enhancing 1-on-1s can significantly reduce turnover risks, leading to a more agile, harmonized, and productive development environment.

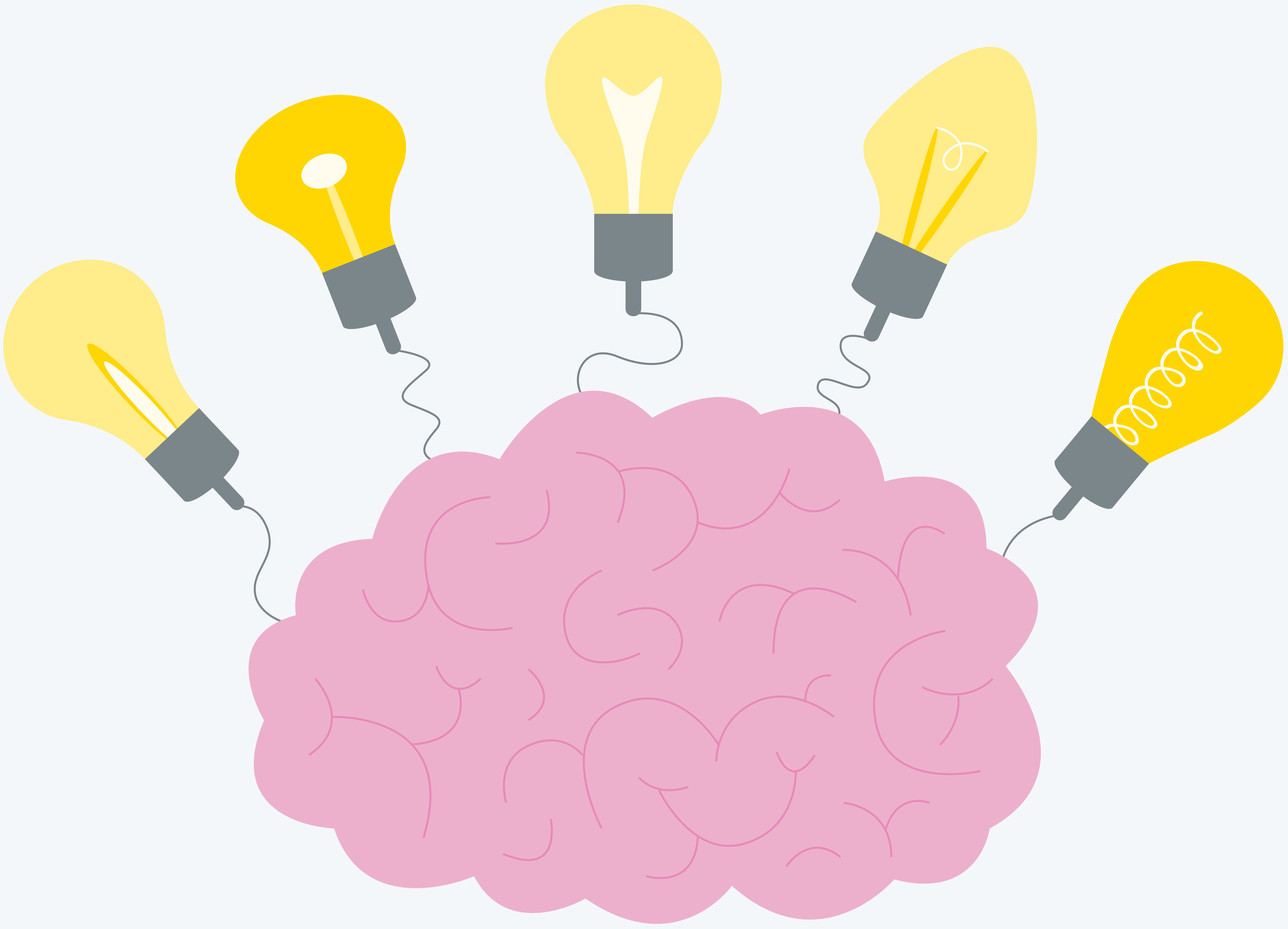
Benefits for Engineering Managers:

1-on-1s can help leaders make better decisions. These sessions allow managers to gain insights into individual challenges, aspirations, and feedback, thereby enabling them to tailor support and guidance effectively. Furthermore, regular 1-on-1s foster trust, enhance team cohesion, and proactively address concerns, ultimately driving both individual and team performance to higher levels.

Benefits for Software Developers:

1-on-1 meetings serve as a crucial avenue for software developers. They provide developers with the opportunity to "speak up," allowing for personalized feedback and career development discussions.

A by product of this is increased motivation, job satisfaction, and, consequently, heightened productivity. An effective, forward-focused 1:1 meeting can help developers thrive both technically and personally within the organization.



Pitfalls for Engineering Managers to *Avoid*

During 1:1 meetings, engineering managers should be aware of several potential pitfalls to ensure that sessions are productive and meaningful.

Here are some to consider:

- **Lack of Preparation:** Failing to prepare can lead to a directionless conversation. It's crucial to have an agenda or key points to address.
- **Dominating the Conversation:** A 1:1 meeting is a two-way dialogue. Engineering managers who do all the talking prevent their developers from voicing their concerns or opinions.
- **Avoiding Difficult Conversations:** Skipping over challenging topics or feedback doesn't help anyone. It's essential to address issues head-on, albeit sensitively.
- **Being Distracted:** Not giving full attention, constantly checking phones or emails, can make your developer feel undervalued.
- **Inconsistency:** Rescheduling or frequently canceling 1:1s sends the message that they aren't important, and suggests that you value your developers even less.
- **Being Defensive:** If a developer provides feedback, getting defensive can shut down open communication.
- **Not Following Up:** If engineering managers don't address or follow up on concerns raised in previous meetings, it can seem like they aren't listening or don't care.
- **Relying Solely on Memory:** Not taking notes during or after the session might lead to forgetting crucial details or follow-up points.

- **Making it all about Work:** While the primary focus is professional growth and tasks, occasionally checking in on a developer's wellbeing or work-life balance can be beneficial.
- **Ignoring the Positive:** Only focusing on areas of improvement without recognizing and appreciating the achievements can demotivate developers.

> "1-on-1s in software development are dedicated spaces for open and honest dialogue that pave the way for trust-building and mutual growth."

The Manager's *Checklist* For Effective 1-on-1s

Here's a detailed roadmap to assist engineering managers in organizing and executing successful 1-on-1 sessions. Please pass it along to anyone in your team who could benefit. It's not about dictating how one should work, but offering enhanced resources to empower them in their position.

Setting the Context

- □ **Importance:** Reinforce the purpose and value of 1:1s to both parties. Stress that it's a safe space for honest discussions.
- □ **Modus Operandi:** Explain the structure and desired outcome of the meeting. Establish what productivity is and how you'd like to track progress and wellbeing. Include time for updates, feedback, career conversations, etc.
- □ **Ownership:** Empower your developer and emphasize the fact that it's their meeting.
- □ **Collaboration:** Ask how collaborating could maximize the purpose of the meeting.
- □ **Frequency:** Determine the length and regularity. We recommend meetings that are weekly to monthly that last 30-60 minutes. Consistency is key.
- □ **Flexibility:** Set a foundational structure and follow it, but remain flexible for any topics your developer might want to bring up unexpectedly.
- □ **Venue Choice:** Decide on the logistics – whether it's via Zoom, in a meeting room, or even stepping out of the office.

Prepare Ahead

□ **Shared Agenda:** Encourage your engineer to bring up topics they'd like to discuss. A day or so in advance, co-create an agenda that incorporates both your concerns and theirs.

□ **Avoid Last-minute Changes:** Unless there's an emergency, try not to reschedule or cancel the 1:1. If you do, you're basically telling your developer that the meeting, and by extension he or her, is not a priority.

Foster a Conducive Ambience

□ **Punctuality:** Start and end on time. This respects everyone's schedules and maintains the session's structure.

□ **Active Presence:** Ensure you're mentally present. Avoid distractions like phones, emails, or other tasks.

□ **Environment:** Opt for a quiet space where both parties can speak freely without distractions or interruptions.

Meeting Coverage

□ **Deep Dive:** Allow for conversations that go beneath the surface. This can bring out underlying issues or aspirations that might not be evident in day-to-day interactions.

□ **Mutual Feedback:** Ensure that the conversation is two-sided. While you provide feedback to the engineer, also be open to receiving feedback about yourself, the team, or the company.

□ **Career Growth:** Dedicate some time to discuss the engineer's career aspirations and potential growth paths.

Conclusion

□ **Summary:** Recap the main points discussed to ensure both parties are aligned and no topic was overlooked.

□ **Action Items:** If there are any actionable steps, tasks, or follow-ups, make sure they're clearly identified.

□ **Set the Tone for the Next Meeting:** Highlight what you'd like to focus on or achieve by the next 1:1 to give direction and purpose.

Become the *Engineering Leader* You've Always Aspired to Be

Conducting effective one-on-one meetings with your software developers requires commitment and effort. However, when approached with empathy, clarity, and purpose, these meetings provide tech leads and engineering managers a valuable chance to nurture relationships and pivot strategies as needed.

To learn more about our Effective Engineering Manager Course, tap the button below.

Learn More

1-to-1 Meeting *Template* for Software Engineering Managers

Date

Enginering name

Current project

Agenda items for engineer:

-
-
-

Agenda items for manager:

-
-
-

Notes:

Action items:

☐

☐

☐

How to use this free 1:1 meeting template with your engineers

- Download this template by clicking on 'GET THE TEMPLATE'.
- Store it in a place you'll remember, so you can reuse it weekly or monthly.
- Replace the logo and customize this template i.e., change font and style.
- Make sure the most recent week is at the top of the template.
- Don't forget to delete the instructions.
- Have fun in your 1:1 meetings! If you'd like to know more information about how to run 1:1 meetings, progress checks, or development conversations, read our blog here.

GET THE TEMPLATE

20 Essential Questions for Software Developers in a 1:1 Meeting

It's a good idea to have a bank of questions to ask in your 1-on-1 meetings. Having a mixture of well-timed and appropriately-applied questions will build rapport and stimulate deeper conversation to provide clarity for learning.

Here's a curated list of 20 crucial questions to elevate your one-on-one meetings. These questions span across four foundational pillars of software development: Work Output, Wellbeing, Team Feedback, and Growth.

Work Output

Sprint Analysis: "How was the last sprint? What could we improve?" Provides insight into team dynamics, and potential blockers, and sets a stage for continuous improvement.

Task Management: "Do you find your tasks manageable or overwhelming?" Helps identify workload issues and promotes discussions on task prioritization.

Purpose and Alignment: "Do you feel your work aligns with the project's larger goals?" Ensures developers see the bigger picture and are engaged with the project's mission.

Productivity Boosters: "What helps you achieve flow during coding?" Recognize the tools and conditions that enhance developer efficiency.

Work Output Assessment: "How do you measure your productivity?" A holistic approach to gauging productivity beyond traditional metrics.

Wellbeing

Job Satisfaction: "How content are you with your role and responsibilities?" Directly correlates with motivation, performance, and retention rates.

Remote Work Dynamics: "How's your remote work experience? Any challenges?" Addressing the unique challenges of remote work fosters a supportive work environment.

Value Contribution: "Do you feel you're making a significant impact with your work?" Reinforces a sense of purpose and motivation.

Work-Life Balance: "How do you strike a balance between work and personal time?" Promotes employee wellbeing and identifies potential burnout indicators..

After-Work Rituals: "How do you decompress after a busy workday?" Helps build rapport and understand personal preferences.

Growth

Short-Term Aspirations: "What are your professional goals for the next 6-12 months?" Aligns team growth with company objectives.

Support Mechanisms: "How can I assist you in achieving these goals?" Positions leadership as enablers rather than just engineering managers.

Long-Term Vision: "Where do you envision yourself in 5 or 10 years?" Navigates career trajectories and aligns with company growth plans.

Skill Assessment: "What are your key skills and how would you rank yourself?" Identifies areas for skill enhancement and training opportunities.

Mentorship: "Would you like mentorship from anyone in the company?" Fosters a culture of learning and collaboration.

Team Feedback

Team Dynamics: "How's your experience with [team member name] or the current dev team you're in?" Spotlights areas for team-building and conflict resolution.

Expectations: "What do you expect from your teammates?" Aligns team values and work ethics.

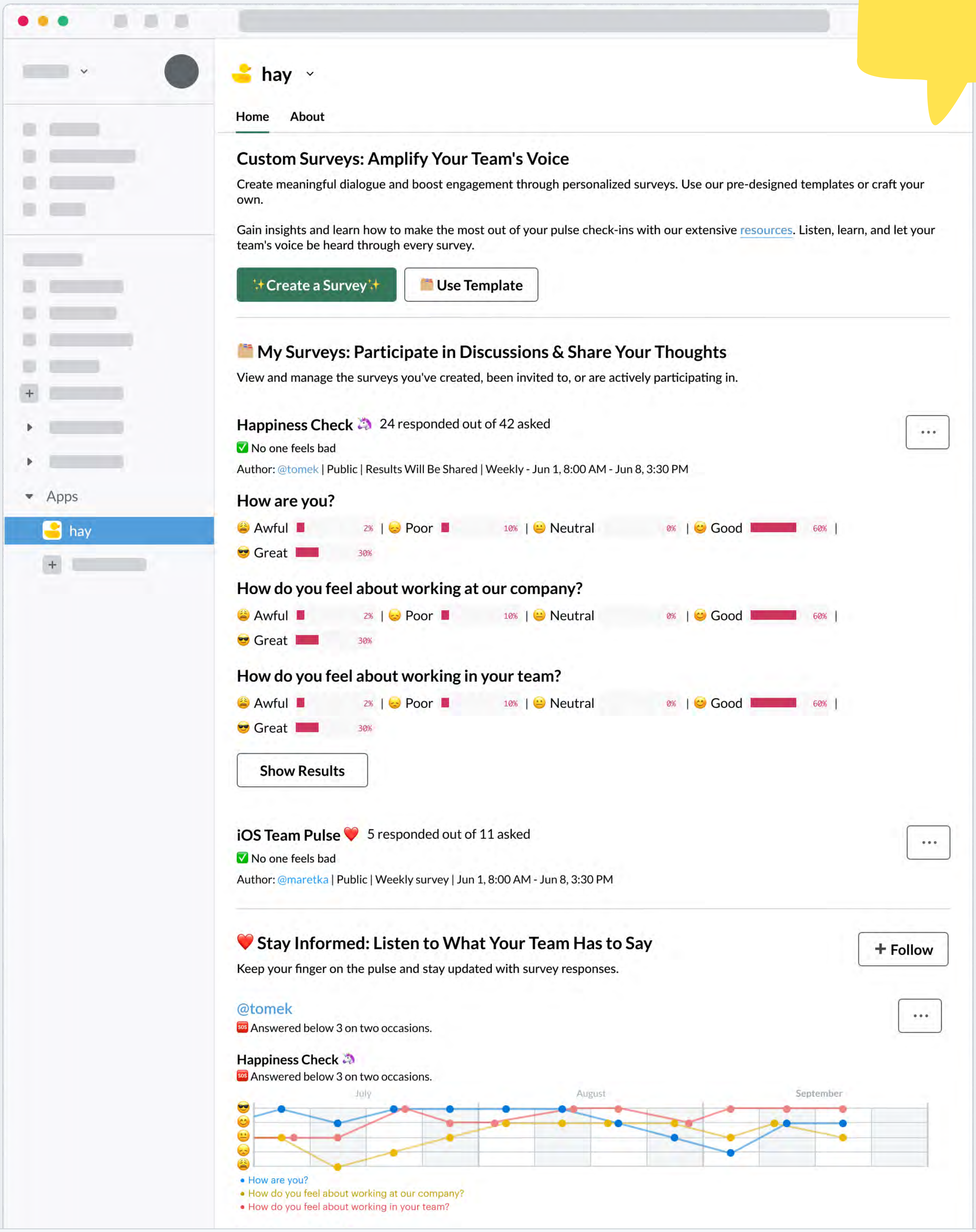
Feedback Loop: "How can I better support you?" Strengthens the manager-developer relationship and encourages open communication.

Feedback Effectiveness: "Is the feedback you receive helping you grow?" Ensures feedback mechanisms are constructive and effective.

One-on-One Effectiveness: "Any suggestions on how I conduct these meetings?" Encourages continuous improvement in meeting dynamics.

Looking for a tool that will help you check in on your developer's wellbeing? Install HAY for Slack!

[Try HAY Now!]



hay

Home About

Custom Surveys: Amplify Your Team's Voice

Create meaningful dialogue and boost engagement through personalized surveys. Use our pre-designed templates or craft your own.

Gain insights and learn how to make the most out of your pulse check-ins with our extensive resources. Listen, learn, and let your team's voice be heard through every survey.

✨Create a Survey✨ | Use Template

My Surveys: Participate in Discussions & Share Your Thoughts

View and manage the surveys you've created, been invited to, or are actively participating in.

Happiness Check 24 responded out of 42 asked

✅ No one feels bad

Author: @tomek | Public | Results Will Be Shared | Weekly - Jun 1, 8:00 AM - Jun 8, 3:30 PM

How are you?

Awful 2% | Poor 10% | Neutral 0% | Good 60% |
Great 30%

How do you feel about working at our company?

Awful 2% | Poor 10% | Neutral 0% | Good 60% |
Great 30%

How do you feel about working in your team?

Awful 2% | Poor 10% | Neutral 0% | Good 60% |
Great 30%

Show Results

iOS Team Pulse ❤️ 5 responded out of 11 asked

✅ No one feels bad

Author: @maretka | Public | Weekly survey | Jun 1, 8:00 AM - Jun 8, 3:30 PM

❤️ Stay Informed: Listen to What Your Team Has to Say

Keep your finger on the pulse and stay updated with survey responses.

+ Follow

@tomek

🆘 Answered below 3 on two occasions.

Happiness Check

🆘 Answered below 3 on two occasions.

Apps
hay
July
August
September
• How are you?
• How do you feel about working at our company?
• How do you feel about working in your team?