

Are You Being Optimized Away?

The Senior Engineer Risk Scorecard — 10 questions to find out if your role is strategic or expendable.

In 2026, a mid-level engineer with AI assistance costs half your salary and ships nearly as fast. Companies aren't anti-senior — they're anti-expensive-without-leverage. This scorecard helps you assess where you actually stand: **strategic asset** or **line item waiting to be cut**.

HOW TO USE THIS SCORECARD

Answer each question honestly. Circle or note the letter (A, B, or C) that best describes your current situation. Each answer carries a point value. Total your score at the end to see your risk level.

A answers = 0 points	High risk — this is where AI-augmented mid-levels replace you
B answers = 1 point	Moderate — you have some leverage but it's not secure
C answers = 2 points	Strong — this is structurally necessary positioning

Answer honestly. Your career depends on it.

Q01 If you were removed tomorrow, what would break?

A (0 pts)	Delivery speed would slow down temporarily, but the team would adapt within a sprint or two.
B (1 pt)	A few important projects would stall, and some tribal knowledge would be lost.
C (2 pts)	Core system integrity, security posture, or architectural decisions that only I fully understand.

Q02 How do you primarily spend your working hours?

A (0 pts)	Writing code, fixing bugs, and shipping features — similar work to mid-level engineers but higher quality.
B (1 pt)	A mix of coding and some design work, code reviews, and mentoring.
C (2 pts)	Designing systems, making architectural decisions, influencing roadmaps, and preventing costly mistakes.

Q03 Can your manager articulate your unique value to their boss?

A (0 pts)	Probably not beyond 'they're very experienced' or 'they write great code.'
B (1 pt)	They could name a few important contributions, but nothing tied to specific business outcomes.
C (2 pts)	Yes — they can point to specific decisions I made that saved money, prevented incidents, or drove revenue.

Q04 How has AI tooling changed your team's output in the last year?

A (0 pts)	Mid-level teammates are now shipping features that previously required my level of experience.
B (1 pt)	Everyone is faster, including me, but the gap between my output and others' hasn't changed much.
C (2 pts)	AI has amplified my strategic value — I use it to design better systems, evaluate more options, and make faster decisions.

Q05 When was the last time you directly influenced a business decision?

A (0 pts)	I mostly influence technical decisions within my team's scope.
B (1 pt)	I've pushed back on timelines or scope in the last quarter.
C (2 pts)	I regularly advise on build vs. buy, tech stack choices, or architecture decisions that impact revenue or cost.

THE 10 QUESTIONS (CONTINUED)

Q06 **How replaceable is your knowledge?**

A (0 pts)	Most of what I know is documented or could be learned from the codebase and existing docs.
B (1 pt)	I carry significant context, but a good engineer with 3-6 months of ramp-up could get there.
C (2 pts)	I hold deep cross-system knowledge, institutional memory, or domain expertise that would take years to rebuild.

Q07 **What's your relationship with leadership above your direct manager?**

A (0 pts)	I don't interact with leadership — my manager represents me in those conversations.
B (1 pt)	I've presented in a few meetings but don't have regular visibility.
C (2 pts)	I have a direct relationship with senior leadership; they consult me on technical strategy.

Q08 **How do you respond when asked to do something you disagree with?**

A (0 pts)	I push back strongly and explain why it's a bad idea. Sometimes this creates tension.
B (1 pt)	I voice concerns but generally commit to the team's direction.
C (2 pts)	I frame my concerns in business terms — cost of failure, risk mitigation, time-to-revenue — and propose alternatives.

Q09 If your salary were cut by 30%, would your company still be getting a bargain?

A (0 pts)	Honestly, they could get 80% of my output from a mid-level engineer using AI tools.
B (1 pt)	It would be unfair, but the market might support it given current trends.
C (2 pts)	Absolutely — I prevent costly mistakes, own critical systems, and my judgment saves multiples of my salary.

Q10 What are you building outside your day job?

A (0 pts)	Nothing really — my skills are tied to my current role and employer.
B (1 pt)	I contribute to open source or do some side projects occasionally.
C (2 pts)	I'm building a personal brand, publishing, teaching, or creating assets that make me visible beyond my company.

Add up your points and face the math.

MY TOTAL SCORE: _____ / 20

CRITICAL RISK

0 - 6 POINTS

You are currently positioned as an expensive individual contributor. The spreadsheet math works against you. A mid-level engineer with AI tools can approximate most of your visible output at half your cost. Without a rapid repositioning, you are a restructuring cycle away from being optimized out. You need to act now.

MODERATE RISK

7 - 12 POINTS

You have some strategic value but it's not consistently visible. Your manager might fight for you in a budget cut, but couldn't make a bulletproof case to the CFO. You're in the danger zone the article describes: too expensive to be a default hire, not positioned as essential leadership. You have time, but not much.

POSITIONED WELL

13 - 17 POINTS

You're doing many things right. Your value goes beyond code output and you have some visibility with leadership. The risk is complacency — the market is shifting fast and today's strong position erodes if you stop actively maintaining it. Focus on making your strategic value even more measurable and visible.

STRUCTURALLY NECESSARY

18 - 20 POINTS

You are the senior engineer the article says will remain 'untouchable.' You own systems, influence roadmaps, tie technical decisions to revenue, and reduce risk in measurable ways. Companies don't just want you — they need you. Your challenge now is leverage: are you being compensated and positioned at the level your impact deserves?

Regardless of your score, here are the 5 moves.

01 — STOP SELLING CODE, START SELLING JUDGMENT

AI commoditized syntax and implementation. What it hasn't commoditized is the ability to decide what to build, how to architect it, and what not to build. Reframe every conversation from 'I wrote this' to 'I decided this, and here's why it saved us X.'

02 — MAKE YOUR VALUE VISIBLE AND MEASURABLE

Invisible value gets optimized away. Start documenting the incidents you prevented, the architectural decisions that saved refactoring time, the technical debt you retired. Create a personal 'value ledger' and reference it in every performance review.

03 — OWN A SYSTEM, NOT A FEATURE

Feature developers are replaceable. System owners are strategic. Position yourself as the person who owns the integrity of a critical system — the one whose removal would create measurable business risk, not just slower delivery.

04 — BUILD RELATIONSHIPS ABOVE YOUR MANAGER

If the only person who can articulate your value is your direct manager, you're one reorg away from invisibility. Get in front of senior leadership. Present at architecture reviews. Make sure the people who approve budgets know your name and your impact.

05 — CREATE LEVERAGE OUTSIDE YOUR EMPLOYER

Publishing, teaching, speaking, and building in public creates positioning that no single company can take away. It also makes you more valuable to your current employer — engineers with external credibility are harder to let go and easier to promote.

Ready to pivot?

This scorecard shows you where you stand. The Dev Pivot Masterclass and 1:1 Strategy Sessions show you exactly how to move.

thedevpivot.com

Book a strategy session or join the masterclass.
Built by Luis Soares — 23 years, 6 industry shifts, still here.