

CTO Readiness - The Agentic SDLC

The CTO Readiness Checklist

A 10-point diagnostic for scaling agentic AI across the SDLC

For CTOs, VPs of Engineering and Chief Architects in banking, insurance and financial services.

Evaluate Before You Scale

Ten questions to test whether your engineering organisation is ready to scale agentic AI safely, or whether context drift and unchecked token spend are quietly compounding. Tick each statement that is true of your organisation today, then read your score below.

30–35%

Potential SDLC productivity gain from AI (Deloitte, 2026 outlook).

Under 10%

Agentic AI programmes reaching meaningful scale today (McKinsey).

33%

Developer trust in AI-generated code accuracy in 2025, down from 43% (Stack Overflow).

The productivity case is real. Few organisations have built the architecture, governance and talent model to capture it safely. This checklist is where that gap

01 | Context and codebase architecture

☐ 01. Context isolation and guardrails

Are your codebases structured with clear modular boundaries that stop agents generating code across domain or security boundaries?

☐ 02. Test coverage for AI output

Is your test suite robust enough to catch non-deterministic AI regressions before they reach senior human review?

☐ 03. Technical-debt telemetry

Do you track whether AI-assisted PR velocity is being offset by rising downstream bug-fix time?

02 | Token economics and regulatory governance

DORA and the EU AI Act make this section a board-level concern, not a platform-team one: cost, control and auditability now travel together.

☐ 04. Model routing and cost control

Are developers guided to route tasks between lighter and heavier models to prevent runaway token spend?

☐ 05. Context-window discipline

Have platform leads set prompt and context practices that stop redundant codebase scanning inflating API spend?

☐ 06. Auditable AI change trail

Can you produce a regulator-ready audit trail for AI-authored changes, with data-loss-prevention enforced across every AI IDE extension and API endpoint?

☐ **07. The context-architect transition**

Have senior engineers moved from syntax-level inspection to system design and supervisory review? The context architect designs intent and governs what agents build, rather than writing most lines by hand.

☐ **08. Persona-specific AI pathways**

Do your backend, SRE, quality and data teams have tailored AI workflows rather than one generic assistant subscription?

☐ **09. Early-career guardrails**

Are junior engineers given structured environments that build architectural fluency rather than over-reliance on AI output?

☐ **10. End-to-end SDLC coverage**

Is AI applied across the whole lifecycle, from specification and testing to release and operations, not raw coding alone?

Count the boxes you ticked.

9-10

Scaling with control. You have the architecture, governance and talent model to scale agentic AI safely. The work now is sustaining it as models and regulation change.

7-8

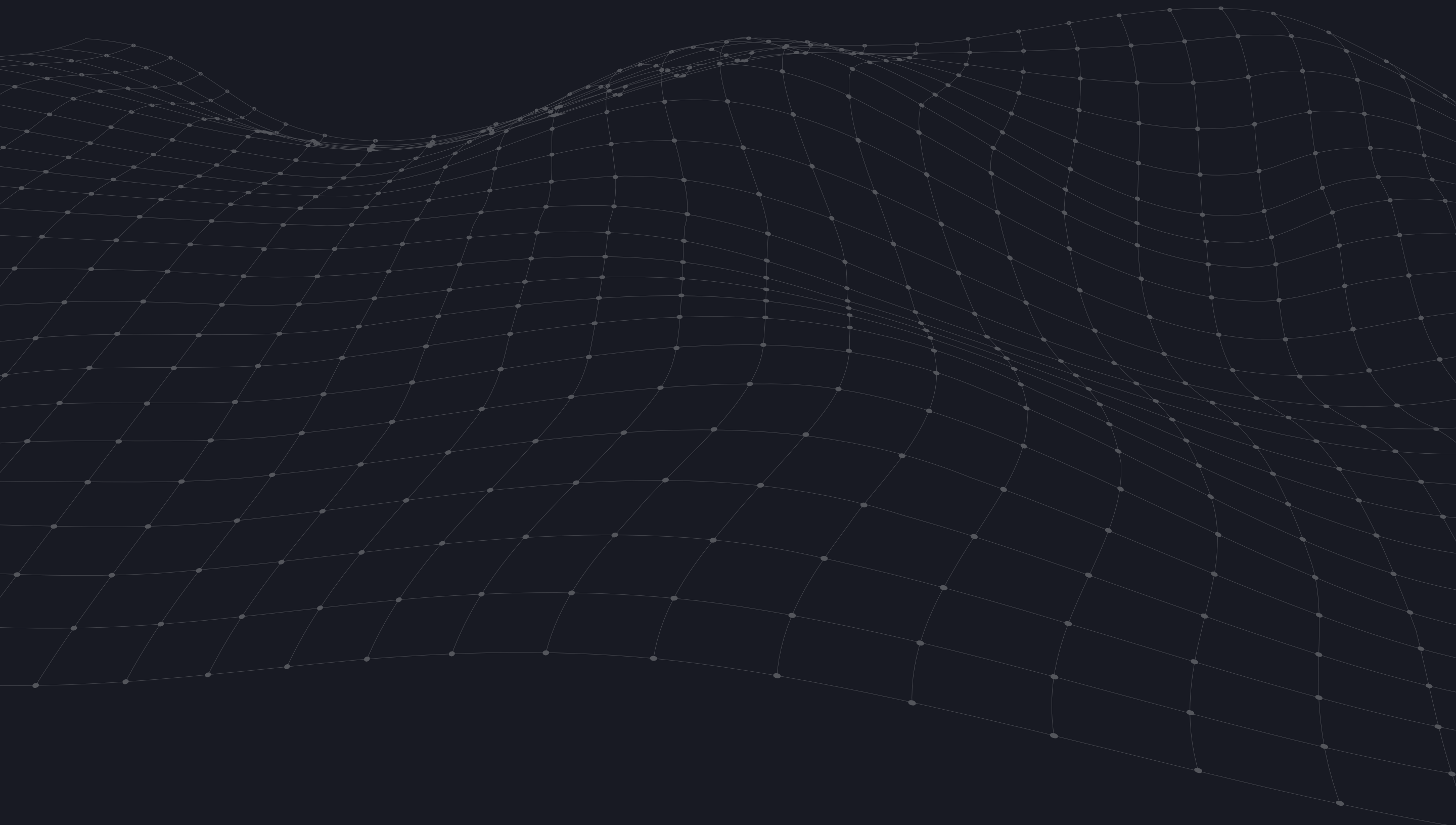
Ready, with gaps. The foundations are largely in place. Close the remaining gaps before you widen rollout.

4-6

Exposed. You are scaling faster than your guardrails. This is where context drift, token spend and review bottlenecks start to compound.

0-3

Pre-readiness. Build the foundations before scaling. Moving now risks the AI productivity paradox: faster code, slower delivery.



The Practitioner Verdict

The AI productivity paradox, writing code faster while feature delivery stalls behind review bottlenecks and rising token costs, is already measurable: developer trust in AI-generated code accuracy fell from 43% to 33% in a single year (Stack Overflow, 2025). This checklist is designed to get ahead of it.

Companion piece. This checklist accompanies our briefing, The State of the Agentic SDLC, which sets out the trends behind these ten questions.

Talk it through with a practitioner

Book a 45-minute stack-tailored AI capability audit with a senior Neueda practitioner: no generic frameworks, just a conversation grounded in your architecture, regulatory environment and team structure.



Sources: Deloitte, 2026 Global Software Industry Outlook; McKinsey, Reimagining tech infrastructure for agentic AI, 2026; Stack Overflow, 2025 Developer Survey. Figures are presented as sector commentary rather than independently verified statistics.