Who needs to review code?

What happened in the room | Participant report

> The room was confident in AI review, while human attention was shifting toward intent, context and consequences.

AI review is becoming routine

52% of respondents described mostly AI review with little human review, and 26% used AI to assist mostly manual review. 83% agreed that they were very satisfied with AI review capabilities. For this group, the practical question was how to work well with AI already in the review process.

The work of understanding remains

Time spent understanding or verifying changes was the most frequently selected individual problem, alongside concerns about complexity and pressure to ship. 61% of respondents said their review focus had shifted toward intent, design and system impact. AI support appears to be changing where people spend attention, while the need to understand a change remains.

Human value needs a clear focus

80% of respondents saw significant value in human review after AI review. In a separate poll, 41% were uncertain what their own review could add. The open responses offered concrete directions: architecture, business context, assumptions, logic, security and side effects. These concerns give human review a more specific purpose.

Oversight depends on consequences

For the initial authorization ticket, 64% chose an agent, developer and peer. After the external medical-records context was revealed, 100% of respondents chose a peer or team alongside the agent and developer. The results highlight why a small diff and passing checks are only part of an oversight decision.

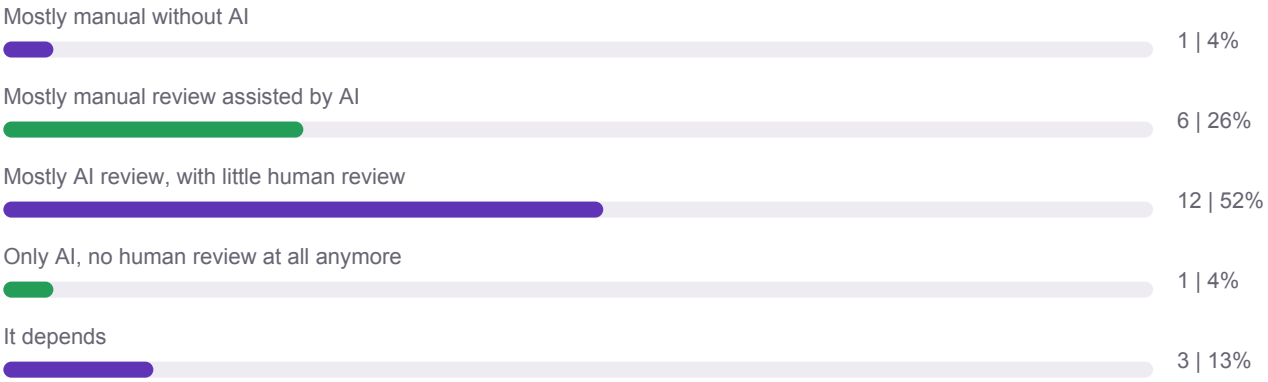
What to take forward

The results point toward focused, proportionate oversight: use AI to support the investigation, make the human review questions explicit, and involve the people who understand the affected risks. The workshop's SCOPE approach puts this into practice by agreeing risk and review focus early, adapting oversight as context emerges, and recording what the review established.

AI review was already established

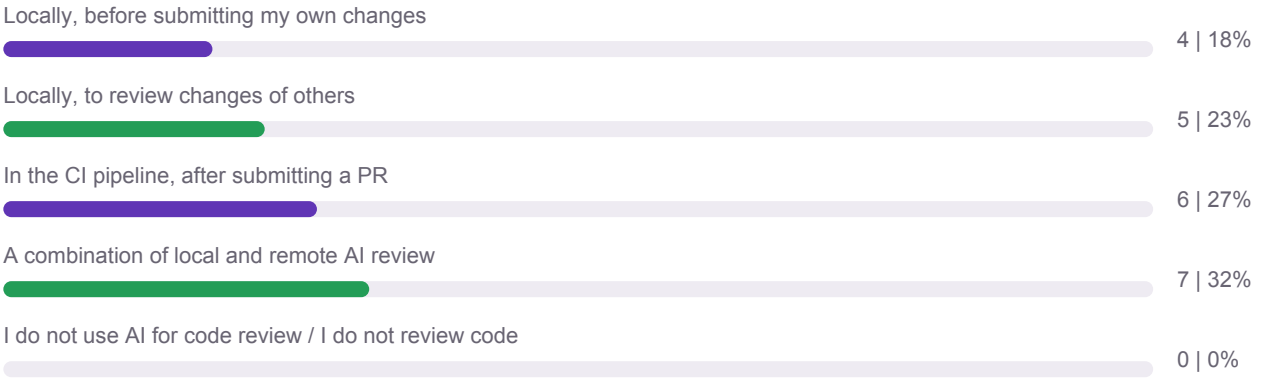
How do you review code today?

Poll 1 · 23 responses



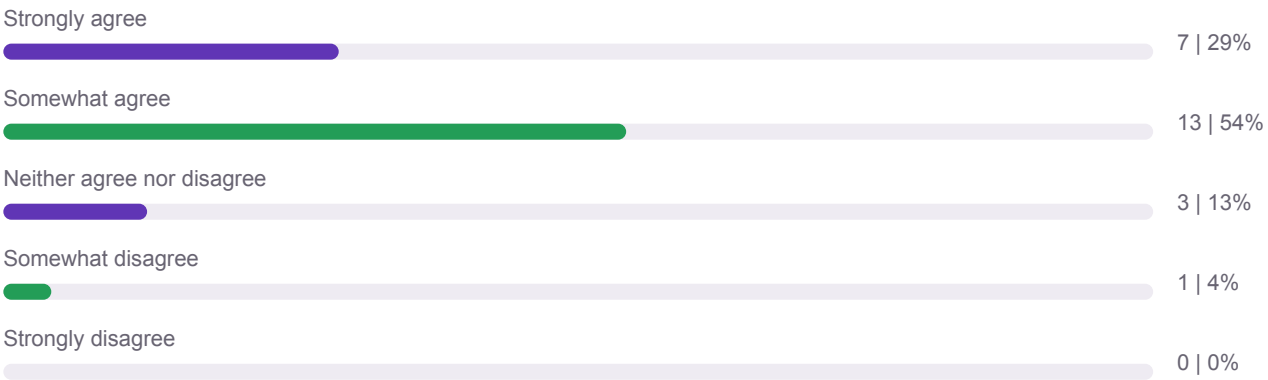
Where do you primarily use AI for code review?

Poll 2 · 22 responses



“I am very satisfied with the AI review capabilities.”

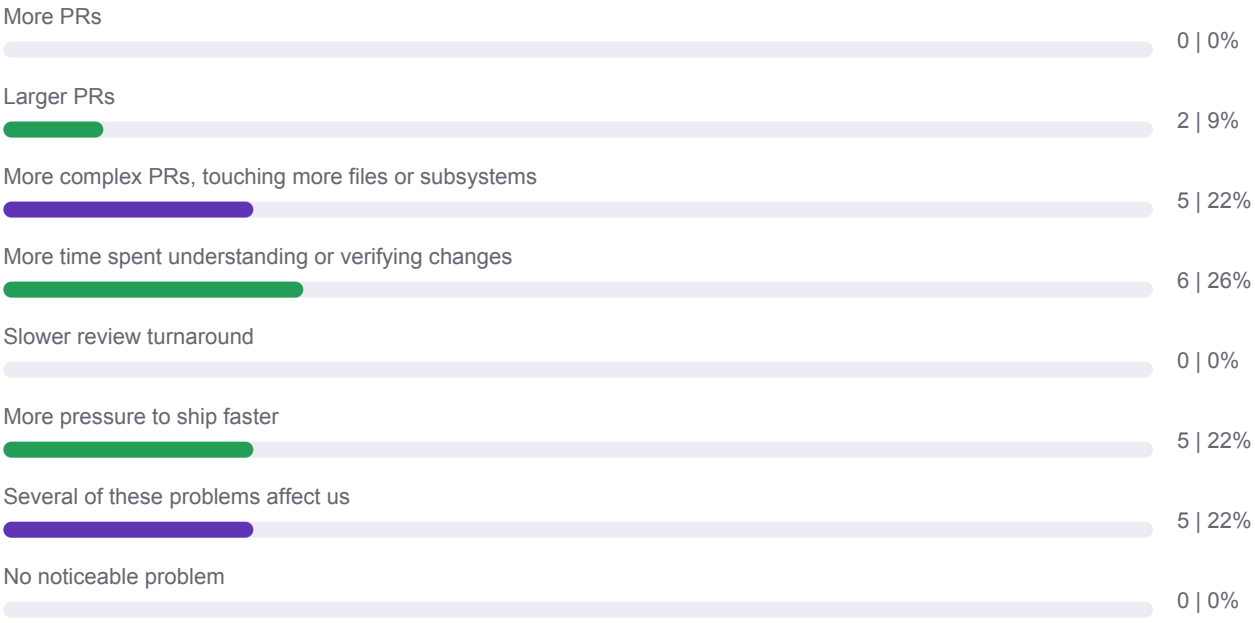
Poll 3 · 24 responses



Understanding was a pressure point

What is the most noticeable problem since AI coding tools became common in your team?

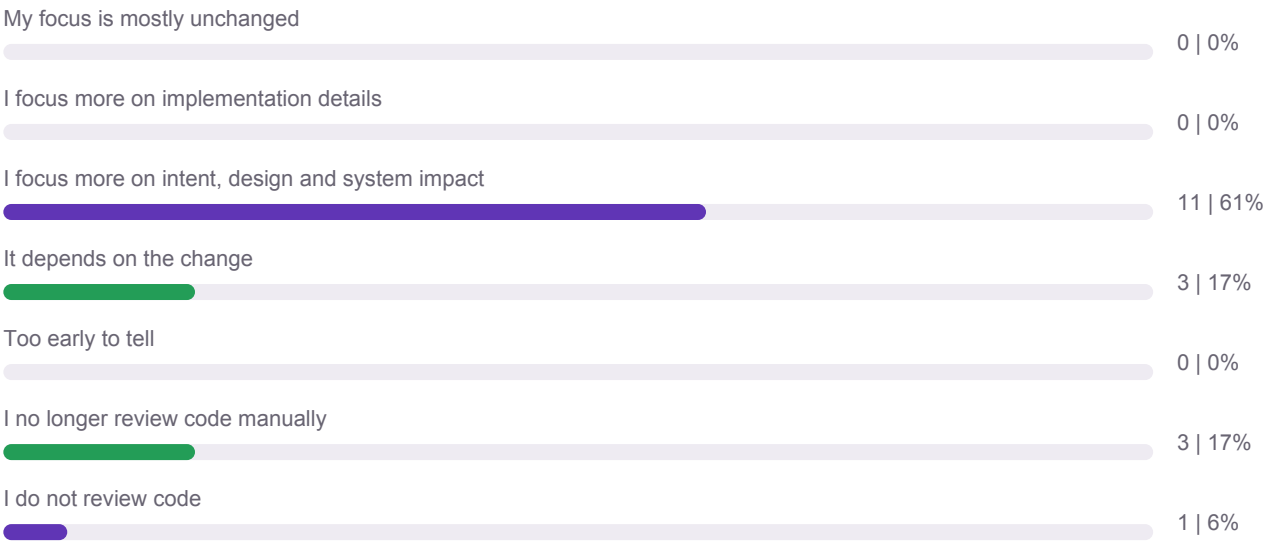
Poll 4 · 23 responses



The largest single category was time spent understanding or verifying changes (26%). Complexity, pressure to ship and the “several problems” option each accounted for 22%. Together, these answers put the effort of making sense of changes at the center of the discussion.

How has AI changed the focus of your own code reviews?

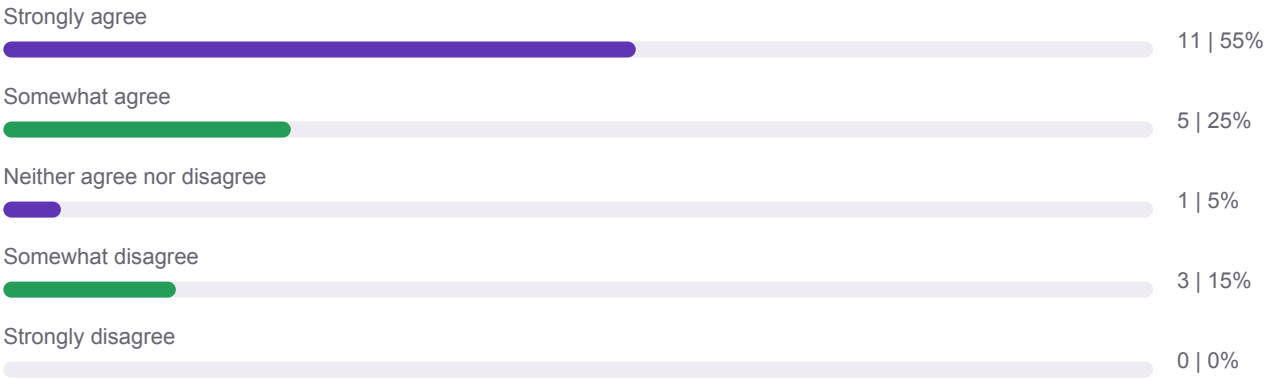
Poll 5 · 18 responses



Value was clear; the role was less settled

“After AI has reviewed the code, a human review still adds significant value.”

Poll 6 · 20 responses



“After AI has reviewed the code, I’m uncertain what my own review can add.”

Poll 7 · 22 responses



80% saw significant value in human review. Separately, 41% were uncertain what their own review could add.

The room valued human review, yet a substantial group wanted a clearer sense of their contribution. A useful response is to define the investigation: which assumptions need challenging, what context needs checking, and what evidence would make the change convincing?

Intent, structure and consequences

Poll 8 · When AI also reviews the code, what do you focus on in your own review?



The room’s responses, grouped by theme

Intent and fit. “Business Context”, “Specs are meet”, “Assumptions” and “suitability” point toward whether a change solves the intended problem.

Structure and maintainability. “Architecture”, “Design”, “Overall modules structure”, “complexity”, “Sprawl” and documentation describe how the change fits the system and how people will maintain it.

Behavior and consequences. “Logic”, “side-effects”, “Security”, “Risk”, “accuracy” and “Hallucinations” keep correctness and consequences in view.

Understanding and explanation. “Understandability”, “Workflow”, “Process” and “chain of thought process” show interest in how a solution works and how its reasoning can be examined.

Context and boundaries stayed central

Poll 9 · Where do you think AI still needs help with?



The room named several kinds of help

Supplying and checking context: “Context”, “Business context”, “Undiscovered context” and “Amnesia” suggest that relevant knowledge may be absent or lost.

Keeping work bounded: “Keep it scoped”, “boundaries”, “Know when to stop”, “saying no” and “Prioritization” concern deciding what work should happen and when it is enough.

Challenging the result: “Validation”, “Side effects”, “Risk”, “Review of review points”, “Explainability” and “admitting failure” concern testing claims and recognizing limitations.

Fitting the codebase: “Reusability of code”, “Adhere to local standards”, “succinct code and docs” and “Unconsidered approach” bring design alternatives and local practice into view.

A review plan can turn these concerns into explicit questions: what context matters, which boundaries must hold, and what evidence would let us stop?

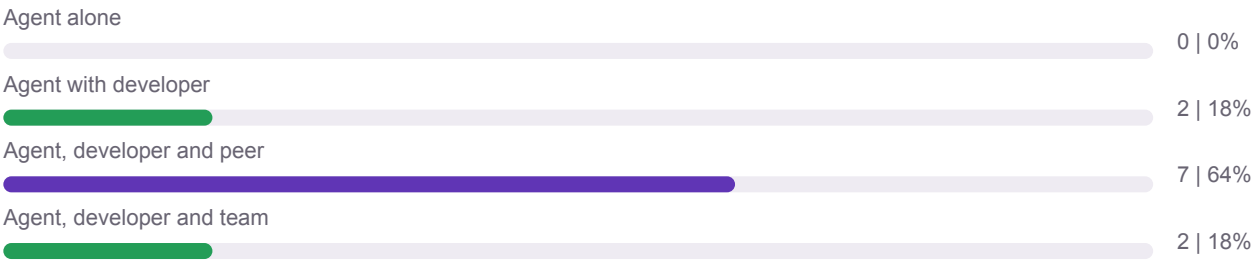
A small authorization change, in stages

Polls 10-12 · AUTH-184: restore report access for legacy SSO users

The ticket proposed using existing membership records when the legacy SSO provider supplies no role claim, while preserving permission checks. The intended boundary was precise: restore entitled access without admitting unentitled users or crossing organization boundaries.

First decision: how much oversight does AUTH-184 need?

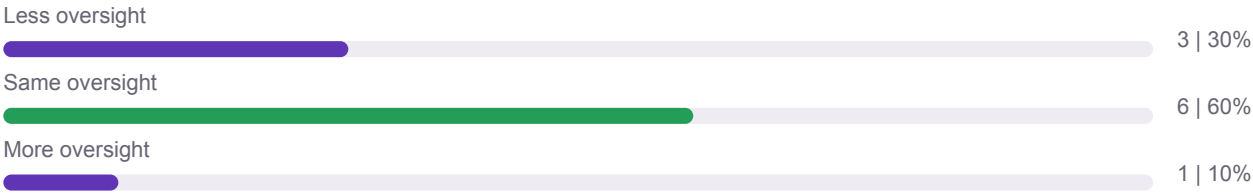
Poll 10 · 11 responses



Next, the PR summary added a senior author, 14 changed lines across two files, passing CI and 96% middleware line coverage.

After seeing the PR: less, the same or more oversight?

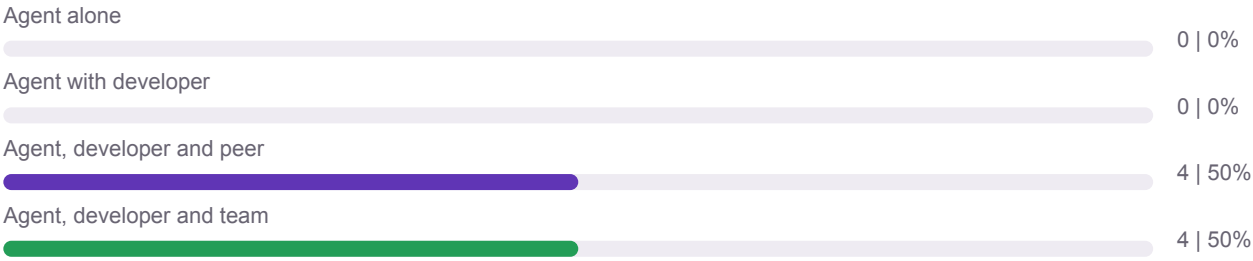
Poll 11 · 10 responses



The later reveal added past sign-in/access problems and an external customer system containing confidential medical records.

After the PR plus system information: who should be involved?

Poll 12 · 8 responses



The initial majority wanted a peer; the PR summary most often left oversight unchanged. In the final vote, 100% wanted a peer or team. This is consistent with the workshop's emphasis on consequences and context, even for a small diff.

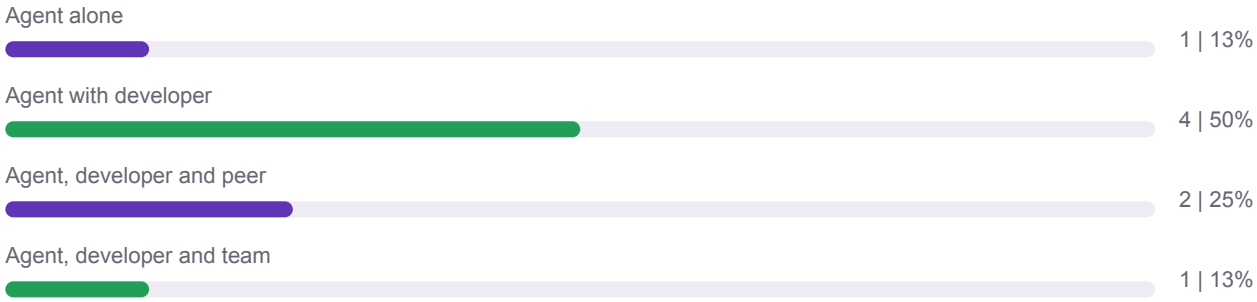
Oversight also depends on how work is shaped

Poll 13 · How much oversight does PR #482 need?

“Improve the survey editor” combined a behavior-preserving refactor, toolbar changes and a database ordering migration: 2,146 changed lines across 10 files, with all checks passing.

The room’s choices for PR #482

Poll 13 · 8 responses



50% chose an agent with a developer; 38% added a peer or team, and 13% chose an agent alone. The spread is part of the result: the room did not converge on a single oversight level for this bundled change.

Separate the concerns, focus the review

The workshop then separated the PR into review questions: preserve behavior in the refactor, confirm the toolbar experience, and establish a safe data transition. The database work was further staged into expansion, migration and switching, then removal of the old path. This approach makes each review more focused and gives the consequential data changes their own oversight plan.

> Take this back to your next change: agree the risk, give review a specific focus, choose who needs to be involved, and record what was established.

A practical review conversation

Before implementation: What could fail, what intent must be preserved, and who needs to understand the result?
During review: What should AI investigate, what should a person challenge, and what would trigger more oversight?
At the decision: What evidence is sufficient, what remains uncertain, and who owns the outcome?