Human-in-the-Loop AI Technical Evaluation

Designing expert review systems that are consistent, efficient and accountable

Sunday Adetona Aderinto

Global IT Specialist | Software Engineer | AI Technical Evaluator | Computer Engineering Researcher

Professional Insight Paper	September 2026
Domains	Software engineering | Trustworthy AI | Distributed systems

Core proposition

Human evaluation is central to improving AI systems that generate code and technical explanations. Yet unstructured review can be inconsistent and difficult to learn from. This paper explains how to design calibrated expert-evaluation operations with clear criteria, adjudication, quality assurance and feedback loops.

Executive abstract

Human evaluation is central to improving AI systems that generate code and technical explanations. Yet unstructured review can be inconsistent and difficult to learn from. This paper explains how to design calibrated expert-evaluation operations with clear criteria, adjudication, quality assurance and feedback loops.

1. Why experts remain necessary

Automated metrics can confirm compilation, test outcomes and detectable vulnerabilities, but they do not fully capture ambiguous requirements, architectural appropriateness or the quality of technical reasoning. Experts interpret context, identify silent assumptions and judge whether an answer would mislead a developer.

2. Task and rubric design

Good evaluation begins with a precise task, reference evidence and criteria that distinguish independent dimensions. Anchored scales should describe observable behaviour at each level. Critical defects need explicit severity rules. Asking reviewers to provide a reason code and concise rationale makes disagreement diagnosable.

3. Calibration and reliability

Reviewers should score shared examples, compare reasoning and resolve interpretation differences before production work. Statistical agreement is useful but should not replace qualitative analysis. Low agreement may reveal unclear prompts, overlapping criteria, insufficient expertise or genuine technical ambiguity.

4. Adjudication and quality control

Sample double-review, targeted audits and escalation pathways balance quality with cost. Adjudicators should not merely choose a preferred score; they should identify the governing evidence and update guidance when a recurring ambiguity appears. Reviewer performance must be assessed fairly across task difficulty.

5. Feedback as a learning system

Evaluation data is valuable when it is structured by error type, severity, language, domain and remediation. Trend analysis can expose systematic failures and identify where training data, prompting, tooling or model behaviour should change. Reviewers also need feedback on overturned decisions to improve consistency.

6. Responsible operation

Protect proprietary code, minimise personal data and separate reviewer identity from unnecessary analytics. Define conflicts of interest, access controls and retention. Human oversight should have real authority to block unsafe outputs, and accountability should remain clear when AI recommendations are accepted.

Practitioner takeaways

• Use observable, anchored evaluation criteria.

• Measure and investigate disagreement.

• Convert adjudication outcomes into updated guidance.

• Give human oversight genuine decision authority.

Conclusion

Responsible technology leadership requires evidence, explicit trade-offs and accountable human judgement. The frameworks in this paper are intended to support disciplined implementation and to create a foundation for deeper empirical research.

Selected references

NIST, Artificial Intelligence Risk Management Framework.

ISO/IEC 42001, Artificial intelligence management systems.

ISO/IEC 25010, Systems and software quality models.

Krippendorff, Content Analysis: An Introduction to Its Methodology.

About the author

Sunday Adetona Aderinto is a senior software engineer and Computer Engineering researcher with more than ten years of experience across full-stack systems, APIs, databases, infrastructure, technical instruction and AI-generated code evaluation. His interests include trustworthy AI, federated learning, differential privacy and production software quality.