www.revistaft.com.br

Revista FT | ISSN 1678-0817 | v. 30, n. 159, 2026 | DOI: 10.69849/efmbbt16

Qualidade de IA generativa em engenharia de software: estudo comparativo entre Claude e Codex

Generative AI quality in software engineering: a comparative study between Claude and Codex

Paulo Henrique da Silva Martins[1]

Resumo: As Inteligências Artificiais Generativas (IAGs) têm sido crescentemente adotadas como apoio ao ensino e à aprendizagem no ensino superior, inclusive na área de computação. Contudo, a qualidade técnica das respostas que essas ferramentas oferecem a questões de Engenharia de Software raramente é avaliada de forma sistemática e comparativa. Este estudo teve por objetivo avaliar comparativamente a qualidade das respostas geradas por duas IAGs — Claude Sonnet 4.6 e Codex (GPT-5.5) — a dez questões técnicas de Engenharia de Software, sob a ótica do uso educacional. Adotou-se um estudo de caso comparativo de abordagem mista, com pontuação em rubrica de três dimensões — precisão, clareza e profundidade — e registro de erros, complementado pela verificação objetiva do código por execução em ambiente controlado. Os resultados apontaram equivalência plena em precisão e clareza, sem ocorrência de erros, e diferença estatisticamente significativa apenas na profundidade, favorável ao Claude e mais acentuada nas questões práticas. Conclui-se que ambas as ferramentas são adequadas como apoio à aprendizagem em conteúdos fundamentais, com perfis complementares, e que a profundidade constitui critério decisivo de qualidade educacional. O estudo contribui metodologicamente ao acoplar à avaliação por rubrica a verificação reproduzível por execução de código.

Palavras-chave: Inteligência Artificial Generativa. Engenharia de Software. Avaliação da qualidade. Educação superior.

Abstract: Generative Artificial Intelligence (GenAI) tools have been increasingly adopted to support teaching and learning in higher education, including in computing. However, the technical quality of the answers these tools provide to Software Engineering questions is rarely assessed in a systematic and comparative way. This study aimed to comparatively evaluate the quality of the answers generated by two GenAI tools — Claude Sonnet 4.6 and Codex (GPT-5.5) — to ten technical Software Engineering questions, from an educational standpoint. A comparative case study with a mixed-methods approach was adopted, scoring three dimensions — accuracy, clarity, and depth — on a rubric and recording errors, complemented by objective verification of the generated code through execution in a controlled environment. The results showed full equivalence in accuracy and clarity, with no errors, and a statistically significant

[1] Graduando em Engenharia de Software pelo Instituto de Ensino Superior iCEV, Teresina, PI, Brasil. E-mail: paulo_henrique.martins@somosicev.com.

difference only in depth, favoring Claude and more pronounced in the practical questions. It is concluded that both tools are suitable as learning support for fundamental content, with complementary profiles, and that depth is a decisive criterion of educational quality. The study contributes methodologically by coupling rubric-based assessment with reproducible verification through code execution.

Keywords: Generative Artificial Intelligence. Software Engineering. Quality assessment. Higher education. Case study.

1 INTRODUÇÃO

A contínua evolução tecnológica no campo da computação culminou na ampla disseminação da Inteligência Artificial Generativa (IAG), materializada em modelos de linguagem de grande escala (LLMs) como o ChatGPT, o Gemini e o Copilot. Tais ferramentas representam um avanço disruptivo, sendo capazes de interpretar dados externos, aprender a partir deles e utilizar essa aprendizagem para gerar textos, códigos e outros conteúdos coerentes (Abbas; Jam; Khan, 2024). A introdução dessas aplicações para a geração de textos e códigos, o suporte à pesquisa e a conclusão de projetos acadêmicos tem transformado o ensino superior, sobretudo pela funcionalidade de fornecer respostas instantâneas que auxiliam na pesquisa de informações, no esclarecimento de conceitos complexos e na produção textual (Henning et al., 2023).

Na área de computação e, particularmente, na Engenharia de Software, essas ferramentas passaram a ser consultadas como apoio à resolução de problemas técnicos, à compreensão de padrões de projeto e à escrita e revisão de código. A literatura reconhece benefícios expressivos dessa adoção aumento do engajamento e da autonomia, feedback rápido e ampliação do acesso ao conhecimento, inclusive em uma perspectiva de inclusão educacional (Zhao et al., 2023; Osorio et al., 2024; Baidoo-Anu; Ansah, 2023), mas alerta para riscos relevantes, como a superficialidade cognitiva e a dependência excessiva, capazes de comprometer habilidades como a criatividade e o pensamento crítico (Larios et al., 2025; Abbas; Jam; Khan, 2024), além de desafios de integridade acadêmica e de regulação institucional (Bond et al., 2024; Akanzire et al., 2025; Jin et al., 2025).

No contexto específico da Engenharia de Software, o uso indiscriminado de IAGs, ainda que otimize a produtividade, pode levar à diminuição no desenvolvimento de habilidades fundamentais, como a criatividade e o pensamento crítico, competências essenciais à resolução de problemas complexos na área (Santos et al., 2024). A dependência tecnológica acarreta o

risco de os estudantes tenderem a aceitar informações sem questionamento, o que torna premente compreender criticamente a qualidade do que essas ferramentas efetivamente entregam.

Diante disso, justifica-se a investigação pela crescente dependência discente dessas respostas como recurso de estudo, pela necessidade de critérios objetivos para a seleção institucional de ferramentas e pela referida lacuna na verificação empírica da qualidade técnica, que ainda consiste, majoritariamente, em discussões conceituais. Ao avaliar comparativamente duas ferramentas amplamente utilizadas e ao acoplar à avaliação por rubrica a verificação por execução de código, o estudo busca oferecer evidências úteis tanto à prática pedagógica quanto à reflexão sobre o uso responsável das IAGs, auxiliando as instituições a repensarem currículos e práticas para que a IA atue como inteligência aumentada, sem comprometer o desenvolvimento cognitivo dos futuros profissionais.

O objetivo geral consiste em avaliar comparativamente a qualidade das respostas geradas por duas Inteligências Artificiais Generativas — Claude Sonnet 4.6 e Codex (GPT-5.5) — a questões técnicas de Engenharia de Software, sob a ótica do uso educacional. Como objetivos específicos, definiram-se: elaborar um instrumento com dez questões abrangendo diferentes áreas de conhecimento da Engenharia de Software; definir a avaliação contemplando as dimensões de precisão, clareza e profundidade, além de uma taxonomia de erros; comparar o desempenho das ferramentas e analisar as implicações para o uso educacional.

Para a consecução dos objetivos propostos, delineou-se uma pesquisa aplicada de abordagem mista com caráter exploratório e descritivo, operacionalizada por meio de um estudo de caso comparativo cuja unidade de análise consistiu nas respostas geradas pelas ferramentas. O instrumento de coleta foi composto por dez questões técnicas (equilibradas entre conceituais e práticas de programação) fundamentadas nas áreas do *Guide to the Software Engineering Body of Knowledge* (SWEBOK). A análise dos dados baseou-se em uma rúbrica dimensional (precisão, clareza e profundidade), em uma taxonomia de severidade de erros e, de forma inovadora, na verificação empírica e cega dos códigos gerados por meio de execução em ambiente controlado (Python, Node.js, pytest, TypeScript e SQLite).

O artigo está organizado em cinco seções. Após esta introdução, a primeira seção apresenta a metodologia, a segunda descreve a metodologia; a terceira apresenta os resultados e a análise comparativa; a quarta discute os achados à luz da literatura; e a quinta traz as considerações finais.

2 METODOLOGIA

Quanto à sua natureza, esta pesquisa caracteriza-se como aplicada, pois visa gerar conhecimentos para aplicação prática na seleção e no uso educacional de Inteligências Artificiais Generativas. Quanto à abordagem, adota uma perspectiva mista, articulando dados quantitativos (pontuações atribuídas em rubrica) e qualitativos (análise de erros e do conteúdo das respostas), combinação que, segundo Gil (2019), favorece uma compreensão mais ampla do fenômeno investigado. Quanto aos objetivos, classifica-se como exploratória e descritiva, uma vez que examina um tema ainda pouco sistematizado e descreve, de forma comparativa, as características das respostas analisadas (Gil, 2019; Lakatos; Marconi, 2017).

Quanto aos procedimentos técnicos, optou-se pelo estudo de caso, estratégia adequada à investigação aprofundada de um fenômeno contemporâneo em seu contexto real (Gil, 2019). No campo da Engenharia de Software, o estudo de caso é amplamente reconhecido como método empírico apropriado quando se busca compreender artefatos e comportamentos em condições próximas às de uso (Runeson; Höst, 2009). O caso, aqui, é definido pelo comportamento de duas ferramentas de IAG diante de um mesmo conjunto de tarefas técnicas, sendo a unidade de análise cada resposta gerada.

O instrumento de coleta foi constituído por dez questões técnicas, distribuídas entre diferentes áreas de conhecimento da Engenharia de Software, tomando-se como referência as áreas descritas no *Guide to the Software Engineering Body of Knowledge* (Bourque; Fairley, 2014). As questões foram equilibradas entre conceituais e práticas — estas últimas exigindo a produção de código e distribuídas em diferentes graus de dificuldade, de modo a abranger requisitos, arquitetura, padrões de projeto, testes, algoritmos, qualidade e refatoração, controle de versão, segurança, banco de dados e processo de software.

A coleta foi conduzida sob protocolo controlado: as duas ferramentas receberam enunciados idênticos, em sessões independentes e sem interações de acompanhamento, na mesma data e horário, registrando-se os metadados de proveniência (versão do modelo, data e condições). Esse cuidado com a padronização e o registro das condições de coleta é condição para a reprodutibilidade e o rigor exigidos da pesquisa científica (Lakatos; Marconi, 2017).

Para a operacionalização das variáveis, definiu-se uma rubrica de avaliação em três dimensões — precisão, clareza e profundidade —, mensuradas em escala de 1 a 5, acompanhada de uma taxonomia de erros (factual, de código, de omissão, de alucinação, de desatualização e de inconsistência) e de uma classificação de severidade. A transformação de conceitos abstratos,

como qualidade, em indicadores observáveis e mensuráveis corresponde ao processo de operacionalização descrito por Lakatos e Marconi (2017) como etapa essencial do delineamento metodológico.

O procedimento de avaliação combinou três salvaguardas. Primeiro, elaborou-se um gabarito de referência para cada questão, com base em fontes consolidadas, utilizado como âncora da dimensão de precisão. Segundo, adotou-se a avaliação cega, com as respostas identificadas apenas como “A” e “B” durante a pontuação. Terceiro, e como diferencial, a precisão das questões práticas não dependeu exclusivamente do julgamento: o código produzido por ambas as ferramentas foi executado em ambiente controlado (Python, Node.js, pytest, TypeScript e SQLite), comparando-se sua saída à esperada.

A análise dos dados empregou estatística descritiva (médias, medianas e desvios-padrão por dimensão, ferramenta e tipo de questão) e, para a comparação da dimensão de profundidade, o teste não-paramétrico de Wilcoxon para amostras pareadas, adequado ao caráter ordinal e pareado dos dados (Gil, 2019). A análise qualitativa, por sua vez, examinou a natureza dos erros e das diferenças observadas. Todos os materiais prompts, respostas brutas, rubrica, código-fonte e logs de execução foram organizados em um pacote de replicação, em consonância com as exigências de transparência e reprodutibilidade da pesquisa científica (Lakatos; Marconi, 2017; Runeson; Höst, 2009).

3 RESULTADOS E ANÁLISE COMPARATIVA

Esta seção apresenta e interpreta os resultados do estudo. Expõem-se, na sequência, os resultados quantitativos agregados, o detalhamento por questão e por tipo, a verificação objetiva por execução de código e a análise qualitativa que sustenta a leitura comparativa entre as duas ferramentas.

A Tabela 1 sintetiza as pontuações médias por dimensão. As duas ferramentas obtiveram desempenho idêntico em precisão e clareza, divergindo exclusivamente na profundidade, dimensão que, por consequência, concentra a diferença observada na média geral. Não foi registrado nenhum erro factual, de código, de omissão crítica ou de alucinação em nenhuma das vinte respostas avaliadas.

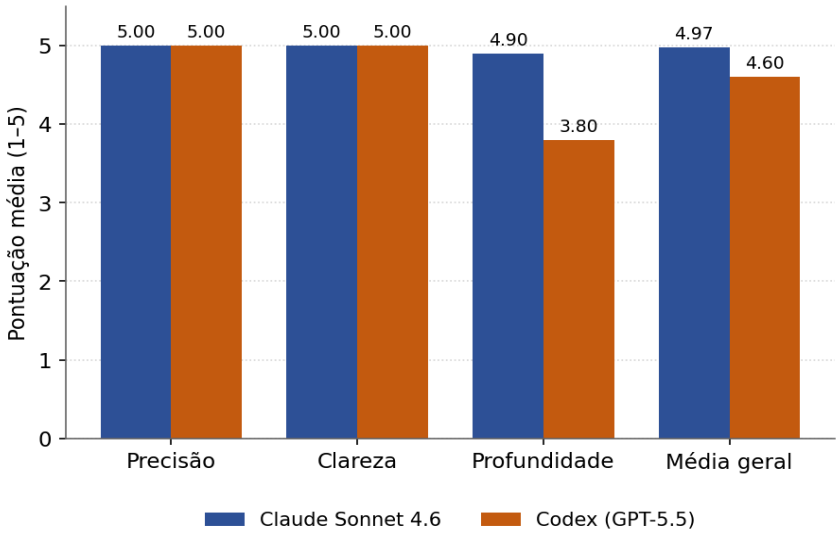
Tabela 1 – Pontuações médias por dimensão e total de erros

Dimensão	Claude	Codex	Diferença (C−Cx)
Precisão	5,00	5,00	0,00
Clareza	5,00	5,00	0,00
Profundidade	4,90	3,80	+1,10
Média geral	4,97	4,60	+0,37
Total de erros	0	0	—

Fonte: elaborado pelo autor (2026).

Os dados da Tabela 1 são reapresentados graficamente na Figura 1, que torna visível a sobreposição das ferramentas em precisão e clareza e o distanciamento na profundidade. A representação reforça que a vantagem observada na média geral do Claude decorre integralmente dessa única dimensão, mantidas iguais as demais.

Figura 1 – Pontuações médias por dimensão



Fonte: elaborado pelo autor (2026).

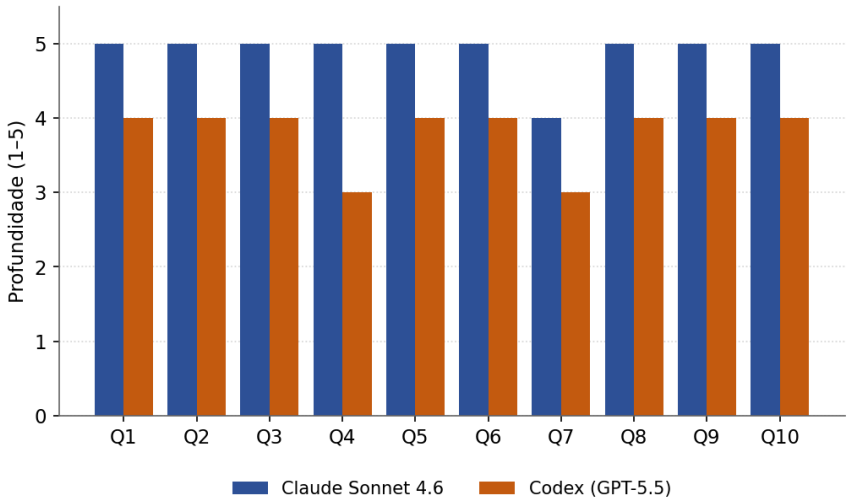
Como se observa na Figura 1, as barras de precisão e de clareza coincidem no valor máximo para as duas ferramentas, ao passo que a barra de profundidade do Codex situa-se visivelmente abaixo da correspondente ao Claude, antecipando o eixo de diferenciação analisado nas subseções seguintes.

Em precisão e em clareza, ambas as ferramentas atingiram a pontuação máxima em todas as dez questões. A equivalência em precisão não se baseou apenas no cotejo com o gabarito de

referência: nas questões práticas, foi confirmada por execução. Todos os artefatos de código compilaram e executaram corretamente, produzindo as saídas esperadas — incluindo a aprovação integral das suítes de teste (24 de 24 casos no validador testado pelo Claude e 7 de 7 asserções no testado pelo Codex), a reprodução correta do defeito proposto na questão de algoritmos e o retorno idêntico das consultas SQL. A equivalência em clareza reflete que ambas as respostas foram bem estruturadas e compreensíveis, ainda que com estilos distintos: o Codex tendeu a respostas mais enxutas e o Claude a respostas mais extensas e segmentadas.

A profundidade foi a única dimensão a distinguir as ferramentas. O Claude obteve média 4,90 (desvio-padrão 0,32; mediana 5,0) e o Codex, 3,80 (desvio-padrão 0,42; mediana 4,0). A diferença foi favorável a Claude em todas as dez questões, sem inversões. Aplicado o teste de Wilcoxon para amostras pareadas, mostrou-se estatisticamente significativa ($W = 0$; $p = 0{,}002$), com tamanho de efeito máximo (correlação bisserial de postos = 1,00). A distribuição questão a questão é apresentada na Figura 2.

Figura 2 – Pontuação de profundidade por questão (Q1–Q10)



Fonte: elaborado pelo autor (2026).

A Figura 2 evidencia o padrão: o Codex manteve-se majoritariamente no nível 4, com queda ao nível 3 nas questões de testes (Q4) e de versionamento (Q7); o Claude permaneceu no nível 5, exceto também na questão de versionamento. A diferença não decorre de erros inexistentes, mas do grau de elaboração: cobertura de trade-offs, casos de borda, justificativas e alternativas, conforme o descritor do nível máximo da rubrica.

Segmentando a profundidade por natureza da questão, a diferença foi consistente em ambos os grupos, porém mais acentuada nas questões práticas, conforme a Tabela 2. Nas

conceituais (Q1, Q2 e Q10), a profundidade média foi 5,00 para o Claude e 4,00 para o Codex; nas práticas (Q3 a Q9), 4,86 e 3,71, respectivamente.

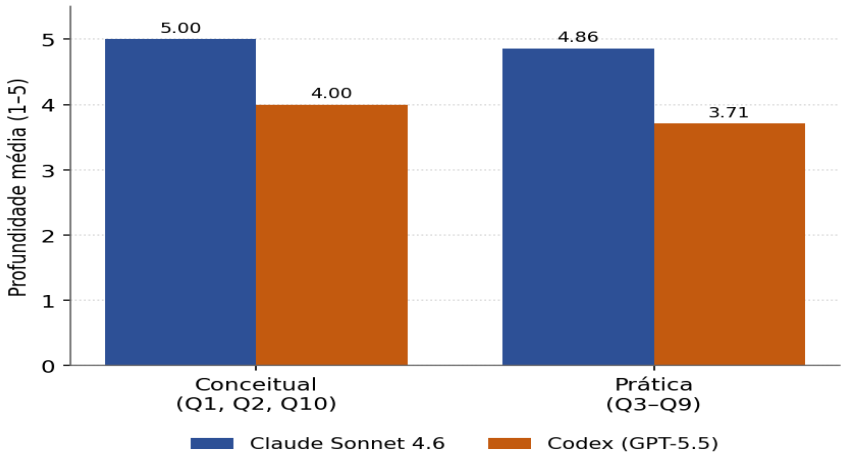
Tabela 2 – Profundidade média por tipo de questão

Tipo de questão	**Claude**	**Codex**
Conceitual (Q1, Q2, Q10)	5,00	4,00
Prática (Q3–Q9)	4,86	3,71

Fonte: elaborado pelo autor (2026).

A Figura 3 ilustra essa comparação. Nota-se que a distância entre as ferramentas é maior no grupo das questões práticas, achado relevante para o uso educacional: nas tarefas em que o estudante busca não só a solução correta, mas o entendimento das decisões de projeto, a diferença de elaboração foi mais expressiva. Ambas entregaram soluções corretas e executáveis; divergiram na riqueza da explicação que as acompanha.

Figura 3 – Profundidade média por tipo de questão (conceitual × prática)



Fonte: elaborado pelo autor (2026).

Como mostra a Figura 3, ainda que o Claude também supere o Codex nas questões conceituais, é nas práticas que a lacuna se amplia, o que sugere que a produção de código acompanhada de explicação aprofundada constitui o terreno em que as ferramentas mais se distinguem.

Para reduzir a subjetividade na avaliação da precisão das questões práticas, o código de ambas as ferramentas foi executado em ambiente controlado (Python 3.12, Node.js 22, pytest 9, TypeScript via compilador tsc e SQLite). O Quadro 2 resume a verificação; a reprodutibilidade integral encontra-se no pacote de replicação.

Quadro 2 – Síntese da verificação por execução (questões práticas)

QUEST.	Verificação executada	Claude	Codex
Q3	Cálculo de frete (padrão Strategy)	OK	OK
Q4	Suíte de testes do validador	24/24 OK	7/7 OK
Q5	Defeito reproduzido + versão O(n)	OK	OK
Q6	Refatoração SOLID (execução)	OK	OK
Q9	Consulta de total por cliente	OK	OK

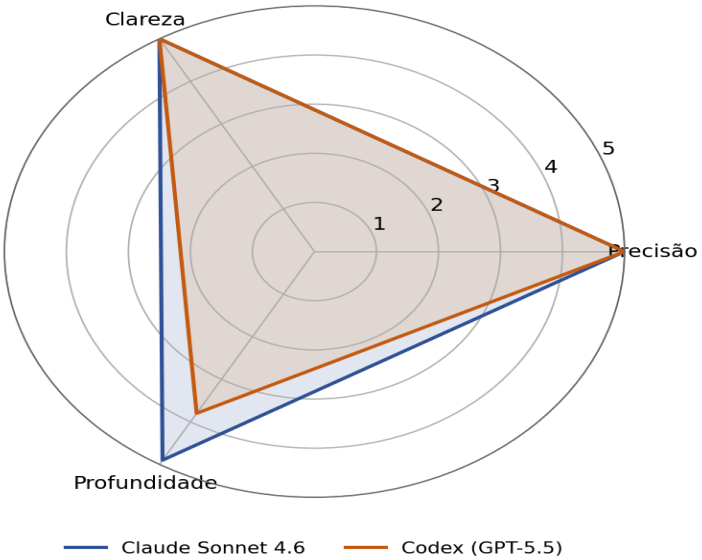
Fonte: elaborado pelo autor (2026).

O Quadro 2 demonstra que a totalidade dos artefatos executados comportou-se conforme o esperado em ambas as ferramentas, o que sustenta empiricamente a atribuição de pontuação máxima em precisão e desloca a verificação da correção do plano do julgamento para o da evidência reproduzível.

Não houve, em nenhuma das vinte respostas, erro factual, de código, de omissão essencial, de alucinação, de desatualização ou de inconsistência interna; a taxonomia de erros retornou contagem zero para ambas as ferramentas, indicando maturidade em conteúdos fundamentais e estáveis da área. A inspeção qualitativa esclarece a origem da diferença de profundidade. Em diversas questões, o Claude incorporou elementos adicionais — a Lei de Conway, o monólito modular e o padrão Saga na arquitetura; os cinco princípios SOLID, interfaces segregadas e injeção de dependências na refatoração; e, na segurança, a ampliação para além da injeção de SQL, tratando de XSS, política de segurança de conteúdo e atributos de cookies.

O Codex, por sua vez, produziu respostas corretas, claras e concisas, com méritos próprios: na segurança, percebeu que o trecho vulnerável também armazenava a senha de forma inadequada e propôs hash forte (bcrypt/Argon2), menor privilégio e registro de tentativas; no processo, acrescentou a observação de que a dívida técnica pode ser uma escolha consciente e legítima, desde que visível e gerida. Suas perdas de profundidade mais nítidas ocorreram na questão de testes em que a justificativa da estratégia de cobertura, solicitada, foi reduzida a uma frase e na de versionamento. O perfil global das ferramentas é sintetizado na Figura 4.

Figura 4 – Perfil comparativo das três dimensões avaliadas



Fonte: elaborado pelo autor (2026).

A Figura 4 confirma visualmente a leitura comparativa: as duas áreas praticamente se sobrepõem nos eixos de precisão e clareza, distanciando-se apenas no eixo de profundidade, em que a área do Claude se projeta para além da do Codex. Essa representação resume o achado central do estudo e prepara a discussão desenvolvida na seção seguinte.

4 DISCUSSÃO

Os resultados dialogam diretamente com o debate sobre a incorporação das IAGs no ensino superior. Enquanto a literatura se concentra, predominantemente, em usos voltados à escrita e à pesquisa acadêmica (Gallent-Torres et al., 2023; Henning et al., 2023; Zhao et al., 2023), este estudo desloca o foco para a avaliação da qualidade técnica das respostas em um domínio específico, oferecendo evidências que permitem qualificar tanto os benefícios quanto os riscos apontados pela produção científica recente.

A equivalência plena em precisão e clareza confirmada, nas questões práticas, pela execução do código converge com a literatura que atribui às IAGs potencial de apoio ao aprendizado, à organização do raciocínio e à clareza textual (Henning et al., 2023; Zhao et al., 2023; Baidoo-Anu; Ansah, 2023), bem como de feedback rápido e ampliação do acesso ao conhecimento, inclusive de forma inclusiva (Gallent-Torres et al., 2023; Osorio et al., 2024). Cumpre delimitar o alcance: o instrumento abrangeu conteúdos fundamentais e estáveis, e a

ausência de erros nesse recorte não autoriza generalização para conteúdos avançados ou emergentes.

A profundidade foi o único eixo de diferenciação, com vantagem consistente do Claude. O achado é significativo por oferecer uma medida observável a uma preocupação que a literatura formula de modo abstrato: o risco de superficialidade cognitiva (Larios et al., 2025). Respostas corretas, porém pouco elaboradas, podem induzir aprendizagem rasa quando consumidas sem mediação, alinhando-se ao alerta de que a dependência excessiva pode comprometer o pensamento crítico (Abbas; Jam; Khan, 2024). Sob a ótica educacional, a correção não é condição suficiente para o valor pedagógico: duas ferramentas igualmente precisas podem oferecer experiências de aprendizado distintas conforme o grau de elaboração de suas explicações.

A inexistência de erros e a verificação objetiva por execução respondem, em parte, à demanda por maior rigor metodológico na pesquisa sobre IA na educação (Zawacki-Richter et al., 2019; Bond et al., 2024). A estratégia de submeter o código a execução controlada constitui contribuição metodológica, ao transformar a aferição de precisão em evidência reproduzível. O cuidado conecta-se ao debate sobre integridade acadêmica, pois a capacidade das IAGs de gerar soluções corretas intensifica preocupações com autoria e avaliação autêntica (Gallent-Torres et al., 2023; Kofinas; Tsay; Pike, 2025); a confiabilidade observada não autoriza, portanto, confiança acrítica.

Ao evidenciar perfis de qualidade igualmente válidos em correção, mas divergentes em profundidade, o estudo fornece subsídios para a seleção criteriosa de ferramentas, reforçando a literatura que defende políticas institucionais claras (Akanzire et al., 2025; Bond et al., 2024; Jin et al., 2025). O próprio protocolo rubrica acompanhada de verificação por execução pode ser apropriado por instituições para avaliar ferramentas antes de sua adoção. Em consonância com as diretrizes éticas internacionais (Unesco, 2022), sugere-se a declaração do uso de IA e a revisão dos instrumentos avaliativos, privilegiando produções autorais (Gallent-Torres et al., 2023; Henning et al., 2023).

A leitura qualitativa sugere perfis complementares: uma ferramenta privilegia a abrangência explicativa, favorável ao estudo aprofundado, e a outra, a objetividade, adequada à consulta rápida. Essa complementaridade abre espaço para estratégias pedagógicas que combinem ferramentas conforme o objetivo de aprendizagem, em sintonia com a tendência de personalização e de aprendizagem adaptativa (Santos et al., 2025; Lee et al., 2024), cabendo ao

docente o papel mediador de orientar quando e como cada tipo de resposta melhor serve ao processo formativo.

Diante disso, os achados devem ser lidos à luz de limitações: dez questões, duas ferramentas e um único ponto no tempo restringem a generalização, sobretudo porque os modelos são atualizados continuamente. Reconhecer tais limites é coerente com o chamado por rigor e transparência da área.

5 CONSIDERAÇÕES FINAIS

Este estudo de caso comparou a qualidade das respostas de duas Inteligências Artificiais Generativas — Claude Sonnet 4.6 e Codex (GPT-5.5) — a dez questões técnicas de Engenharia de Software, sob a ótica do uso educacional. Os resultados indicaram equivalência plena em precisão e clareza, ambas confirmadas, nas questões práticas, por execução do código, e divergência significativa apenas na profundidade, favorável ao Claude e mais acentuada nas questões práticas, sem ocorrência de erros.

Conclui-se que ambas as ferramentas se mostram adequadas como apoio à aprendizagem em conteúdos fundamentais, desde que mediadas criticamente, e que apresentam perfis complementares uma orientada ao aprofundamento, a outra à objetividade. O principal aporte é duplo: de um lado, evidência empírica que ancora, em métricas observáveis, a preocupação da literatura com a superficialidade cognitiva; de outro, a contribuição metodológica de acoplar à rubrica a verificação objetiva do código por execução, ampliando o rigor e a reprodutibilidade.

Como sugestão para pesquisas futuras, recomenda-se ampliar o número de questões e de ferramentas, contemplar conteúdos avançados e emergentes, repetir a coleta em diferentes momentos para capturar a evolução dos modelos, fortalecendo a validade dos achados e a consolidação de critérios robustos de avaliação de IAGs com finalidade educacional.

REFERÊNCIAS

ABBAS, Muhammad; JAM, Farooq Ahmed; KHAN, Tariq Iqbal. Is it harmful or helpful? Examining the causes and consequences of generative AI usage among university students. **International Journal of Educational Technology in Higher Education**, v. 21, n. 1, p. 10, fev. 2024. Disponível em: https://doi.org/10.1186/s41239-024-00444-7. Acesso em: 28 set. 2025.

AKANZIRE, Bismark Nyaaba; NYAABA, Matthew; NABANG, Machário. IA generativa na formação de professores: percepção e preparação dos formadores de professores. **Revista de Tecnologia Educacional Digital**, v. 1, p. 1-17, 2025. Disponível em:

https://www.jdet.net/article/generative-ai-in-teacher-education-teacher-educators-perception-and-preparedness-15887. Acesso em: 10 set. 2025.

AKGUN, Selin; GREENHOW, Christine. Inteligência artificial na educação: abordando desafios éticos em contextos de ensino fundamental e médio. **AI and Ethics**, v. 2, n. 3, p. 431-440, 2022.

BAIDOO-ANU, David; ANSAH, Leticia Owusu. Educação na era da inteligência artificial generativa (IA): compreendendo os benefícios potenciais do ChatGPT na promoção do ensino e da aprendizagem. **Journal of AI**, v. 7, n. 1, p. 52-62, 2023. Disponível em: https://dergipark.org.tr/en/doi/10.61969/jai.1337500. Acesso em: 28 nov. 2025.

BARBOSA, R.; PORTES, L. Modelos generativos de linguagem natural no ensino superior: potenciais e limites. **Revista de Educação e Tecnologia**, v. 2, p. 47-61, 2023.

BOMMASANI, Rishi et al. On the opportunities and risks of foundation models. **arXiv**, 2022. Disponível em: http://arxiv.org/abs/2108.07258. Acesso em: 26 nov. 2025.

BOND, Melissa et al. A meta systematic review of artificial intelligence in higher education: a call for increased ethics, collaboration, and rigour. **International Journal of Educational Technology in Higher Education**, v. 21, n. 4, p. 1-41, jan. 2024. Disponível em: https://doi.org/10.1186/s41239-023-00436-z. Acesso em: 1 set. 2025.

BOURQUE, Pierre; FAIRLEY, Richard E. (ed.). **Guide to the Software Engineering Body of Knowledge (SWEBOK)**: version 3.0. Los Alamitos: IEEE Computer Society, 2014.

DOSOVITSKIY, Alexey et al. An image is worth 16x16 words: transformers for image recognition at scale. **arXiv**, 2021. Disponível em: http://arxiv.org/abs/2010.11929. Acesso em: 26 nov. 2025.

GALLENT-TORRES, Cinta; ZAPATA-GONZÁLEZ, Alfredo; ORTEGO-HERNANDO, José Luis. The impact of Generative Artificial Intelligence in higher education: a focus on ethics and academic integrity. **RELIEVE**, v. 29, n. 2, art. M5, 2023. Disponível em: http://doi.org/10.30827/relieve.v29i2.29134. Acesso em: 1 set. 2025.

GIL, Antônio Carlos. **Métodos e técnicas de pesquisa social**. 7. ed. São Paulo: Atlas, 2019.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. **Deep learning**. Cambridge: MIT Press, 2016.

HENNING, Mauricio et al. Impactos da Inteligência Artificial na Educação Superior: uma revisão da literatura. In: COLÓQUIO INTERNACIONAL DE GESTIÓN UNIVERSITARIA, 22., 2023, Assunção. **Anais [...]**. Assunção: XXII CIGU, 2023. Disponível em: https://repositorio.ufsc.br/bitstream/handle/123456789/253863/1230145.pdf. Acesso em: 10 set. 2025.

HOLMES, Wayne; BIALIK, Maya; FADEL, Charles. **Artificial intelligence in education**: promises and implications for teaching and learning. Boston: Center for Curriculum Redesign, 2019.

JIN, Yueqiao et al. Generative AI in higher education: a global perspective of institutional adoption policies and guidelines. **Computers and Education: Artificial Intelligence**, v. 8, p. 100-348, 2025. Disponível em: https://www.sciencedirect.com/science/article/pii/S2666920X24001516. Acesso em: 15 nov. 2025.

KOFINAS, Alexander K.; TSAY, Crystal Han-Huei; PIKE, David. The impact of generative AI on academic integrity of authentic assessments within a higher education context. **British Journal of Educational Technology**, v. 56, n. 4, p. 2549-2565, 2025. Disponível em: https://bera-journals.onlinelibrary.wiley.com/doi/full/10.1111/bjet.13585. Acesso em: 17 nov. 2025.

LAKATOS, Eva Maria; MARCONI, Marina de Andrade. **Fundamentos de metodologia científica**. 8. ed. São Paulo: Atlas, 2017.

LARIOS SOLDEVILLA, Omar Antonio et al. Adoption of AI tools and their impact on the academic research output of business school students. **Cogent Education**, v. 12, n. 1, 2025. Disponível em: https://doi.org/10.1080/2331186X.2025.2556892. Acesso em: 20 nov. 2025.

LECUN, Yann; BENGIO, Yoshua; HINTON, Geoffrey. Deep learning. **Nature**, v. 521, n. 7553, p. 436-444, 2015. Disponível em: https://www.nature.com/articles/nature14539. Acesso em: 16 nov. 2025.

LEE, Daniel et al. The impact of generative AI on higher education learning and teaching: a study of educators' perspectives. **Computers and Education: Artificial Intelligence**, v. 6, p. 100221, 2024. Disponível em: https://www.sciencedirect.com/science/article/pii/S2666920X24000225. Acesso em: 12 nov. 2025.

OSORIO, Celia et al. Enhancing accessibility to analytics courses in higher education through AI, simulation, and e-collaborative tools. **Information**, v. 15, p. 4-30, 2024. Disponível em: https://doi.org/10.3390/info15080430. Acesso em: 20 nov. 2025.

RIEDNER, D. D. T. Expediente. **Revista Edutec - Educação, Tecnologias Digitais e Formação Docente**, [S. l.], v. 2, n. 1, p. 1-5, 25 set. 2022. Disponível em: https://doi.org/10.55028/edutec.v2i1.17267. Acesso em: 30 nov. 2023.

RUNESON, Per; HÖST, Martin. Guidelines for conducting and reporting case study research in software engineering. **Empirical Software Engineering**, v. 14, n. 2, p. 131-164, 2009.

SANTOS, Carlos Eduardo Vieira et al. Inteligência Artificial Generativa na Educação: uma pesquisa sobre impactos e percepções. **South American Development Society Journal**, v. 10, n. 29, p. 222-240, 2024.

SANTOS, F. V. S. dos et al. **Inteligência Artificial no Ensino Superior**: desenvolvendo o aprendizado para o futuro. 1. ed. Alfenas: UNIFAL-MG, 2025. Disponível em: https://www.unifal-mg.edu.br/bibliotecas/wp-content/uploads/sites/125/2025/03/Inteligencia-Artificial-no-Ensino-Superior.pdf. Acesso em: 16 nov. 2025.

TEIXEIRA, Magno Felipe Holanda Barboza; GOMES FILHO, Lourival. O impacto da inteligência artificial entre estudantes do ensino superior. **Revista Eletrônica da Estácio Recife**, v. 12, n. 1, p. 86-100, 2025. Disponível em: https://reer.emnuvens.com.br/reer/article/view/852. Acesso em: 23 nov. 2025.

TOUVRON, Hugo et al. LLaMA: open and efficient foundation language models. **arXiv**, 2023. Disponível em: http://arxiv.org/abs/2302.13971. Acesso em: 26 nov. 2025.

UNESCO. **Recomendação sobre a ética da inteligência artificial**. Paris: UNESCO, 2022. Disponível em: https://unesdoc.unesco.org/ark:/48223/pf0000381137. Acesso em: 17 nov. 2025.

VASWANI, Ashish et al. Attention is all you need. **Advances in Neural Information Processing Systems**, v. 30, 2017. Disponível em: https://arxiv.org/abs/1706.03762. Acesso em: 26 nov. 2025.

ZAWACKI-RICHTER, Olaf et al. Systematic review of research on artificial intelligence applications in higher education – where are the educators? **International Journal of Educational Technology in Higher Education**, v. 16, n. 1, p. 1-27, 2019. Disponível em: https://link.springer.com/article/10.1186/s41239-019-0171-0. Acesso em: 26 nov. 2025.

ZHA0, Ruilin et al. O impacto do uso do ChatGPT no aprimoramento do engajamento e dos resultados de aprendizagem dos alunos no ensino superior: uma revisão. **International Journal of Academic Research in Business and Social Sciences**, v. 13, n. 12, p. 3734-3744, 2023. Disponível em: https://kwpublications.com/papers_submitted/7960/. Acesso em: 20 nov. 2025.

APÊNDICE A - MATERIAIS, PROTOCOLO E EVIDÊNCIAS DO ESTUDO DE CASO COMPARATIVO

Este apêndice reúne os materiais necessários à reprodução e à auditoria do estudo de caso comparativo sobre a qualidade de respostas geradas por IAs generativas em questões de Engenharia de Software. Todos os artefatos citados (prompts integrais, respostas brutas das ferramentas, código-fonte e logs de execução) compõem o pacote de replicação descrito na pesquisa.

A. Procedência e protocolo de coleta

As respostas foram coletadas em sessões independentes, com enunciados idênticos para as duas ferramentas e sem interações de acompanham ento. Os metadados abaixo são parte essencial da reprodutibilidade, pois modelos de IA são atualizados com frequência.

Ferramenta	**Versão (conf. relatada)**	**Data/hora da coleta**	**Condições**
Claude	Sonnet 4.6 (esforço máximo)	31/05/2026, 16:10	Sessão limpa; prompt em português; sem follow-up
Codex	GPT-5.5 (esforço máximo)	31/05/2026, 16:10	Sessão limpa; prompt em português; sem follow-up

Observação de transparência: os rótulos de versão são os relatados no momento da coleta e devem ser reportados como tais; o estudo se refere a esses pontos específicos no tempo e não generaliza para versões futuras.

B. Instrumento — as dez questões

As questões foram distribuídas entre áreas de conhecimento da Engenharia de Software (inspiradas no SWEBOK), equilibrando questões conceituais e práticas (com produção de código) e variando a dificuldade. Os enunciados integrais, incluindo os trechos de código

fornecidos nas questões 4 a 9, estão no pacote de replicação (pasta instrument/).

Q	**Área**	**Foco**	**Tipo**	**Dificuldade**
Q1	Requisitos	RF vs. RNF	Conceitual	Fácil
Q2	Arquitetura	Microsserviços vs. monólito	Conceitual	Média
Q3	Padrões de projeto	Padrão Strategy	Prática	Média
Q4	Testes	Testes unitários	Prática	Média
Q5	Algoritmos	Bug + otimização $O(n^2) \rightarrow O(n)$	Prática	Difícil
Q6	Qualidade	Code smells + SOLID	Prática	Média
Q7	Versionamento	Git Flow + hotfix	Prática	Fácil-Média
Q8	Segurança	Vulnerabilidades + correção	Prática	Difícil
Q9	Banco de dados	Normalização 3FN + query	Prática	Média
Q10	Processo	Dívida técnica	Conceitual	Média

C. Rubrica de avaliação e taxonomia de erros

Cada resposta foi pontuada em três dimensões, em escala de 1 a 5. Os erros foram registrados separadamente, por tipo e severidade, para evitar dupla contagem entre 'precisão' e 'erros'.

Dimensão	**1 (Muito ruim)**	**3 (Adequada)**	**5 (Excelente)**
Precisão	Majoritariamente incorreta	Correta no essencial, imprecisões menores	Tecnicamente correta e atualizada
Clareza	Confusa, mal estruturada	Compreensível, organização razoável	Clara, bem estruturada, didática
Profundidade	Superficial	Cobre o principal, alguns trade-offs	Trade-offs, casos de borda, justificativas e alternativas

Taxonomia de erros:

- Factual/conceitual — afirmação tecnicamente incorreta.
- Código — não compila, não roda ou produz resultado errado.
- Omissão — falta de informação essencial.
- Alucinação — API, biblioteca, comando ou fato inexistente.
- Desatualização — informação obsoleta.
- Inconsistência — contradição interna.

Severidade: Baixa (não compromete) · Média (compromete em parte) · Alta (compromete o uso da resposta).

D. Procedimento de avaliação

- Gabarito de referência: para cada questão foi elaborada uma resposta-referência baseada em fontes consolidadas, usada como âncora para a dimensão de precisão.
- Avaliação cega: as respostas foram identificadas como 'A/B' durante a pontuação, sem indicação da ferramenta de origem.
- Verificação objetiva de código (questões 3 a 9): em vez de julgamento subjetivo, o código foi executado e comparado à saída esperada (Seção F).

E. Pontuação por questão, com justificativa

Escala 1–5 por dimensão (Pre = Precisão; Cla = Clareza; Pro = Profundidade). Nenhum erro foi identificado em nenhuma das respostas.

Claude Sonnet 4.6:

Q	IA	Pre	Cla	Pro	Justificativa
Q1	Claude	5	5	5	Definições corretas; exemplos mensuráveis (P95, TLS 1.3, PCI-DSS); distingue a verificabilidade de RF/RNF.
Q2	Claude	5	5	5	Tabela comparativa ampla; cita Conway's Law, modular monolith, strangler fig, Saga.
Q3	Claude	5	5	5	Strategy correto; value object e método 'cheapest'; explica OCP e DIP. Código executa (Seção F).
Q4	Claude	5	5	5	Validador de CPF e suíte pytest; 24/24 testes passam. Partições + valores-limite + branch/mutation coverage.
Q5	Claude	5	5	5	Bug (i==j) identificado com exemplo; correção $O(n^2)$ e versão O(n); execução confirma [0,0]→[1,2].
Q6	Claude	5	5	5	Refatoração cobre os 5 princípios SOLID; Protocol (ISP/DIP), Factory (OCP), injeção de dependências; executa.
Q7	Claude	5	5	4	Git Flow e hotfix corretos e detalhados. Profundidade 4: não cita alternativas modernas (trunk-based/GitHub Flow).
Q8	Claude	5	5	5	SQL injection + XSS, com prepared statements, ORM, textContent/DOMPurify, CSP, cookies HTTPOnly.
Q9	Claude	5	5	5	Normalização 1FN→3FN com tabela Categorias; DDL com FKs/CHECK; query correta (Seção F).
Q10	Claude	5	5	5	Conceito (Cunningham), causas, métricas (debt ratio, complexidade), ferramentas, estratégias ágeis e quadrante de hot spots.

Codex (GPT-5.5):

Q	IA	Pre	Cla	Pro	Justificativa
Q1	Codex	5	5	4	RF/RNF corretos, RNF mensurável (P95, HTTPS). Profundidade 4: pouca elaboração; sem discutir verificabilidade/trade-offs.
Q2	Codex	5	5	4	Comparativo correto e bom critério de quando usar. Profundidade 4: sem modular monolith, Conway's Law ou Saga.
Q3	Codex	5	5	4	Strategy correto (TypeScript); explica OCP; código executa. Profundidade 4: sem DIP nem recursos extras.
Q4	Codex	5	5	3	Função e testes Jest corretos; 7/7 asserções passam. Profundidade 3: justificativa da estratégia de cobertura muito breve.
Q5	Codex	5	5	4	Bug (j=i) identificado; correção $O(n^2)$ e versão O(n) com Set; execução confirma. Conciso.
Q6	Codex	5	5	4	Refatora com Strategy + injeção + interfaces; cobre SRP/OCP/DIP; executa. Não aborda LSP/ISP.
Q7	Codex	5	5	3	Fluxo de hotfix correto. Profundidade 3: não trata a feature em andamento nem cita alternativas modernas.
Q8	Codex	5	5	4	SQL injection corrigido; bom insight ao trocar senha por password_hash (bcrypt/Argon2), menor privilégio, logs. Sem XSS/CSP.
Q9	Codex	5	5	4	Normalização 1FN→3FN e query corretas (executa). Profundidade 4: sem DDL com constraints explícitas.
Q10	Codex	5	5	4	Causas, métricas e estratégias corretas, com a nuance de dívida como escolha consciente. Menos específico em ferramentas/ADRs.

F. Evidência objetiva de execução de código (questões 3–9)

Para reduzir a subjetividade na dimensão de precisão, o código produzido por ambas as ferramentas foi executado em ambiente controlado (Python 3.12, Node.js 22, pytest 9, TypeScript via tsc; SQL em SQLite). O quadro abaixo resume a verificação; os logs completos seguem em sequência e estão no pacote de replicação (pasta execution/).

Q	**O que foi executado**	**Claude**	**Codex**
Q3	Cálculo de frete (Strategy)	OK — saídas coerentes	OK — frete=25
Q4	Suíte de testes do validador	24/24 testes OK	7/7 asserções OK
Q5	Bug vs. versão otimizada	Bug confirmado; O(n) OK	Bug confirmado; O(n) OK
Q6	Refatoração SOLID (executa)	OK — descontos 10/7/5%	OK — injeção funciona
Q9	Query de total por cliente	OK — 599,80 / 359,60	OK — 599,80 / 359,60

F.1 — Log de execução: Claude Sonnet 4.6

```
===== CLAUDE Q3 (Strategy) =====
Correios PAC: R$ 62.50
Jadlog .Package: R$ 57.00
Total Express: R$ 19.75
Melhor: Total Express - R$ 19.75

===== CLAUDE Q4 (pytest CPF) =====
........................                                                [100%]
24 passed in 0.02s

===== CLAUDE Q5 (two_sum bug vs otimizado) =====
buggy([3,2,4],6) = [0, 0] <- usa indice 0 duas vezes (BUG)
optimized([3,2,4],6) = [1, 2] <- correto
===== CLAUDE Q6 (SOLID refatorado, com injecao) =====
  [repo] salvo ped-VIP
  [email] enviado ped-VIP
  [report] ped-VIP -> R$ 180.00
VIP: final = R$ 180.00
  [repo] salvo ped-PREMIUM
  [email] enviado ped-PREMIUM
  [report] ped-PREMIUM -> R$ 186.00
PREMIUM: final = R$ 186.00
  [repo] salvo ped-COMUM
  [email] enviado ped-COMUM
  [report] ped-COMUM -> R$ 190.00
COMUM: final = R$ 190.00

===== CLAUDE Q9 (normalizacao 3FN + query) =====
  ('Bia Souza', 1, 599.8)
  ('Ana Lima', 1, 359.59999999999997)
```

F.2 — Log de execução: Codex (GPT-5.5)

```
===== CODEX Q3 (Strategy) =====
Correios frete(5kg,100km) = 25

===== CODEX Q5 (hasDuplicate bug vs otimizado) =====
buggy([1,2,3]) = true <- deveria ser false (BUG: j=i)
otimizado([1,2,3]) = false <- correto
otimizado([1,2,2]) = true <- correto (true)

===== CODEX Q6 (SOLID refatorado, com injecao) =====
  [pix] R$ 150
  [repo] salvo
  [email] enviado
OK: finalizarPedido executado com injecao de dependencias

===== CODEX Q4 (testes de e-mail, asserções literais) =====
Resultado: 7 passaram, 0 falharam (de 7 casos)

===== CODEX Q9 (normalizacao 3FN + query) =====
  (2, 'Bia', 599.8)
  (1, 'Ana', 359.59999999999997)
```

G. Resultados agregados

Dimensão	Claude	Codex	Diferença (C−Cx)
Precisão	5,00	5,00	0,00
Clareza	5,00	5,00	0,00
Profundidade	4,90	3,80	+1,10
Média geral	4,97	4,60	+0,37
Total de erros	0	0	—

Profundidade por tipo de questão:

Tipo	Claude	Codex
Conceitual (Q1, Q2, Q10)	5,00	4,00
Prática (Q3–Q9)	4,86	3,71

Síntese: as ferramentas empataram em precisão e clareza — nenhuma cometeu erro factual, de código ou alucinação, conforme verificado por execução. A diferença concentrou-se na profundidade e foi mais acentuada nas questões práticas, sugerindo, dentro dos limites do estudo, maior tendência do Claude a explorar trade-offs e alternativas, e do Codex a respostas corretas e mais concisas.

H. Ameaças à validade

Validade de construto.

A 'qualidade' foi operacionalizada em três dimensões; outros aspectos relevantes (custo, latência, criatividade, segurança por padrão) não foram medidos. A rubrica foi ancorada na literatura, mas a dimensão de profundidade permanece a mais subjetiva.

Validade interna.

Avaliação cega, gabarito de referência e verificação objetiva por execução de código.

Validade externa.

Apenas dez questões, duas ferramentas e um único ponto no tempo. Modelos de IA são atualizados continuamente; os resultados valem para as versões e a data declaradas na Seção A e não devem ser generalizados.

Confiabilidade.

Todo o protocolo, os prompts, as respostas brutas, o código e os logs estão disponíveis no pacote de replicação, permitindo que terceiros reproduzam a coleta, reexecutem o código e contestem ou confirmem as notas atribuídas.

I. Pacote de replicação

Recomenda-se depositar os materiais em repositório público com identificador persistente (por exemplo, GitHub vinculado ao Zenodo, gerando DOI e carimbo de data), citado no corpo do artigo como material suplementar. Estrutura sugerida:

```
replication-package/
├── README.md            (visão geral, versões, como reproduzir)
├── PROTOCOL.md          (protocolo e pré-registro do método)
├── instrument/          (enunciados integrais + trechos de código Q4-Q9)
├── rubric/              (rubrica 1-5 e taxonomia de erros)
├── reference-answers/   (gabarito de referência)
├── responses/
│   ├── claude/          (respostas brutas, por questão)
│   └── codex/           (respostas brutas, por questão)
├── provenance/          (versões, datas, screenshots/PDF das sessões)
├── execution/           (código executável + logs de saída)
├── scoring/             (planilha de avaliação com justificativas)
└── results/             (tabelas e figuras agregadas)
```

Ambiente de execução utilizado: Python 3.12.3, Node.js 22, pytest 9.0.3, TypeScript (tsc) e SQLite (via biblioteca padrão do Python). Comandos de reprodução documentados no README.