

Laudo Oficial de Correção — Prova Teórica 1

Gestão de Projetos de Software (GPS) ▪ CEFET-MG 2026.2

Prof. Aléssio Miranda Júnior

2026-10-19

i Dados do Discente

- **Discente:** André Guilherme Rodrigues de Souza
- **Matrícula:** 20243000982
- **Data da Avaliação:** 19/10/2026
- **Pontuação Final:** 16,4 / 25,0 (Aproveitamento: 65,6%)
- **Legibilidade da Caligrafia:** 95% (Alta)

Desempenho Consolidado por Questão

Questão / Tema	Pontos Obtidos	Valor	Aproveitamento
Q1: Projeto, Ciclo de Vida e Trade-offs	4,7	5,0	94,0%
Q2: Escopo, EAP e Backlog	3,0	5,0	60,0%
Q3: Qualidade, CoQ e Definição de Pronto	3,4	5,0	68,0%
Q4: Riscos, Segurança de IA e Documentação	3,3	5,0	66,0%
Q5: Comunicação, Stakeholders e Conflitos	2,0	5,0	40,0%
TOTAL GERAL	16,4	25,0	65,6%

Questão 1 — Projeto, Ciclo de Vida e Trade-offs (4,7 / 5,0 pts)

1.1 Classificação como Projeto vs. Operação e Evidências (1,5 / 1,5)

Resposta Transcrita do Aluno

“O professor agente é um projeto porque ele é algo que possui um objetivo final / finalização, não é algo que será trabalhado por um período indeterminado e tem um custo fixo para a sua criação, ao contrário de uma operação.”

Parecer do Docente

Excelente. Identificou com precisão a natureza de projeto e justificou mobilizando três características fundamentais do Capítulo 1: objetivo final delimitado (resultado único), temporalidade (não é contínuo/indeterminado) e custos/recursos específicos de criação.

1.2 Alternativas de Trade-off no Triângulo de Ferro (1,8 / 2,0)

Resposta Transcrita do Aluno

“Sem sacrificar a qualidade, podemos diminuir o escopo do projeto e aumentar o tempo, que faria um projeto com menos features mas o aumento no tempo garantiria a qualidade, e também podemos diminuir o tempo e aumentar o custo, que significa fazer o projeto com a mesma qualidade, menos tempo mas maior custo.”

Parecer do Docente

Muito bom raciocínio. Propôs duas soluções viáveis preservando rigorosamente a qualidade como restrição intransigível: (1) redução de escopo com dilatação de prazo e (2) compressão de prazo via aumento de custos. Pequeno desconto de 0,2 pela descrição simplificada de como o custo seria injetado operacionalmente.

1.3 Classificação do Ciclo de Vida (1,4 / 1,5)

Resposta Transcrita do Aluno

“O ciclo de vida adotado foi criado pensando que cada aluno da equipe teria 4h por semana para o projeto, 8h por sprint e 40h no total em 5 sprints, a abordagem escolhida foi a híbrida, com a criação de todas as issues no começo do semestre mas com alterações sendo possíveis conforme o progresso do semestre.”

Parecer do Docente

Classificação válida e bem sustentada. Embora a cadência central seja ágil (Sprints quinzenais), o aluno defendeu legitimamente a presença de traços preditivos/híbridos (planejamento inicial das issues conjugado a ajustes adaptativos contínuos com cadência fixa de esforço).

Questão 2 — Escopo, EAP e Backlog (3,0 / 5,0 pts)

2.1 Estrutura Analítica do Projeto (EAP) e Regra dos 100% (0,8 / 1,5)

Resposta Transcrita do Aluno

Diagrama: Raiz [Professor Agente] com ramificações: [Integração com a plataforma], [Testes Socráticos], [Implementação no servidor]. Texto: “A EAP é hierárquica porque ela define objetivos compostos por partes menores de menor significância, mas quando se sobe na árvore a significância aumenta até chegar no projeto em si, criando uma hierarquia.”

Parecer do Docente

Estrutura montada, porém os pacotes foram nomeados com ações/testes em vez de entregáveis puros baseados em substantivos (ex: Módulo de Chat, Camada de Autenticação, Pipeline). Explicou a hierarquia, mas omitiu a definição formal da Regra dos 100% (cobertura total e exclusiva do escopo pai).

2.2 Detalhamento de Issues e Estimativa Relativa (1,4 / 1,5)

Resposta Transcrita do Aluno

Issue 1: “Teste de conexão com a API inicial” (Desc: Conectar API do LLM com a plataforma; Critério: Ser possível ter uma conversa com a LLM a partir da plataforma; Est: 4h). Issue 2: “Testes contra Prompt Injection” (Est: 8h). Justificativa: “Como prompt injection é algo mais difícil e que leva mais tempo pra tratar do que a outra issue, ela recebe mais tempo.”

Parecer do Docente

Issue 1 muito bem especificada com critérios verificáveis de aceite e estimativa em horas. Issue 2 com estimativa comparativa coerente baseada na incerteza/complexidade técnica de segurança de IA.

2.3 Milestones vs. Sprints no Modelo Ágil / Kanban (0,8 / 2,0)

Resposta Transcrita do Aluno

“Milestones não substituem sprints por serem pacotes de trabalho mais longos, que raramente vão ser possíveis de serem feitos em 1 sprint, Kanban sem sprints pode ser adequado em um projeto onde todos os requisitos já foram levantados e não será necessário mudanças futuras, com todas as sprints pré-definidas.”

Parecer do Docente

Identificou que Milestones e Sprints possuem naturezas distintas (0,5), mas equivocou-se gravemente sobre o Kanban: Kanban sem sprints é adequado para fluxo contínuo e demandas imprevisíveis/manutenção, e não para cenários com requisitos congelados em cascata (0,3).

Questão 3 — Qualidade, CoQ e Definição de Pronto (3,4 / 5,0 pts)

3.1 Definição de Pronto (DoD) e Relação com CI/CD (1,3 / 1,5)

Resposta Transcrita do Aluno

“Definição de pronto é como o projeto considera o que é uma feature completa e integrada / o que é necessário para ir para a próxima issue, caso os testes de segurança sejam omitidos, a feature feita não vai estar pronta, que nesse caso seria o chat.”

Parecer do Docente

Compreensão correta e segura da DoD como acordo contratual de qualidade da equipe, concluindo com precisão que a omissão de segurança impede que a entrega seja declarada “pronta”.

3.2 Classificação PAF dos Custos da Qualidade (CoQ) (0,7 / 1,5)

Resposta Transcrita do Aluno

“(c) é o mais caro porque um erro de segurança encontrado por um aluno pode gerar processos com custos exorbitantes, seguido por (d) que impede o usuário de usar o software completamente, (a) que gasta tempo de outros devs e (b) que gasta apenas o tempo de 1 dev.”

Parecer do Docente

Acertou com ótima justificativa o custo mais alto (falha externa em produção gerando passivo legal/reputacional - 0,5). No entanto, não utilizou os termos técnicos formais de CoQ solicitados para os 4 itens: Prevenção (a), Avaliação (b), Falha Externa (c) e Falha Interna (d).

3.3 Heurística 1:10:100 e Particularidades de Projetos de IA (1,4 / 2,0)

Resposta Transcrita do Aluno

“A regra 1:10:100 diz que o custo é 1 quando o erro é encontrado pelo dev, 10 quando é encontrado na hora do deploy e 100 quando o usuário encontra um bug no sistema, em softwares que usam IA, colocar testes de segurança para evitar que um usuário gaste tokens à vontade ou quebre as regras da IA para ter acesso irrestrito a API pode salvar muito dinheiro, bem mais do que foi gasto implementando os testes de segurança.”

Parecer do Docente

Explicou a progressão geométrica da heurística e conectou ao benefício econômico de blindar o consumo de tokens na IA (1,4). Deixou de pontuar nos limites específicos da heurística em IA (não-determinismo e imprevisibilidade dos custos de alucinação).

Questão 4 — Riscos, Segurança de IA e Documentação (3,3 / 5,0 pts)

4.1 Risco vs. Problema e Posicionamento na Matriz P×I (1,0 / 1,5)

Resposta Transcrita do Aluno

Posicionamento: Impacto Alto e Probabilidade Baixa (quadrante inferior direito).
Texto: “Esse evento era um risco que se tornou um problema por ter apenas uma chance de acontecer e era baseado na responsabilidade de um dev. O risco tinha uma probabilidade bem baixa de acontecer, por ser algo facilmente evitável, mas tem um impacto enorme gerando possíveis custos grandes.”

Parecer do Docente

Diferenciou perfeitamente incerteza futura (risco) de fato consumado com impacto (problema - 0,5). Posicionou na matriz justificando o impacto (0,5). Faltou nomear a estratégia original de resposta planejada (mitigação/prevenção).

4.2 Ações Imediatas e Prevenção Permanente de Chaves de API (1,1 / 1,5)

Resposta Transcrita do Aluno

“Desativar aquela chave, deletar o commit e atualizar o histórico, clonar o repositório em um novo sem o leak. Uma ação preventiva seria, além de adicionar no .gitignore o arquivo com as chaves, efetuar uma rotação constante das chaves de API.”

Parecer do Docente

Excelente foco na ação imediata mais crítica: desativar/revogar a credencial exposta. Como prevenção permanente, citou e rotação regular. Faltou citar verificação automatizada via Secret Detection no GitLab CI.

4.3 Modelagem de Riscos de IA e Artefatos de Documentação (ADRs) (1,2 / 2,0)

Resposta Transcrita do Aluno

Risco de entrada: “Uso exacerbado de Tokens” (Causa: Falta de limitação; P: Baixa; I: Alto; Mitigação: Rate limit; Contingência: Limite por IP). [Risco de saída: Omitido].
Documentação: “Um ADR seria a melhor escolha, por ser feito justamente para documentar issues.”

Parecer do Docente

Risco de entrada completo e bem mitigado (0,75). Recomendou corretamente o uso de ADR (0,45). Omitiu completamente a resposta para a mitigação de risco de saída (ex: alucinação pedagógica, parametrização de temperatura ou testes de regressão - perdeu 0,5).

Questão 5 — Comunicação, Stakeholders e Conflitos (2,0 / 5,0 pts)

5.1 Comunicação Síncrona vs. Assíncrona no GitLab (0,8 / 1,5)

Resposta Transcrita do Aluno

“Uma comunicação síncrona permite uma discussão mais fluida, onde é fácil descobrir problemas e discutir ideias, portanto documentar essas conversas é uma tarefa difícil, exigindo muitas vezes alguém na reunião ficar apenas por conta disso. Já a assíncrona a documentação já vem pronta, que é o histórico, mas perde na fluidez e facilidade de discutir ideias; o project charter é aonde é documentado as decisões técnicas do projeto, por ser feito no início e algo que é difícil de ser modificado.”

Parecer do Docente

Excelente análise dos prós e contras da comunicação síncrona versus assíncrona (0,7). No entanto, indicou erroneamente o Project Charter como veículo de registro de decisões técnicas contínuas (o Charter é termo de autorização inicial; para decisões técnicas usa-se ADR, Issues ou MRs).

5.2 Matriz Poder × Interesse e Estratégia de Comunicação (0,7 / 1,5)

Resposta Transcrita do Aluno

Matriz: Provedor em ‘Manter satisfeito’; Professor e Equipe em ‘Gerenciar de perto’; Alunos em ‘Monitorar’; Coordenação em ‘Manter informado’. Texto: “Para essa comunicação específica dos dois stakeholders mais críticos, que é o professor e os alunos, a própria aula semanal sobre a matéria seria o momento perfeito para se discutir sobre o projeto, dado que o conteúdo da aula é a matéria de GPS onde discutimos esse tipo de coisa.”

Parecer do Docente

Posicionamento impreciso na matriz: os Alunos (usuários finais) possuem alto interesse e baixo poder (Manter informado), enquanto o Provedor possui baixo interesse e poder reduzido de intervenção direta no produto acadêmico. A estratégia proposta de usar a aula é informal e genérica.

5.3 Modos de Resolução de Conflito de Thomas-Kilmann (0,5 / 2,0)

Resposta Transcrita do Aluno

“Uma abordagem escolhida seria a de respostas curtas, porque além de salvar tokens que é economizar, algo que seria gasto em excesso em respostas mais longas, a chance do modelo alucinar é bem mais baixo, por não ter que criar uma frase de resposta longa para algo que é simples e objetivo, e por causa da urgência do projeto, tratar esses riscos já aumentaria o custo do projeto sem aumentar a qualidade, e ainda adicionaria riscos maiores ao projeto, como a alucinação.”

Parecer do Docente

O aluno defendeu apenas o argumento técnico das respostas curtas. Não utilizou o modelo teórico de Thomas-Kilmann (Colaborar, Forçar, Conceder, Evitar, Acomodar) e não explicou por que duas outras posturas seriam inadequadas ao caso pedagógico. Recebe 0,5 apenas pelos argumentos válidos sobre trade-off de tokens/latência.

*Lauda gerado oficialmente pela disciplina Gestão de Projetos de Software (GPS) • CEFET-MG
Timóteo*