

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:** Mateus Silva Lopes
- **Matrícula:** 20243004300
- **Data da Avaliação:** 19/10/2026
- **Pontuação Final:** 18,7 / 25,0 (Aproveitamento: 74,8%)
- **Legibilidade da Caligrafia:** 88% (Média)

Desempenho Consolidado por Questão

Questão / Tema	Pontos Obtidos	Valor	Aproveitamento
Q1: Projeto, Ciclo de Vida e Trade-offs	4,2	5,0	84,0%
Q2: Escopo, EAP e Backlog	2,9	5,0	58,0%
Q3: Qualidade, CoQ e Definição de Pronto	3,7	5,0	74,0%
Q4: Riscos, Segurança de IA e Documentação	4,5	5,0	90,0%
Q5: Comunicação, Stakeholders e Conflitos	3,4	5,0	68,0%
TOTAL GERAL	18,7	25,0	74,8%

Questão 1 — Projeto, Ciclo de Vida e Trade-offs (4,2 / 5,0 pts)

1.1 Classificação como Projeto vs. Operação e Evidências (1,2 / 1,5)

Resposta Transcrita do Aluno

“É um projeto, pois apresenta características do mesmo, como utilização de método ágil, possuir seu escopo bem definido e possuir tempo para executar (início + meio + fim).”

Parecer do Docente

Classificou corretamente como Projeto . Apontou a temporalidade (início, meio e fim) e a delimitação de escopo. “Utilização de método ágil” é método de gestão, não característica ontológica de projeto (faltou citar recursos finitos ou produto exclusivo). Pontuação: 1,2 / 1,5.

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

Resposta Transcrita do Aluno

“Se o tempo foi impactado, existem duas alternativas, diminuir o escopo reduzindo a quantidade de funcionalidades, ou aumentar o custo para por exemplo aumentar a equipe e manter o escopo e o tempo.”

Parecer do Docente

Excelente! Formulou com precisão as duas alternativas do Triângulo de Ferro sem comprometer a qualidade: (1) redução de escopo e (2) injeção de recursos/custo para acelerar a equipe e preservar o cronograma. Pontuação integral: 2,0 / 2,0.

1.3 Classificação do Ciclo de Vida (1,0 / 1,5)

Resposta Transcrita do Aluno

“O ciclo de vida de sprints de duas semanas é uma característica de metodologia ágil.”

Parecer do Docente

Classificou acertadamente como Ágil com base nas sprints quinzenais. A justificativa foi excessivamente sucinta, deixando de articular a adaptação contínua e as incertezas tecnológicas do modelo de IA. Pontuação: 1,0 / 1,5.

Questão 2 — Escopo, EAP e Backlog (2,9 / 5,0 pts)

2.1 Estrutura Analítica do Projeto (EAP) e Regra dos 100% (0,6 / 1,5)

Resposta Transcrita do Aluno

[Diagrama com lista funcional 1. Autenticação, 2. Uso LLM, 3. Registro, 4. Dashboard, 5. Objetivos SMART e esquema de mensagens Pergunta/Servidor] “Os 3 pacotes de trabalho são Front para interface, back para tratamento e conexão com LLM e banco, e banco para armazenar as conversas. EAP é hierarquica pois uma parte depende da outra.”

Parecer do Docente

Identificou pacotes de trabalho (front, back, banco), mas não desenhou a árvore hierárquica por entregas (desenhou lista/fluxo de comunicação). Além disso, confundiu a Regra dos 100% com dependência de cronograma (“uma parte depende da outra”). Pontuação: 0,6 / 1,5.

2.2 Detalhamento de Issues e Estimativa Relativa (0,5 / 1,5)

Resposta Transcrita do Aluno

“Título: Interface do chat, descrição: desenvolver a interface do chat com a entrada e saída. Critério: interface facilitada e acessível. Estimativa: usando $(O + 4M + P)/6$ para definir o tempo, $O=1$ semana, $M=2$ semanas, $P=4$ semanas, $(1 + 8 + 4)/6$ 2,16 semanas. A partir dessa issue deve-se testar se o back end está se comunicando bem com a interface.”

Parecer do Docente

Apresentou a Issue 1 com cálculo engenhoso da fórmula PERT tri-ponto. Contudo, omitiu a segunda issue e a justificativa comparativa de esforço relativo solicitadas pelo enunciado. Pontuação: 0,5 / 1,5.

2.3 Milestones vs. Sprints no Modelo Ágil / Kanban (1,8 / 2,0)

Resposta Transcrita do Aluno

“Milestone são mais completas que uma sprint, ou seja, carregam mais informações, uma milestone pode ter várias sprints, então nenhuma substitui a outra. Por outro lado deixar o Kanban puro sem sprints dificulta o processo de desenvolvimento pois além de existir issues que dependem de outras issues, permite que a equipe tenha muita liberdade de quando fazer, o que pode prejudicar a qualidade, pois se for usado as ultimas 3 aulas para um projeto de 15 aulas, claramente haverá problemas.”

Parecer do Docente

Excelente análise! Explicou que milestones agrupam entregas maiores e não substituem sprints, e alertou brilhantemente sobre o perigo de descompasso temporal se o Kanban for aplicado sem a cadência das sprints em ambiente acadêmico com prazos rígidos. Pontuação: 1,8 / 2,0.

Questão 3 — Qualidade, CoQ e Definição de Pronto (3,7 / 5,0 pts)

3.1 Definição de Pronto (DoD) e Relação com CI/CD (1,2 / 1,5)

Resposta Transcrita do Aluno

“A definição de pronto depende do contexto, o programa pode estar pronto e rodando no computador do dev mas quando ele envia para o git, ao realizar os testes no CI, ele falha e o CD não adiciona as alterações. Ou seja, ao omitir os testes de segurança, pode haver uma falsa premissa que o sistema está pronto. Por fim, a definição de pronto depende dos testes.”

Parecer do Docente

Muito bom raciocínio! Diferenciou o funcionamento isolado no ambiente do desenvolvedor da real conformidade validada no pipeline CI/CD, concluindo com precisão sobre o risco de uma falsa sensação de entrega caso testes sejam ignorados. Pontuação: 1,2 / 1,5.

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

Resposta Transcrita do Aluno

“a: custo de prevenção, b: avaliação, c: falha interna, d: falha externa. O maior custo neste caso é a falha externa, pois ela está prejudicando o usuário final, ela já parou de ser um risco, ela virou problema, e o pior deles neste caso, pois o chat ficou inviabilizado.”

Parecer do Docente

Classificou (a) e (b) corretamente. Inverteu (c) e (d) (bug encontrado pelo aluno em produção é Falha Externa e falha no deploy é Falha Interna). A justificativa da falha externa como a mais onerosa foi fundamentada com êxito. Pontuação: 1,0 / 1,5.

3.3 Heurística 1:10:100 e Particularidades de Projetos de IA (1,5 / 2,0)

Resposta Transcrita do Aluno

“A regra diz que o custo é 100 encontrar um erro na produção, 1000 encontrar nos testes e 100 quando encontrada pelo cliente. Ou seja é muito mais barato corrigir o erro na base, do que nas pontas. Entretanto, não se pode tratar como lei universal, pois na matriz de problema x impacto, existem problemas de baixo impacto ou baixa probabilidade, que podem ser mitigados ou aceitos, sem a necessidade de diversos testes para remove-lo.”

Parecer do Docente

Embora tenha se equivocado na numeração geométrica (1:10:100), captou perfeitamente o fundamento econômico e a relatividade da regra em matrizes de risco. Faltou abordar o fator específico do não-determinismo dos modelos de IA. Pontuação: 1,5 / 2,0.

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

4.1 Risco vs. Problema e Posicionamento na Matriz P×I (1,2 / 1,5)

Resposta Transcrita do Aluno

“Esse evento foi um problema, pois ele já aconteceu, mas antes de ocorrer isso era risco. O problema de vaziar a API da LLM, pode até encerrar um projeto, pois com ela não há barreiras, quem tem acesso pode esgotar todos os tokens em segundos, ou pior, gerar um gasto suficiente para quebrar a empresa.” [Posicionou X em Crítico: Alto × Alto].

Parecer do Docente

Excelente distinção entre risco prévio e problema consumado. Justificou de forma contundente o impacto financeiro catastrófico do vazamento de credenciais com cobrança por tokens. Pontuação: 1,2 / 1,5.

4.2 Ações Imediatas e Prevenção Permanente de Chaves de API (1,35 / 1,5)

Resposta Transcrita do Aluno

“A primeira e mais urgente ação é cancelar essa chave e criar uma nova, depois avaliar o dano financeiro e por fim decidir se isso inviabiliza o projeto ou se o projeto ficará limitado. Mas para evitar isso é só botar de regra naquele arquivo que ele não será commitado.”

Parecer do Docente

Ações imediatas muito bem estruturadas (revogação da chave, auditoria financeira e análise de viabilidade). Ação preventiva adequada (regra de exclusão de commits / .gitignore). Pontuação: 1,35 / 1,5.

4.3 Modelagem de Riscos de IA e Artefatos de Documentação (ADRs) (1,9 / 2,0)

Resposta Transcrita do Aluno

“Prompt Injection, causa: uso de frases para burlar as diretrizes, impacto: alto, pode ferir a segurança do projeto, para evitar que isso aconteça, o texto enviado deve passar por verificação antes de ser enviado a LLM, sendo a probabilidade de prompt injection alta. Qualquer sistema que use LLM terá tentativas de prompt injection, onde o

usuário poderia ter acesso ao código completo da questão. Tudo isso seria registrado no ADR.”

Parecer do Docente

Resposta completa e de alto nível! Modelou com precisão o risco de injeção de prompt (causa, P, I, verificação de entrada e segurança da saída) e apontou corretamente o ADR (Architecture Decision Record) como artefato de governança das decisões de mitigação. Pontuação: 1,9 / 2,0.

Questão 5 — Comunicação, Stakeholders e Conflitos (3,4 / 5,0 pts)

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

Resposta Transcrita do Aluno

“A comunicação síncrona é boa para discutir o andamento do projeto como a daily do SCRUM, mas péssima para decisões importantes do projeto, confiar apenas na memória da pessoa para lembrar tudo é arriscado. A assíncrona ela é mais rígida, não possibilitando esquecer ou fugir do escopo, mas ela é engessada, não possibilita resoluções rápidas. Utilizar um Kanban garante que o time tenha registro das issues, e que seja fácil rastrear o andamento de cada uma.”

Parecer do Docente

Ótimo contraste entre rituais síncronos (daily) e a memória técnica das decisões assíncronas no Kanban. Pontuação: 1,1 / 1,5.

5.2 Matriz Poder × Interesse e Estratégia de Comunicação (0,8 / 1,5)

Resposta Transcrita do Aluno

“Os alunos possuem o maior poder e interesse, pois são eles que vão desenvolver o programa, logo eles precisam de contato frequente, utilização de Kanban, e alto conteúdo. Por outro lado o professor tem baixo poder mas um interesse mais alto por ser o cliente.” [Diagrama: (4, 5) Manter satisfeito; (3) Gerenciar de perto; (2) Monitorar; (1) Manter informado]

Parecer do Docente

Houve inversão dos papéis: considerou os alunos como a equipe de desenvolvimento (3) e atribuiu baixo poder ao professor (2) que é o patrocinador/avaliador. A estratégia descrita reflete boas práticas de comunicação interna na equipe. Pontuação: 0,8 / 1,5.

5.3 Modos de Resolução de Conflito de Thomas-Kilmann (1,5 / 2,0)

Resposta Transcrita do Aluno

“Usaria mediação, tentando explicar o porquê, de maneira que pudesse talvez achar o meio termo. Não concederia, pois o risco de alucinação é real, sendo assim prejudicaria a qualidade, e nem usaria autoridade pois poderia afastar o cliente.”

Parecer do Docente

Muito perspicaz! Embora não tenha citado os nomes técnicos formais em inglês do modelo de Thomas-Kilmann, expressou perfeitamente os três modos: Conciliar / Meio-termo (escolhido com mediação), Acomodar (rejeitado para não conceder qualidade diante do risco de alucinação) e Forçar / Poder (rejeitado para preservar o relacionamento com o cliente). Pontuação: 1,5 / 2,0.

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