

## Questão 1 (3 pontos)

**Enunciado:** Compare e contraste "programar" com "fazer engenharia de software". Explique como a gestão de projetos se relaciona com a engenharia de software e qual o papel do gerente de projetos neste contexto. Utilize exemplos no projeto Portfólio ou prático do tema do seu grupo para ilustrar sua resposta.

### ✓ RESPOSTA POSITIVA (EXCELENTE - 3 pontos)

#### Programar:

- Atividade técnica de codificação
- Foco na implementação de funcionalidades específicas
- Escrever código seguindo lógica e sintaxe de uma linguagem
- Atividade pontual e específica

#### Fazer Engenharia de Software:

- Disciplina abrangente que inclui múltiplas atividades:
  - Análise de requisitos
  - Design e arquitetura
  - Implementação (codificação)
  - Testes e validação
  - Manutenção e evolução
  - Gestão de qualidade
  - Gestão de projeto

- Processo sistemático e disciplinado
- Considera todo o ciclo de vida do software

#### Relação com Gestão de Projetos:

- Gestão de Projetos organiza e coordena as atividades de Engenharia de Software
- Garante que o processo de desenvolvimento seja eficiente e eficaz
- Facilita a comunicação entre stakeholders
- Controla recursos, tempo, custo e qualidade

#### Papel do Gerente de Projetos:

- Planejamento e controle do projeto
- Comunicação entre stakeholders
- Alocação de recursos
- Gestão de riscos

- Mediação de conflitos
- Garantir que objetivos sejam alcançados

#### **Exemplo Prático - Projeto Portfólio:**

- **Programar:** Escrever código HTML/CSS para criar a página de perfil de um membro da equipe
- **Engenharia de Software:** Definir a estrutura do site, escolher tecnologia (Hugo), planejar arquitetura, definir padrões de código, planejar testes
- **Gestão de Projetos:** Organizar equipe, definir prazos, controlar entregas, gerenciar issues no GitLab, coordenar sprints

### **✗ RESPOSTA NEGATIVA (INSUFICIENTE - 0-1 ponto)**

#### **Erros Comuns:**

- Confundir programar com engenharia de software (tratá-los como sinônimos)
- Não diferenciar claramente os conceitos
- Não explicar a relação com gestão de projetos
- Papel do gerente de projetos muito vago ou incorreto
- Falta de exemplos práticos ou exemplos genéricos demais

#### **Exemplo de Resposta Incorreta:**

*"Programar é fazer engenharia de software. O gerente de projetos é quem programa. No projeto Portfólio, programamos o site."*

#### **Por que está errado:**

- Não diferencia os conceitos
- Papel do gerente de projetos incorreto
- Exemplo muito genérico e sem detalhamento

## Questão 2 (5 pontos)

**Enunciado:** Explique os tipos de viabilidade de um projeto de software e descreva critérios de viabilidade técnica, econômica e operacional para o projeto Portfólio ou prático do tema do seu grupo. Em sua resposta, justifique a importância de cada tipo de viabilidade e explique o que é um Project Charter, relacionando seus principais elementos ao processo de iniciação formal do projeto.

### ✓ RESPOSTA POSITIVA (EXCELENTE - 5 pontos)

#### Tipos de Viabilidade:

1. **Viabilidade Técnica:** Recursos técnicos disponíveis, tecnologias necessárias, competências da equipe, infraestrutura técnica
2. **Viabilidade Econômica:** Custos do projeto, benefícios esperados, ROI (Retorno sobre Investimento), recursos financeiros disponíveis
3. **Viabilidade Operacional:** Processos organizacionais, pessoas e capacidades, infraestrutura operacional, capacidade de operação e manutenção
4. **Outros tipos (opcional):** Legal, Temporal, Organizacional

#### Critérios para Projeto Portfólio:

**Viabilidade Técnica:** Conhecimento de HTML/CSS/JavaScript pela equipe, disponibilidade de Hugo e GitLab, competência da equipe em desenvolvimento web. **Justificativa:** Garantir que equipe possui habilidades técnicas necessárias para desenvolver o site.

**Viabilidade Econômica:** Custos de hospedagem (GitLab Pages é gratuito), tempo investido pela equipe (custo de oportunidade), benefício: Portfólio profissional, aprendizado, experiência. **Justificativa:** Validar que benefícios (aprendizado, portfólio) superam custos (tempo investido).

**Viabilidade Operacional:** Tempo disponível dos membros da equipe, infraestrutura (computadores, internet), processos de trabalho em equipe, capacidade de manter e atualizar o site. **Justificativa:** Garantir que projeto pode ser executado na prática com recursos disponíveis.

#### Project Charter:

**Definição:** Documento que autoriza formalmente o início do projeto, define objetivos, escopo inicial e recursos, define autoridade do gerente de projetos, estabelece contexto e limites do projeto.

**Elementos Principais:** Objetivos e justificativa do projeto, escopo inicial (alto nível), stakeholders principais, recursos necessários, riscos iniciais identificados, critérios de sucesso, restrições e premissas, orçamento inicial (se aplicável), autoridade do gerente de projetos.

**Relação com Processo de Iniciação:** Primeiro documento formal do projeto, autoriza início do trabalho, base para planejamento detalhado, comunica projeto aos stakeholders, estabelece contexto e limites, formaliza compromisso com o projeto.

## ✗ RESPOSTA NEGATIVA (INSUFICIENTE - 0-2 pontos)

### Erros Comuns:

- Listar apenas 1-2 tipos de viabilidade
- Não fornecer exemplos práticos do Portfólio
- Exemplos genéricos sem conexão com o projeto
- Não justificar importância dos tipos de viabilidade
- Project Charter definido incorretamente ou de forma muito vaga
- Elementos do Project Charter incompletos (menos de 3 itens)
- Não relacionar Project Charter com processo de iniciação

### **Exemplo de Resposta Incorreta:**

*"Viabilidade técnica é saber programar. Viabilidade econômica é ter dinheiro. Project Charter é um documento que autoriza o projeto."*

**Por que está errado:** Definições muito vagas e superficiais, falta de exemplos práticos do Portfólio, não justifica importância, Project Charter sem elementos principais, não relaciona com processo de iniciação.

## Questão 4 (3 pontos)

**Enunciado:** Explique a diferença entre escopo do projeto e escopo do produto. Diferencie também requisitos funcionais de requisitos não funcionais, fornecendo dois exemplos de cada para o projeto Portfólio ou prático do tema do seu grupo.

### ✓ RESPOSTA POSITIVA (EXCELENTE - 3 pontos)

**Escopo do Projeto:** Trabalho necessário para entregar o produto. Inclui todas as atividades de gestão, planejamento, execução e controle.

Exemplo: Criar EAP, planejar sprints, desenvolver site, fazer testes, documentar.

**Escopo do Produto:** Características e funções do produto final. Foco nas funcionalidades e entregas do produto. Exemplo: Páginas do site, navegação, formulário de contato, perfis dos membros.

**Requisitos Funcionais (O QUE o sistema deve fazer):** Funcionalidades específicas do sistema. Exemplos para Projeto Portfólio:

1. Página principal com apresentação da equipe
2. Navegação entre páginas de perfil dos membros
3. Formulário de contato funcional
4. Exibição de projetos realizados

**Requisitos Não Funcionais (COMO o sistema deve funcionar):**

Características de qualidade e performance. Exemplos para Projeto Portfólio:

1. Carregamento em menos de 3 segundos
2. Design responsivo (mobile, tablet e desktop)
3. Compatibilidade com navegadores modernos (Chrome, Firefox, Edge)
4. Acessibilidade (WCAG básico)

### ✗ RESPOSTA NEGATIVA (INSUFICIENTE - 0-1 ponto)

**Erros Comuns:**

- Confundir escopo do projeto com escopo do produto
- Não diferenciar claramente requisitos funcionais e não funcionais
- Exemplos genéricos sem conexão com o Portfólio
- Menos de 2 exemplos de cada tipo
- Exemplos incorretos (ex: classificar requisito não funcional como funcional)

**Exemplo de Resposta Incorreta:**

*"Escopo do projeto é o que vai ser feito. Escopo do produto é o produto. Requisitos funcionais são os que funcionam. Requisitos não funcionais são os que não funcionam."*

**Por que está errado:** Definições muito vagas, não diferencia corretamente os conceitos, interpretação incorreta de requisitos não funcionais, falta de exemplos práticos.

## Questão 5 (3 pontos)

**Enunciado:** O que é uma EAP (Estrutura Analítica do Projeto)? Descreva os três níveis típicos de uma EAP e construa um exemplo simplificado para o projeto Portfólio ou prático do tema do seu grupo com pelo menos três níveis hierárquicos, incluindo as principais entregas do projeto.

### ✓ RESPOSTA POSITIVA (EXCELENTE - 3 pontos)

**Definição de EAP:** Decomposição hierárquica do trabalho do projeto. Organiza trabalho em entregas e componentes. Estrutura em níveis progressivos de detalhamento. Ferramenta fundamental para planejamento.

#### Três Níveis Típicos:

1. **Nível 1:** Projeto completo (visão macro)
2. **Nível 2:** Fases ou grandes entregas (visão intermediária)
3. **Nível 3:** Pacotes de trabalho ou componentes específicos (visão detalhada)

#### Exemplo Simplificado - Projeto Portfólio:

```
Nível 1: Projeto Portfólio de Equipe
|
├─ Nível 2: Planejamento e Setup
|   └─ Nível 3: Project Charter
|   └─ Nível 3: Estrutura GitLab (repositório, issues, milestones)
|   └─ Nível 3: Configuração Hugo (instalação, tema, estrutura)
|
├─ Nível 2: Desenvolvimento
|   └─ Nível 3: Página Inicial (homepage)
|   └─ Nível 3: Páginas de Perfil (perfil de cada membro)
|   └─ Nível 3: Formulário de Contato
|
└─ Nível 2: Deploy e Documentação
    └─ Nível 3: Configuração CI/CD (GitLab CI)
    └─ Nível 3: Deploy do Site (GitLab Pages)
    └─ Nível 3: Documentação Final (README, guias)
```

### ✗ RESPOSTA NEGATIVA (INSUFICIENTE - 0-1 ponto)

#### Erros Comuns:

- Definição de EAP incorreta ou muito vaga
- Não explicar os três níveis

- Exemplo sem hierarquia clara (menos de 3 níveis)
- Exemplo genérico sem conexão com o Portfólio
- Estrutura confusa ou sem entregas identificáveis

***Exemplo de Resposta Incorreta:***

*"EAP é uma lista de tarefas. Nível 1 é o projeto, nível 2 são as tarefas.  
Exemplo: Fazer site, programar, testar."*

***Por que está errado:*** *Definição muito simplificada, não explica adequadamente os níveis, exemplo sem hierarquia adequada, falta de detalhamento e entregas específicas.*

## Questão 6 (3 pontos)

**Enunciado:** Explique o conceito de "custo da qualidade" e diferencie custos de conformidade e de não conformidade, dando exemplos práticos de cada tipo para o projeto Portfólio ou prático do tema do seu grupo. Em seguida, descreva o processo de controle de qualidade em projetos de software e proponha um plano de controle para garantir que o site do Portfólio atenda aos requisitos funcionais e não funcionais, mencionando pelo menos três técnicas utilizadas no controle da qualidade. Comente também sobre a importância do investimento em ações preventivas.

### ✓ RESPOSTA POSITIVA (EXCELENTE - 3 pontos)

**Conceito de Custo da Qualidade:** Custo total associado à garantia de qualidade do produto. Inclui custos de prevenir problemas e custos de corrigir problemas. Investir em prevenção reduz custos totais a longo prazo. Filosofia: "Prevenir é mais barato que corrigir".

#### **Custos de Conformidade (Prevenção e Avaliação):**

- **Prevenção:** Planejamento de qualidade, treinamento, processos bem definidos, design cuidadoso
- **Avaliação:** Testes, inspeções, auditorias, revisões de código, validações

#### **Custos de Não Conformidade (Falhas):**

- **Falhas Internas:** Retrabalho, scrap, correções antes da entrega ao cliente
- **Falhas Externas:** Garantia, perda de clientes, retificações pós-entrega, danos à reputação

#### **Exemplos Práticos - Projeto Portfólio:**

**Custos de Conformidade:** Tempo para planejar estrutura do site (prevenção), revisões de código entre membros (avaliação), testes de funcionalidades antes do deploy (avaliação), estabelecer padrões de código (prevenção).

**Custos de Não Conformidade:** Retrabalho devido a bugs não detectados (falha interna), perda de credibilidade se site tiver problemas (falha externa), tempo para corrigir erros pós-deploy (falha externa), reescrita de código mal planejado (falha interna).

#### **Processo de Controle de Qualidade:**

1. **Medir:** Coletar dados de qualidade (bugs, tempo de carregamento, etc.)
2. **Comparar:** Analisar contra padrões/critérios estabelecidos
3. **Analisar:** Identificar causas de problemas
4. **Corrigir:** Implementar melhorias e correções

**Plano de Controle para Portfólio:** Definir critérios de aceitação claros para cada funcionalidade, estabelecer pontos de verificação (checkpoints) em cada sprint, criar lista de verificação (checklist) de requisitos, realizar testes sistemáticos antes de cada merge.

**Técnicas de Controle de Qualidade (mínimo 3):**

1. **Inspeção:** Revisão manual de código e design
2. **Testes:** Testes funcionais, de usabilidade, responsividade
3. **Peer Review:** Revisão entre pares antes de merge no GitLab
4. **Validação com Stakeholders:** Feedback dos usuários/testadores
5. **Métricas:** Medição de performance (tempo de carregamento, acessibilidade)
6. **Checklist:** Lista de verificação de requisitos funcionais e não funcionais

**Importância do Investimento em Ações Preventivas:** Previne problemas antes que ocorram, reduz custos totais a longo prazo, melhora qualidade do produto final, aumenta satisfação do cliente/usuário, evita retrabalho e atrasos, melhora reputação e credibilidade.

**✗ RESPOSTA NEGATIVA (INSUFICIENTE - 0-1 ponto)**

**Erros Comuns:**

- Não diferenciar custos de conformidade e não conformidade
- Exemplos genéricos sem conexão com o Portfólio
- Processo de controle de qualidade não explicado
- Plano de controle muito vago ou ausente
- Menos de 3 técnicas de controle mencionadas
- Não mencionar importância da prevenção
- Confundir controle de qualidade com garantia de qualidade

**Exemplo de Resposta Incorreta:**

*"Custo da qualidade é o dinheiro gasto. Controle de qualidade é testar. Técnicas: testar, revisar, verificar."*

**Por que está errado:** Definições muito vagas, não diferencia custos de conformidade e não conformidade, falta de exemplos práticos,

*técnicas muito genéricas sem detalhamento, não menciona processo ou plano de controle.*

## Questão 7 (3 pontos)

**Enunciado:** Explique o que é risco em projetos de software e diferencie risco negativo (ameaça) e risco positivo (oportunidade). Cite e descreva dois riscos reais relevantes para o projeto Portfólio ou prático do tema do seu grupo, classificando-os quanto à probabilidade e impacto (use termos como baixo, médio, alto). Em seguida, explique como a documentação pode ajudar na identificação, exposição e mitigação desses riscos durante o ciclo de vida do projeto, indicando exemplos concretos de documentos (como matriz de riscos, atas, histórico de mudanças ou plano de ação) que auxiliam nesse processo.

### ✓ RESPOSTA POSITIVA (EXCELENTE - 3 pontos)

**Conceito de Risco:** Evento ou condição incerta que pode afetar objetivos do projeto. Pode ser negativo (ameaça) ou positivo (oportunidade). Caracterizado por probabilidade e impacto. Requer identificação, análise e resposta.

**Risco Negativo (Ameaça):** Pode causar danos ao projeto (atrasos, custos, qualidade). Requer estratégias: Evitar, Transferir, Mitigar, Aceitar.

**Risco Positivo (Oportunidade):** Pode trazer benefícios ao projeto (aceleração, economia, melhorias). Requer estratégias: Explorar, Compartilhar, Melhorar, Aceitar.

#### Dois Riscos Reais para Projeto Portfólio:

##### Risco 1: Dificuldade da equipe em usar Hugo

- **Tipo:** Ameaça (negativo)
- **Probabilidade:** Alta
- **Impacto:** Alto
- **Classificação:** Crítico (Alta probabilidade + Alto impacto)
- **Estratégia:** Mitigar - Treinar equipe, fornecer documentação, tutoriais, apoio técnico

##### Risco 2: Conflitos de versão no Git

- **Tipo:** Ameaça (negativo)
- **Probabilidade:** Média
- **Impacto:** Médio
- **Classificação:** Médio (Média probabilidade + Médio impacto)
- **Estratégia:** Mitigar - Estabelecer práticas Git, branch protection, code review obrigatório

#### Como a Documentação Ajuda na Gestão de Riscos:

**Identificação:** Matriz de riscos documenta todos os riscos identificados, brainstorming documentado em atas de reunião, histórico de projetos anteriores (lições aprendidas).

**Exposição:** Matriz de riscos visualiza probabilidade e impacto, registro de riscos comunica riscos aos stakeholders, relatórios de status incluem riscos ativos.

**Mitigação:** Plano de ação documenta estratégias de resposta, atas de reunião registram decisões sobre riscos, histórico de mudanças mostra evolução dos riscos, documentação técnica ajuda prevenir riscos técnicos.

#### **Exemplos Concretos de Documentos:**

1. **Matriz de Riscos:** Tabela com riscos, probabilidade, impacto e estratégias
2. **Atas de Reunião:** Registro de discussões sobre riscos e decisões
3. **Histórico de Mudanças:** Rastreamento de mudanças que podem gerar novos riscos
4. **Plano de Ação:** Documento com ações específicas para mitigar riscos
5. **Registro de Riscos:** Lista detalhada de todos os riscos do projeto

### **✗ RESPOSTA NEGATIVA (INSUFICIENTE - 0-1 ponto)**

#### **Erros Comuns:**

- Não diferenciar risco negativo e positivo
- Riscos genéricos sem conexão com o Portfólio
- Classificação de probabilidade/impacto incorreta ou ausente
- Não explicar como documentação ajuda na gestão de riscos
- Exemplos de documentos muito vagos ou incorretos
- Menos de 2 riscos citados
- Riscos irreais ou não relevantes

#### **Exemplo de Resposta Incorreta:**

*"Risco é algo que pode dar errado. Risco negativo é ruim, positivo é bom. Exemplo: site não funcionar. Documentação ajuda porque documenta."*

**Por que está errado:** Definições muito vagas, não diferencia adequadamente riscos negativos e positivos, risco genérico sem classificação, explicação sobre documentação muito superficial, falta de exemplos concretos de documentos.

## Questão 8 (2 pontos)

**Enunciado:** Explique um desafio significativo que sua equipe enfrentou no desenvolvimento do Checkpoint 1 e outro no Checkpoint 2 do projeto Portfólio ou prático do tema do seu grupo. Detalhe qual foi o seu papel diante desse desafio em cada um dos checkpoints, descrevendo as ações, contribuições ou aprendizados adquiridos.

### ✓ RESPOSTA POSITIVA (EXCELENTE - 2 pontos)

**Estrutura da Resposta:** Desafio no Checkpoint 1: descrição clara e específica. Papel do aluno no Checkpoint 1: ações, contribuições, aprendizados. Desafio no Checkpoint 2: descrição clara e específica. Papel do aluno no Checkpoint 2: ações, contribuições, aprendizados.

#### Exemplo de Resposta Positiva:

**Checkpoint 1 - Desafio:** Configuração inicial do GitLab e criação da estrutura do projeto. Dificuldade em entender como organizar issues e milestones. Coordenação inicial da equipe para definir escopo.

**Meu Papel no Checkpoint 1:** Pesquisei documentação do GitLab e criei tutorial para equipe. Organizei reunião para alinhar escopo do projeto. Criei template de issues para padronizar trabalho. Aprendi importância de documentação e comunicação clara.

**Checkpoint 2 - Desafio:** Criação do documento de projeto (Project Charter) completo. Dificuldade em estimar prazos realistas. Gestão de riscos e identificação de dependências.

**Meu Papel no Checkpoint 2:** Liderou criação do Project Charter, organizando contribuições da equipe. Pesquisou técnicas de estimativa e aplicou no cronograma. Identificou riscos principais e criou matriz de riscos. Aprendi importância de planejamento detalhado e realismo nas estimativas.

### ✗ RESPOSTA NEGATIVA (INSUFICIENTE - 0-0,5 pontos)

#### Erros Comuns:

- Desafios muito genéricos ou vagos
- Não descrever papel pessoal do aluno
- Falta de ações, contribuições ou aprendizados
- Apenas um checkpoint mencionado
- Resposta muito curta ou superficial
- Desafios irreais ou não relacionados ao projeto

**Exemplo de Resposta Incorreta:**

*"No Checkpoint 1 foi difícil. No Checkpoint 2 também foi difícil. Eu ajudei."*

**Por que está errado:** *Desafios muito vagos, não descreve papel pessoal, falta de detalhamento sobre ações e aprendizados, resposta muito superficial.*

## Questão 9 (6 pontos)

**Enunciado: CASO PRÁTICO:** Você é o gerente de projetos do Portfólio de Equipe ou do projeto prático do tema de seu grupo. O projeto está na fase de execução quando você identifica que: o escopo começou a expandir sem controle (novos requisitos sendo adicionados informalmente), a equipe está atrasada em relação ao cronograma planejado, foram identificados bugs que estão afetando a qualidade do produto, há riscos de alguns membros não conseguirem finalizar suas partes no prazo.

**Com base nos conhecimentos das aulas 1 a 6, desenvolva um plano de ação integrado que:**

- **a)** Contenha o escopo usando técnicas de gestão de escopo (2 pontos)
- **b)** Ajuste o cronograma de forma realista (1 ponto)
- **c)** Implemente medidas de controle de qualidade (1 ponto)
- **d)** Trate os riscos identificados com estratégias apropriadas (1 ponto)
- **e)** Documente as ações tomadas (1 ponto)

### ✓ RESPOSTA POSITIVA (EXCELENTE - 6 pontos)

Esta questão avalia integração de conhecimentos. A resposta deve demonstrar compreensão sistêmica e capacidade de aplicação integrada.

#### **a) Conter o Escopo (2 pontos)**

##### **Processo de Controle de Escopo:**

- Revisar escopo original do projeto e comparar com situação atual
- Documentar todas as mudanças solicitadas informalmente
- Avaliar impacto de cada mudança (tempo, custo, recursos)
- Estabelecer processo formal de mudanças (Change Control Board)

##### **Técnicas de Gestão de Escopo:**

- **Freeze de escopo:** Congelar escopo para iterações específicas
- **Backlog de mudanças:** Criar backlog para novas solicitações (avaliação posterior)
- **Validação de critérios de aceitação:** Revisar e reforçar critérios originais
- **Comunicação clara:** Comunicar limites do escopo aos stakeholders
- **WBS revisão:** Revisar EAP para garantir que trabalho está alinhado
- **Documentação de mudanças:** Registrar todas as solicitações de mudança

**Exemplo Prático:** Criar issue no GitLab para cada nova solicitação de mudança. Avaliar impacto antes de aprovar. Manter backlog separado para "nice-to-have". Comunicar à equipe que escopo está congelado para Sprint atual.

## **b) Ajustar Cronograma (1 ponto)**

### **Reavaliação do Cronograma:**

- Analisar atividades atrasadas e identificar causas
- Reestimar durações de forma realista (considerando aprendizado)
- Identificar dependências críticas e gargalos
- Recalcular caminho crítico (se usando PERT/CPM)

### **Ajustes Possíveis:**

- Redefinir prioridades usando MoSCoW (Must, Should, Could, Won't)
- Reduzir escopo não essencial (focar em MVP - Minimum Viable Product)
- Aumentar recursos (se viável) - mais pessoas ou mais tempo
- Estender prazo final (com aprovação de stakeholders)
- Paralelizar atividades quando possível
- Remover funcionalidades de baixa prioridade

**Exemplo Prático:** Revisar milestones no GitLab e ajustar datas. Priorizar funcionalidades essenciais (Must have). Adiar funcionalidades opcionais (Could have) para sprint seguinte. Comunicar novo cronograma à equipe e stakeholders.

## **c) Implementar Controle de Qualidade (1 ponto)**

### **Medidas de Controle de Qualidade:**

- Implementar testes sistemáticos antes de cada merge
- Estabelecer code review obrigatório (todos os merges revisados)
- Criar checklist de qualidade para cada entrega
- Definir critérios de aceitação claros para cada funcionalidade
- Realizar inspeções regulares (daily checks)
- Implementar testes automatizados (se possível)
- Estabelecer ambiente de staging para testes antes de produção

**Exemplo Prático:** Configurar GitLab CI para rodar testes automáticos. Criar template de code review no GitLab. Definir que nenhum merge é aceito sem aprovação de 2 membros. Criar checklist: funcionalidade testada, responsivo verificado, sem erros de console.

## **d) Tratar Riscos Identificados (1 ponto)**

### **Risco: Membros não finalizarem no prazo**

### **Estratégia: MITIGAR**

### **Ações Específicas:**

- Redistribuir trabalho entre membros disponíveis

- Aumentar comunicação (daily standups, checkpoints frequentes)
- Estabelecer checkpoints semanais para acompanhar progresso
- Criar plano de contingência (backup de membros para tarefas críticas)
- Identificar tarefas críticas e garantir que têm backup
- Oferecer suporte adicional para membros com dificuldades

**Reavaliação de Riscos:** Atualizar registro de riscos com novas probabilidades e impactos. Identificar novos riscos decorrentes da situação atual. Revisar estratégias de resposta para riscos existentes. Documentar riscos que se materializaram e ações tomadas.

**Exemplo Prático:** Atualizar matriz de riscos no GitLab Wiki. Criar issues para ações de mitigação. Estabelecer daily check-in para identificar problemas cedo. Designar "backup" para cada tarefa crítica.

### **e) Documentar Ações Tomadas (1 ponto)**

#### **Documentação Necessária:**

- **Registro de Mudanças de Escopo (Change Log):** Documentar todas as mudanças solicitadas e decisões tomadas
- **Cronograma Revisado:** Atualizar cronograma com novas datas e justificativas
- **Plano de Ação de Qualidade Atualizado:** Documentar novas medidas de controle
- **Registro de Riscos Atualizado:** Atualizar matriz de riscos e estratégias de resposta
- **Ata de Reunião:** Registrar reunião onde decisões foram tomadas, participantes, ações acordadas
- **Plano de Ação Integrado:** Documento consolidado com todas as ações (escopo, cronograma, qualidade, riscos)
- **Lições Aprendidas Parciais:** Registrar o que foi aprendido com a situação

**Onde Documentar:** GitLab Wiki (documentação centralizada), Issues no GitLab (rastreamento de ações), Milestones atualizados (cronograma), Commits com mensagens descritivas.

**Exemplo Prático:** Criar página na Wiki: "Plano de Ação - Contenção de Escopo". Atualizar Project Charter com mudanças aprovadas. Criar issues para cada ação do plano. Documentar em ata de reunião as decisões tomadas.

**CrITÉRIOS Adicionais para Excelência:** Integração clara entre diferentes áreas de conhecimento, exemplos práticos relacionados ao Portfólio, ações específicas e acionáveis, demonstração de compreensão sistêmica do problema, plano coerente e viável.

## ✗ RESPOSTA NEGATIVA (INSUFICIENTE - 0-2 pontos)

### Erros Comuns:

- Respostas muito genéricas sem detalhamento
- Falta de integração entre áreas de conhecimento
- Ações não específicas ou não acionáveis
- Não mencionar técnicas específicas de gestão
- Falta de exemplos práticos
- Documentação muito vaga ou incompleta
- Não demonstrar compreensão sistêmica
- Plano incoerente ou inviável

### **Exemplo de Resposta Incorreta:**

*"Para conter escopo, não adicionar mais coisas. Para ajustar cronograma, adiar prazos. Para qualidade, testar. Para riscos, evitar. Documentar tudo."*

**Por que está errado:** Respostas muito genéricas, falta de técnicas específicas, não demonstra integração de conhecimentos, ações não detalhadas, falta de exemplos práticos, documentação muito vaga.