

GPS - Gestão de Projetos de Software

Aula 1: Apresentacao e Iniciacao

Prof. Aléssio Miranda Júnior
alessio@cefetmg.br

CEFET-MG - Campus Timoteo
Dep. Engenharia de Computação

Agosto de 2026

- 1 Apresentação da Disciplina
- 2 Introdução a Gestão de Projetos
- 3 O Papel do Gerente de Projeto
- 4 Projeto Integrador: Assistente IA (Tutor ICPC)
- 5 Cronograma e Avaliação
- 6 Ferramentas e Recursos
- 7 Próximos Passos

Disciplina: Gestao de Projetos de Software

Codigo: G07GPSO0.01

Carga Horaria: 30 horas

Creditos: 2

Periodo: 6º

Natureza: Teorica

Area: Profissionalizante

Departamento: DECOMTM

Pre-requisitos: Modelagem de Sistemas

Importante!

Questionario Google Forms ao final de cada aula para validacao de presenca e aprendizado.

Objetivos da Disciplina

1. **Compreender** os fundamentos da gestao de projetos de software
2. **Conhecer** abordagens tradicionais, ageis e hibridas de gestao
3. **Aplicar** praticas de planejamento, execucao, monitoramento e encerramento
4. **Utilizar** ferramentas de apoio ao gerenciamento de projetos
5. **Desenvolver** habilidades de lideranca e trabalho em equipe

Meta

Ao final da disciplina, voce sera capaz de gerenciar projetos de software reais!

- **Aulas Expositivas e Dialogadas** - Teoria com discussao
- **Aprendizagem Baseada em Projetos (PBL)** - Projeto integrador
- **Aprendizagem Colaborativa** - Trabalho em equipe
- **Atividades Hands-on** - Uso de ferramentas praticas
- **Avaliacoes Continuas** - Questionarios semanais

Destaque

O nucleo da disciplina e o **trabalho pratico integrador** - **Assistente IA (Tutor ICPC)**

O que vimos:

- Informacoes gerais da disciplina
- Objetivos de aprendizado
- Metodologia de ensino

Proximo:

- Introducao a Gestao de Projetos
- Conceitos fundamentais
- Diferencas com operacoes

20% Concluido



O que é um Projeto?

Definicao

Um projeto é um esforço **temporário** empreendido para criar um produto, serviço ou resultado **único**.

Características principais:

- **Temporário:** tem início e fim definidos
- **Único:** produto/serviço específico
- **Progressivo:** desenvolvido incrementalmente
- **Recursos limitados:** tempo, custo e escopo

Projetos:

- Temporarios
- Unicos
- Mudanca
- Inovacao
- Risco

Operacoes:

- Continuas
- Repetitivas
- Estabilidade
- Eficiencia
- Previsibilidade

Reflexao

Qual atividade voce considera mais desafiadora: gerenciar projetos ou operacoes?

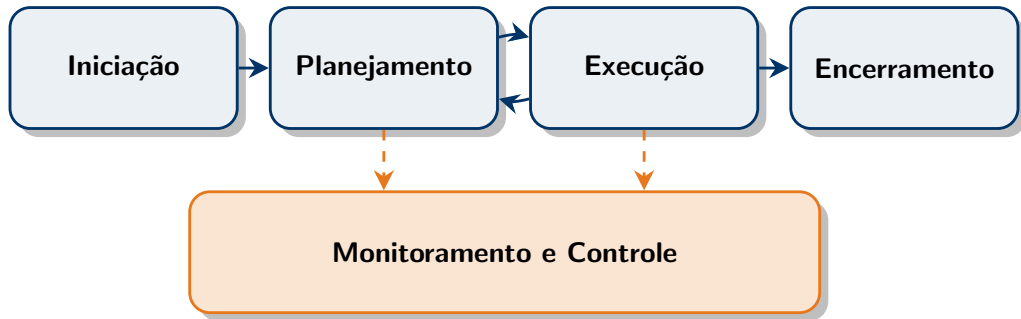
Por que Gerenciar Projetos?

- **Complexidade crescente** dos sistemas de software
- **Competicao acirrada** no mercado
- **Recursos limitados** (tempo, dinheiro, pessoas)
- **Expectativas dos stakeholders** cada vez maiores
- **Necessidade de controle** e visibilidade
- **Reducao de riscos** e incertezas

Estatistica Alarmante

70% dos projetos de software falham devido a problemas de gestao, nao tecnicos.

O Ciclo de Vida: Os 5 Grupos de Processos



Quem é o Gerente de Projeto?

Definicao

O Gerente de Projeto é a pessoa responsável por **alcançar os objetivos do projeto** dentro das restrições de tempo, custo e escopo.

Responsabilidades principais:

- **Planejar** o projeto
- **Organizar** recursos
- **Liderar** a equipe
- **Controlar** o progresso
- **Comunicar** com stakeholders

Habilidades Tecnicas:

- Conhecimento da area
- Metodologias de gestao
- Ferramentas e tecnicas
- Analise de dados

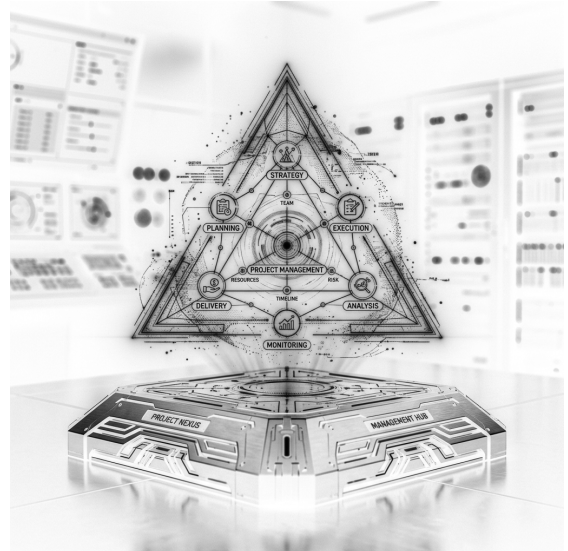
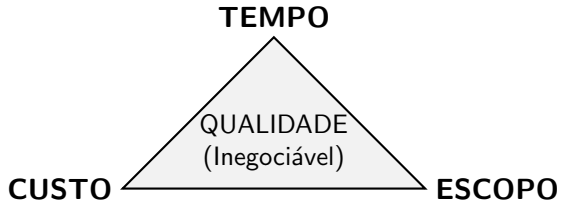
Habilidades Comportamentais:

- Lideranca
- Comunicacao
- Negociacao
- Resolucao de conflitos
- Trabalho em equipe

Dica

70% do sucesso de um projeto depende das habilidades comportamentais!

Triângulo de Ferro



Tutor IA para Maratona

Desenvolver um Bot Assistente utilizando a API de um LLM (ex: Gemini) focado em apoiar estudantes na resolucao de problemas da Maratona de Programacao (ICPC), aplicando os conceitos de gestao de projetos.

Caracteristicas do projeto:

- **Duracao:** 15 semanas
- **Equipes:** 3 alunos
- **Tecnologias:** Python, API Gemini/OpenAI, Telegram/Streamlit, GitLab
- **Entregas:** Incrementais a cada sprint
- **Avaliacao:** 30 pontos da nota final

O Problema e o Desafio

O Contexto (ICPC):

- Alunos treinam resolvendo problemas complexos (ex: Beecrowd).
- Muitas vezes travam em pequenos detalhes de logica ou sintaxe.

O Desafio da IA:

- **Nao dar a resposta pronta!**
- Atuar como um **Tutor**, fornecendo dicas, identificando erros e direcionando o raciocinio.

Engenharia de Prompt

O maior desafio tecnico e ajustar o *prompt* para que a IA se comporte estritamente como um treinador.

Usuario: Bot, estou tentando resolver o problema 1001 do Beecrowd, mas recebo "Time Limit Exceeded".

Tutor IA: Ola! Notei que voce esta usando dois lacos de repeticao aninhados (for dentro de for), resultando em complexidade $O(N^2)$. Para $N = 10^5$, isso e muito lento. Tente pensar em um *Hash Map* para fazer a busca em $O(1)$. Quer tentar implementar?

- **Custo:** Precisamos pagar pela API?
 - **Nao.** APIs como Gemini possuem um Free Tier perfeito para o projeto.
- **Scope Creep:** "Podemos fazer um app mobile completo?"
 - **Cuidado!** Foquem no simples (ex: Telegram Bot). O foco da disciplina e a **Gestao do Projeto**, e nao a arquitetura do software.
- **Limitacoes Tecnicas:**
 - APIs gratuitas limitam requisicoes por minuto (Rate Limits). A equipe precisara lidar com isso.

Estratégias de Sobrevivência (O Código não é tudo)

- **A Morte do "Lobo Solitário":** O GitLab entregará quem não trabalha. Tarefas rastreáveis via *Issues* e *Merge Requests* são inegociáveis.
- **A Armadilha do Prompt (Scope Creep):** Domar a IA e evitar "alucinações" consumirá o dobro do tempo planejado na Sprint.
- **Evite o "Escopo Ouro":** Esqueça interfaces extravagantes. Foque 100% em construir um **MVP** (Assistente funcional e limpo).
- **Versionamento de Conhecimento:** Documentem na *Wiki* do GitLab cada *System Prompt* que funcionou e o porquê.
- **Comunicação Transparente:** Um atraso reportado cedo é contornável; escondê-lo é fatal.

Critérios para formacao:

- **Tamanho:** 3 alunos por equipe
- **Diversidade:** Diferentes niveis de conhecimento
- **Compatibilidade:** Horarios e disponibilidade
- **Complementaridade:** Habilidades tecnicas variadas

Papeis na equipe:

- **Scrum Master** - Facilita o processo agil
- **Product Owner** - Define prioridades
- **Desenvolvedores** - Implementam funcionalidades
- **Tester** - Garante qualidade

Modulo 1 (Aulas 1-3):

- Fundamentos e Iniciacao
- Viabilidade e Estrutura
- Planejamento no GitLab

Modulo 2 (Aulas 4-7):

- Gestao Tradicional
- PMBOK e Metodologias
- Planejamento Detalhado

Modulos 3-4: Gestao Agil, Hibrida e Encerramento

20% do Semestre

Distribuicao de Pontos

Total: 100 pontos

- **Questionarios semanais:** 20 pontos (sem reposicao)
- **Trabalho pratico (Assistente IA):** 30 pontos
- **Prova 1 (Modulos 1-2):** 25 pontos
- **Prova 2 (Modulos 3-4):** 25 pontos

Importante!

Questionario Google Forms ao final de cada aula para validacao de presenca e aprendizado.

Desenvolvimento e I.A.:

- **Python** - Linguagem principal
- **APIs (Gemini/OpenAI)** - Inteligência Artificial
- **Telegram / Streamlit** - Interface do Chat
- **Markdown / Wiki** - Documentação

Gestao:

- **GitLab Issues** - Rastreamento de tarefas
- **GitLab Boards** - Quadros Kanban
- **GitLab CI/CD** - Automacao
- **GitLab Issues** - Avaliações Individuais

Materiais disponiveis:

- **Slides das aulas** em LaTeX/Beamer
- **Templates** para documentos de projeto
- **Exemplos praticos** de cada conceito
- **Bibliografia** recomendada
- **Repositorio GitLab** da disciplina

Suporte:

- **Aulas** - Segunda-Feira, 7h às 8:40h
- **Email:** alessio@cefetmg.br
- **WhatsApp** para duvidas rapidas
- **Documentacao online** sempre atualizada

O Ritual de Passagem (Individual)

Antes de formar as equipes, todo engenheiro precisa preparar seu ambiente de avaliações.

1. **Aceite o Convite:** Você receberá no e-mail o convite para o *Grupo Principal* da disciplina no GitLab (cefet-gps-2026).
2. **Crie seu Repositório Pessoal:** Dentro do subgrupo gps-avaliacoes-individuais, clique em *New Project* e crie o repositório avaliacao-SeuNome.
3. **Fique Pronto:** Todas as provas da disciplina chegarão automaticamente como **Issues** dentro do seu repositório pessoal! A nota será calculada via API do professor.

Aula 2: Viabilidade e Estrutura

- **Conteudo Teorico:**
 - Engenharia de Software e Gestao de Projetos
 - Estudo de viabilidade tecnico-economica
 - Analise de riscos iniciais
- **Conteudo Pratico:**
 - Definicao inicial do escopo do Assistente de I.A.
 - Alinhamento de expectativas da equipe
 - Analise de viabilidade tecnica do projeto

Tarefas para a Proxima Aula

1. **Formar equipes** (3 alunos)
2. **Definir nome da equipe** e identidade visual
3. **Instalar bibliotecas (ex: `python-telegram-bot`)** localmente para testes
4. **Criar conta no GitLab** (se nao tiver)
5. **Ler material** sobre viabilidade de projetos
6. **Responder questionario** da Aula 1

Lembrete

Questionario Google Forms ao final da aula para validacao de presenca e aprendizado.

?

Pense sobre o que acabamos de discutir...

Alguma duvida sobre:

- Objetivos da disciplina?
- Conceito de projeto?
- Papel do gerente?
- Projeto integrador?

Levante a mao se tiver duvidas!

Duvidas?

Email: alessio@cefetmg.br

Aula: Segunda-Feira, 7h às 8:40h

WhatsApp: Dúvidas rápidas

Lembrete

Nao esqueca de responder o questionario!

Proxima aula: **Viabilidade e Estrutura**

100% da Aula 1 Concluida!