

CURSO:	Análise e Desenvolvimento de Sistemas – Gerência de Projetos	Avaliação Formadora (Competência 3)
Analisar, projetar e programar a Gerência de Projetos 4.0		

Tipo de Atividade

- a) Trabalho Individual
- b) Valor: 15 pontos

Notebook de Monitoramento e Análise do Projeto "Smart Office"

Objetivos de Aprendizagem:

- Aplicar métricas de desempenho de projeto (KPIs) em um contexto prático.
- Utilizar **Python (com Pandas)** para carregar, limpar e analisar múltiplos datasets (progresso do projeto e dados de IoT).
- Criar visualizações de dados eficazes (**Matplotlib/Seaborn**) para comunicar o status do projeto e os resultados dos sensores.
- Desenvolver a habilidade de **Data Storytelling**, interpretando gráficos e dados para suportar a tomada de decisão.
- Realizar uma análise exploratória cruzando dados de diferentes fontes para gerar insights de negócio.

Contexto do Projeto:

Você continua no papel de Gerente de Projetos da iniciativa "**Smart Office**". A fase de planejamento analítico foi concluída. Agora, o projeto está em execução e você precisa monitorar o seu progresso e começar a analisar os dados que estão sendo gerados.

Seu principal instrumento de trabalho será um **Jupyter Notebook**, que funcionará como um dashboard analítico vivo, combinando código, visualizações e suas conclusões.

Datasets Necessários:

1. `smart_office_data.csv`: O arquivo com os dados simulados dos sensores de IoT que você criou na competência anterior.
2. `project_progress.csv`: Um novo arquivo que você deverá criar para simular o progresso das tarefas do projeto.

Estrutura do arquivo `project_progress.csv` (Crie este arquivo):

task_id	task_name	sprint	planned_hours	actual_hours	status	completion_date
1	Seleção de Fornecedores	1	20	25	Done	2025-08-08
2	Instalação Sensores Ocupação	1	40	38	Done	2025-08-12
3	Instalação Sensores Luz	2	30	35	Done	2025-08-20
4	Desenvolvimento Backend	2	50	0	In Progress	
5	Desenvolvimento Dashboard	3	60	0	To Do	
6	Testes de Integração	3	40	0	To Do	

Descrição da Atividade (Fases de Construção do Notebook)

O seu Jupyter Notebook (`analise_smart_office.ipynb`) será desenvolvido ao longo de 4 fases.

Fase 1: Definição de KPIs e Análise Inicial

1. **Definição de KPIs:** Na primeira célula (em Markdown), defina 3 KPIs que você usará para monitorar o projeto. Pelo menos um deve ser de projeto (ex: Variação de Custo/Prazo - SPI/CPI) e um de produto/resultado (ex: Redução percentual no consumo de energia).
2. **Carregamento de Dados:** Em uma célula de código, importe a biblioteca Pandas e carregue os dois arquivos (smart_office_data.csv e project_progress.csv) em dois DataFrames distintos.
3. **Análise Exploratória Básica:** Exiba as primeiras linhas (.head()), informações gerais (.info()) e estatísticas descritivas (.describe()) de cada DataFrame para garantir que os dados foram carregados corretamente.

Fase 2: Visualização do Progresso do Projeto

Nesta fase, foque apenas no DataFrame de progresso do projeto.

1. **Gráfico de Esforço (Planejado vs. Real):**
 - Crie um gráfico de barras comparando as planned_hours com as actual_hours para as tarefas que já foram concluídas (Done).
 - Utilize Matplotlib ou Seaborn. Adicione títulos claros ao gráfico e aos eixos.
2. **Criação de um Gráfico de Burndown:**
 - Calcule o "trabalho restante" ao longo do tempo (ou por sprint). O gráfico de Burndown mostra a quantidade de trabalho planejado que ainda falta para ser concluído.
 - Crie um gráfico de linhas que represente o Burndown do projeto. O eixo X pode representar os Sprints e o eixo Y as horas restantes.
 - Em uma célula de Markdown abaixo do gráfico, explique o que o gráfico está mostrando (Data Storytelling). O projeto está adiantado ou atrasado em relação ao plano?

Fase 3: Análise Exploratória dos Dados de IoT

Agora, mude o foco para o DataFrame dos sensores.

1. **Análise de um Sensor:**
 - Escolha um dos sensores (ex: temperatura ou luminosidade).
 - Crie um gráfico de linhas para visualizar a variação dos dados do sensor ao longo de um período (ex: 24 ou 48 horas). Isso ajuda a identificar padrões diários.
2. **Cruzamento de Dados (Projeto x IoT):**
 - Esta é a análise mais importante. Crie uma visualização que cruze uma informação do projeto com uma informação dos sensores.
 - **Exemplo de Insight:** Filtre os dados de luminosidade para mostrar a média de consumo *antes* e *depois* da completion_date da tarefa "Instalação Sensores Luz". Crie um gráfico de barras que mostre essa comparação, comprovando (ou não) a eficácia da instalação.
 - **Dica:** Você precisará converter as colunas de data para o formato datetime do Pandas para fazer a filtragem corretamente.
 -

Fase 4: Conclusões e Suporte à Decisão

Esta é a fase final de análise e comunicação.

1. **Identificação de Desvios e Insights:**
 - Em uma célula de Markdown, escreva um resumo executivo (2-3 parágrafos).
 - **Parágrafo 1 (Progresso do Projeto):** Com base nos gráficos da Fase 2, descreva o status do projeto. A tarefa "Seleção de Fornecedores" estourou o planejado? O Burndown indica algum risco?
 - **Parágrafo 2 (Resultados do Produto):** Com base na análise de cruzamento de dados da Fase 3, qual o resultado inicial do projeto? A instalação dos sensores de luz já demonstra uma redução no consumo?

2. Tomada de Decisão:

3.

- Com base na sua análise, identifique **uma decisão** que precisa ser tomada.
- **Exemplo:** "Com base no estouro de horas na seleção de fornecedores, recomendo uma revisão do planejamento de horas para a tarefa 'Desenvolvimento Backend', que é mais complexa e pode estar subestimada." ou "Os resultados positivos do sensor de luz justificam a aceleração da instalação dos sensores de temperatura para maximizar a economia de energia."

Formato de Entrega Final:

- Um único arquivo Jupyter Notebook (analise_smart_office.ipynb) contendo todo o código, as visualizações e as análises em Markdown, organizado de forma clara e sequencial.
- O notebook deve ser "auto-contido", ou seja, deve poder ser executado do início ao fim sem erros (assumindo que os arquivos .csv estão na mesma pasta).