

ADS /GTI**SPRINT 1 – MISSÃO 1****PROJETO: “DEPLOYMENT QUALITY ASSURANCE”****ESTUDO DE CASO**

Uma certa empresa decidiu estabelecer uma cultura *QUALITY ASSURANCE* em seu modelo de negócio, visando impactar positivamente processos de qualidade em suas áreas de operação e tecnologia.

ESCOPO DO PROJETO

O projeto será composto por 2 Sprints (AC-1 e AC-2) que se complementam, onde os alunos deverão construir ações que validem a empresa a cultura orientada a Q.A.

Os alunos desenvolverão projeto 3 em Sprints:

- SPRINT 1: Vale até 1,25 ponto na AC-1
- SPRINT 2: Vale até 1,25 ponto na AC-2

OBJETIVO

Aprender as nuances e aplicabilidade do *QUALITY ASSURANCE* em uma organização. Construir um projeto de implementação de Gerenciamento de Qualidade Total e realizar atividades que valem nota.

SPRINT 1 (1,25 ponto)

Composto por 4 missões que se complementam para a entrega total do projeto:

- Missão 1: Plano do Projeto e Preparação – Vale 20% AC-1
- Missão 2: Testes Funcionais – Vale 20% AC-1
- Missão 3: Testes Não Funcionais – Vale 20% AC-1
- Missão 4: Testes de Integração de automação com CI/CD (parte 1) – Vale 20% AC-1
- Missão 4: Documentação e entrega final – Vale 20% AC-1

MISSÃO 1**VALE 20% DA NOTA AC-1****Etapas preliminar:**

1. Definir a Grupo (máximo 5 alunos);
2. Enviar email ao professor oficializando os Nomes e RAs de cada dupla ou trio para a FASE 1: flavio.santarelli@pro.fecaf.com.br;
3. Possuir ou Abrir conta no GitHub.

Etapa 1 - Planejamento do Projeto (GITHUB - método Kanban):

4. No GitHub decidir qual conta será utilizada de forma oficial para interação com o professor;
5. Todos os alunos devem ter ou criar uma conta no GitHub para execução e acompanhamento das atividades do projeto e incrementar seu portfólio individual. Além de subir códigos, arquivos e produtos que serão criados neste projeto;
6. Criar repositório público. Em seguida, criar projeto com nome “PROJETO-QA-ADS”;
7. Adicionar os demais integrantes e o professor (PROFSANTARELLI);
8. Selecionar modelo “quadro” (na versão inglês escolher “frame”)
9. Adicionar listas com os seguintes nomes:
 - **A INICIAR**
 - **EM ANDAMENTO**
 - **EM VALIDAÇÃO**
 - **FINALIZADO**
10. Em A INICIAR criar o item “MISSÃO-1”, “MISSÃO-2”, “MISSÃO-3”, “MISSÃO-4” e “MISSÃO-5”
11. Colocar “MISSÃO-1” na lista EM ANDAMENTO;

ETAPA 2 – EXECUÇÃO DE TESTE UNITÁRIO:

12. Acesse o Google Colab e crie uma conta de uso, ou acesse com sua conta Google;
13. Abra um novo NOTEBOOK COLAB para realizar a criação e execução do código e script de teste;
14. Configuração e uso do Pytest:

O Pytest é um framework de testes para Python, amplamente utilizado para escrever testes automatizados de software. Oferece funcionalidades como fixtures, parametrização e relatórios.

15. Acesse o repositório GitHub do Professor para pegar o código, através do endereço abaixo:

<https://github.com/PROFSANTARELLI/TESTES-QA-2025>

16. Suba os arquivos para o Google Colab;
17. Código e os Testes: vamos executar uma função simples com código Python e criar testes para ela. A função será uma calculadora básica com soma, subtração, multiplicação e divisão;
18. Instale o Pytest: `pip install pytest`
19. Realize os testes automatizados e colete os resultados;
20. Execute o comando: `pytest test_calculadora.py`
21. Teste 1: código com erro;
22. Após execução do Teste 1, colete os dados da primeira rodada, volte ao quadro Kanban do GitHub, abra e edite o cartão Missão 1 e copie o resultado no quadro;
23. Teste 2: código corrigido;
24. Após execução do Teste 2, colete os dados da primeira rodada, volte ao quadro Kanban do GitHub, abra e edite o cartão Missão 1 e copie o resultado na sequência do anterior;

TAREFA 3 - FINALIZAÇÃO:

25. Após execução e cópia dos Testes no cartão, vocês irão pular duas linhas após a o último resultado e inserir nome completo e RA dos alunos presentes (sem RA não consigo computar presença);
26. Grave (comittar) as informações e feche o cartão;
27. Suba e grave os arquivos da calculadora e de teste da calculadora
28. Coloque o cartão MISSÃO 1 na lista: EM VALIDAÇÃO;
29. Volte no AVA e baixe esse documento no computador;
30. Volte no GitHub e entre novamente no repositório criado;
31. Anexe o documento baixado no computador e grave (comittar) o anexo;

FIM – SUCESSO A TODOS!!!