Projetos-Stress

**. Fundamentos do Stress Testing em Projetos Web-Nicolly**

* **Definição**: Explicar o que é stress testing e por que ele é essencial em projetos web.
* **Objetivos**: Identificar o propósito principal, como testar a resistência de um sistema sob carga extrema.
* **Importância**: Discutir como o stress testing ajuda a garantir que o sistema não falhe durante picos de tráfego ou eventos inesperados.

**2. Planejamento e Preparação para o Stress Test-Renan**

* **Identificação de Cenários**: Definir os cenários de estresse que serão testados, como picos de tráfego, uso excessivo de recursos e falhas de rede.
* **Escolha de Ferramentas**: Apresentar ferramentas comuns para stress testing em projetos web, como Apache JMeter, Gatling, Locust, etc.
* **Ambiente de Teste**: Discutir a configuração necessária para realizar os testes de forma eficaz sem afetar o ambiente de produção.

**3. Execução do Stress Test-Yasmim**

* **Metodologia**: Explicar como o stress test será conduzido, incluindo a definição de metas (número de usuários simultâneos, duração, intensidade).
* **Monitoramento e Coleta de Dados**: Descrever as métricas importantes a serem monitoradas durante o teste, como tempo de resposta, taxa de falhas e utilização de recursos (CPU, memória, etc.).
* **Escalonamento Progressivo**: Detalhar a abordagem de aumentar gradualmente a carga para identificar o ponto de falha do sistema.

**4. Análise de Resultados e Melhoria Pós-Teste-Alysson**

* **Interpretação de Dados**: Analisar os resultados do teste, identificando gargalos, falhas ou limitações de desempenho.
* **Ações Corretivas**: Discutir as possíveis ações corretivas, como otimização de código, melhorias na infraestrutura ou ajustes no balanceamento de carga.
* **Revisão e Iteração**: Enfatizar a importância de realizar novos testes após as correções e melhorias, para garantir que o sistema continue robusto.