**Pílula do conhecimento: Requisitos Não-Funcionais**

A primeira e mais crucial etapa na área de performance é o acordo dos **requisitos não-funcionais**. Esses requisitos são a respeito de exigências como desempenho, confiabilidade, manutenção, segurança e entre outros. Neles são definidas várias informações-chave, principalmente para a estratégia e o monitoramento de um **teste de performance**.

É importante ressaltar que esses mesmos requisitos são acertados no padrão de qualidade e nas necessidades do cliente, ou seja, exigências apontadas pela equipe que solicitou o **teste de performance**.

A parte disso, cada equipe ou projeto podem possuir requisitos diferentes, porém há alguns que são extremamente frequente e essenciais como:

1. **Objetivos e Prazos**: o primeiro passo é compreender os objetivos do sistema ou da funcionalidade, seus problemas de performance, sua situação e principalmente seus **prazos de entrega**. Esses são pontos cruciais para poder marcar **testes de performance** ou até mesmo começar o desenvolvimento de um script do teste.
2. **Infraestrutura e Ambiente**: para a criação do script é necessário entender o ambiente que será testado. Um mapeamento da infraestrutura e das tecnologias utilizadas é ideal para os primeiros passos antes de um teste. Coleções ou documentações de ferramentas como *Postman* e *Swagger* são os principais pontos para agilizar esse processo.
3. **Volumetria e Massa de Dados**: para cumprir o objetivo de um teste de performance é necessário entender os padrões de qualidade exigidos. Informações como expectativas de Vazão (requisições por segundo), tempo de resposta, porcentagem de erros e entre outros. Outro ponto a ser definido é se o fluxo necessita de uma massa de dados (exemplo: lista de CPFs cadastrados).
4. **Monitoramento e Entrega**: a disponibilidade de um sistema de monitoração é importante para análise do teste, podendo assim ver como a aplicação se comporta internamente com altas cargas de usuários simultâneos, ferramentas como *Dynatrace* ou *Datadog* auxiliam nesse aspecto. Com todas essas informações é possível a entrega de um relatório ou apresentação solida do estado de uma aplicação e assim apontar melhorias.