

## Desenho do Experimento

### 1. A - Hipóteses Nula e Alternativa

- **RQ1 - Respostas à consultas GraphQL são mais rápidas que respostas à consultas REST?**

**Hipótese nula:** O tempo de resposta médio das consultas GraphQL não é significativamente diferente do tempo de resposta médio das consultas REST.

**Hipótese alternativa:** O tempo de resposta médio das consultas GraphQL é significativamente menor que o tempo de resposta médio das consultas REST.

- **RQ2 - Respostas à consultas GraphQL tem tamanho menor que respostas à consultas REST?**

**Hipótese nula:** O tamanho médio das respostas de consultas GraphQL não é significativamente diferente do tamanho médio das respostas de consultas REST.

**Hipótese alternativa:** O tamanho médio das respostas de consultas GraphQL é significativamente menor que o tamanho médio das respostas de consultas REST.

### B - Variáveis Dependentes

- Tempo de resposta das consultas (em milissegundos).
- Tamanho da resposta (em bytes).

### C - Variáveis Independentes

- Tipo de API: REST ou GraphQL.
- Complexidade da consulta (simples ou complexa).
- Quantidade de dados retornados pela consulta.

### D - Tratamentos

Implementação de consultas em GraphQL e REST que retornem o mesmo conjunto de informações para diferentes níveis de complexidade

### E - Objetos Experimentais

- Endpoints de uma aplicação hipotética que expõem APIs tanto em GraphQL quanto em REST.
- Banco de dados que suporte ambas as implementações.

### F - Tipo de Projeto Experimental

**Desenho cruzado (crossing):** Cada tipo de consulta será avaliado nas duas APIs para garantir comparabilidade.

## **G - Quantidade de Medições**

Para responder às questões de pesquisa (RQ1 e RQ2), foi utilizado um experimento com diferentes scripts para medir o desempenho das APIs GraphQL e REST. Cada script realizou várias medições para diferentes tipos de consultas, conforme descrito abaixo:

### **Primeiro Script**

- Foram realizadas 30 medições para cada método (GraphQL e REST).
- Cada medição consistiu em executar uma consulta simples que retornava informações básicas sobre repositórios do GitHub.
- Os tempos de resposta e tamanhos das respostas foram registrados, permitindo uma análise estatística de média, desvio padrão e intervalos de confiança.

### **Segundo Script**

- Para maior robustez, foram realizadas 10 medições adicionais para dois tipos de consultas (simples e complexas) em ambas as APIs.
- Consultas Simples: Retornavam informações básicas sobre repositórios, como nome e descrição.
- Consultas Complexas: Retornavam informações mais detalhadas, incluindo dados como número de estrelas, forks, idioma principal, proprietário e detalhes sobre as issues.
- Para o GraphQL, foi utilizada paginação para garantir a obtenção de até 100 itens, divididos em páginas de 20.
- No REST, foram usadas requisições paginadas com parâmetros semelhantes, e, no caso das consultas complexas, requisições adicionais foram realizadas para coletar dados complementares (como informações do proprietário e issues).
- Os tempos de resposta e tamanhos das respostas foram registrados para cada consulta e agrupados por método e tipo de consulta.

## **H - Ameaças à Validade**

### **1. Ameaças internas:**

Variações de desempenho no servidor ou banco de dados durante a execução das medições.

Diferenças na configuração dos endpoints GraphQL e REST.

### **2. Ameaças externas:**

Generalização dos resultados para outros contextos ou APIs.

Tamanho e complexidade limitados das consultas testadas.

### 3. **De Construção:**

Inconsistências na implementação das consultas em GraphQL e REST.