

**INFORMAÇÕES SOBRE A AVALIAÇÃO**

<b>LAB05</b>	<b>Laboratório 05 - 20 pontos</b>
--------------	-----------------------------------

INFORMAÇÕES DOCENTE						
CURSO: ENGENHARIA DE SOFTWARE	DISCIPLINA: LABORATÓRIO DE EXPERIMENTAÇÃO DE SOFTWARE	TURNO	MANHÃ	TARDE	NOITE	PERÍODO/SALA: 6º
					x	
PROFESSOR (A): João Paulo Carneiro Aramuni						

### GraphQL vs REST - Um experimento controlado

A linguagem de consulta GraphQL, proposta pelo Facebook como metodologia de implementação de APIs Web, representa uma alternativa às populares APIs REST. Baseada em grafos, a linguagem permite que usuários consultem banco de dados na forma de schemas, de modo que se possa exportar a base e realizar consultas num formato definido pelo fornecedor da API. Por outro lado, APIs criados com base em abordagens REST baseiam-se em endpoints: operações pré-definidas que podem ser chamadas por clientes que desejam consultar, deletar, atualizar ou escrever um dado na base. Desde o seu surgimento, vários sistemas realizaram a migração entre ambas as soluções, mantendo soluções compatíveis REST, mas oferecendo os benefícios da nova linguagem de consulta proposta. Entretanto, **não está claro quais os reais benefícios da adoção de uma API GraphQL em detrimento de uma API REST**. Nesse contexto, o objetivo deste laboratório é realizar um **experimento controlado** para avaliar quantitativamente os benefícios da adoção de uma API GraphQL. Especificamente, as seguintes perguntas devem ser respondidas:

**RQ1.** Respostas às consultas GraphQL são mais rápidas que respostas às consultas REST?

**RQ2.** Respostas às consultas GraphQL tem tamanho menor que respostas às consultas REST?

Para tanto, espera-se que a cada sprint o grupo realize algumas das seguintes etapas:

#### 1. Desenho do Experimento

Nessa etapa, realize o desenho do seu experimento, definindo (pelo menos) cada um dos tópicos a seguir:

- A. Hipóteses Nula e Alternativa
- B. Variáveis Dependentes
- B. Variáveis Independentes
- C. Tratamentos
- D. Objetos Experimentais
- E. Tipo de Projeto Experimental
- F. Quantidade de Medições
- G. Ameaças à Validade

## **2. Preparação do Experimento**

Em seguida, realize o que for necessário para aplicação dos tratamentos (desenvolvimento de scripts, consultas, escolha de bibliotecas, entre outros). Monte todo o cenário experimental, com base na definição anterior.

## **3. Execução do Experimento**

Siga cuidadosamente os passos do plano, realizando as medições e aplicando os tratamentos.

## **4. Análise de Resultados**

Inicialmente, realize uma revisão dos valores obtidos, com objetivo de validar as suas medições e observar se algum valor parece fora do esperado. Realize ajustes, se necessário. Em seguida, analise o conjunto de dados com os princípios estatísticos mais adequados.

## **5. Relatório Final**

Por fim, elabore um documento que apresente (i) uma introdução que descreva as hipóteses levantadas; (ii) a metodologia que você utilizou para executar seu experimento, detalhando todos os aspectos necessários para possibilitar a reprodução e replicação do seu experimento (inclua o ambiente em que os trials foram realizados); (iii) os resultados obtidos para cada uma das perguntas, bem como as respostas estatísticas obtidas; (iv) a discussão final sobre os resultados obtidos.

## **6. Dashboard de Visualização**

Importe os dados resultantes do experimento (Sprint 2) e processe-os para gerar tabelas e gráficos que permitam uma interpretação clara das diferenças entre REST e GraphQL com base nas métricas avaliadas nas perguntas de pesquisa (RQ1 e RQ2).



Para isso, utilize ferramentas como Pandas, para manipulação e análise de dados, Matplotlib e/ou Seaborn, para criação de visualizações gráficas.

### **Processo de Desenvolvimento**

Sprints e Datas de Entrega

**Lab05S01:** Desenho do experimento, (Passo 1) e preparação (Passo 2). (5 pontos)

**Lab05S02:** Execução do experimento (Passos 3); Análise dos resultados e produção do relatório final (Passos 4 e 5). (10 pontos)

**Lab05S03:** Criação de um dashboard para exibição dos resultados (Passo 6). (5 pontos)

Prazo final (Acesse o cronograma): <https://github.com/joaopauloaramuni/laboratorio-de-experimentacao-de-software/tree/main/CRONOGRAMA>

Valor total: 20 pontos