



---

## Atividades Assíncronas 2 | Desenvolvimento Web Cliente

Olá, esta é a descrição de um conjunto de atividades assíncronas para a semana. As atividades aqui relacionadas, quando realizadas por você e entregues dentro do prazo, serão registradas conforme a referência de datas acima.

### 1. Atividade 1 - Vídeo: Protocolo HTTP

Assista ao vídeo que apresenta o Protocolo HTTP (também postado na seção Aulas Online - aula 5): [https://drive.google.com/file/d/1uO2X4uJ\\_7vm-RB2MEDFPwdgQplyjC4iY/view?usp=sharing](https://drive.google.com/file/d/1uO2X4uJ_7vm-RB2MEDFPwdgQplyjC4iY/view?usp=sharing).

### 2. Atividade 2 - Vídeo: Arquitetura REST

Assista ao vídeo abaixo, que apresenta a arquitetura REST e seu uso em aplicações web: <https://www.youtube.com/watch?v=S7MduKwvVGk>.

### 3. Atividade 3: Pesquisa:

Elabore um texto, de pelo menos 1 página, que explique a Arquitetura REST e o GraphQL, citando suas fontes. Seu texto deve conter os seguintes tópicos:

- Histórico: O que são e quem os criou?;
- Fundamentos: Qual a relação com HTTP?;
- Funcionamento: Quais verbos o REST possui? e Quais os termos do GraphQL?;
- Implementações frameworks/bibliotecas disponíveis.

### 4. O que deve ser entregue?

O arquivo contendo o texto da pesquisa realizada na atividade 3, ou uma explicação detalhada de suas dificuldades para concluir a tarefa.



---

## Resposta

### REST e seus fundamentos

REST é o acrônimo para *Representational State Transfer* (em português, Transferência de Estado Representativo), Consiste em um estilo de desenvolvimento de serviços web que teve origem na tese de doutorado de Roy Fielding, que é um dos autores do protocolo HTTP (*HyperText Transfer Protocol*).

A arquitetura REST fornece acesso aos recursos para que o cliente REST acesse e renderize os recursos no lado do cliente. No estilo REST, URI ou IDs globais ajudam a identificar cada recurso. Um dos princípios da REST é o chamado STATELESSNESS (sem estado), em que o servidor não precisa saber em qual estado o cliente está e vice-versa. Possuindo a seguinte separação:

- Cliente: é o responsável por solicitar um serviço;
- Servidor: é o responsável por prover os serviços.

O REST precisa que um cliente faça uma requisição para o servidor para enviar ou modificar dados. Uma requisição consiste em:

- Um verbo ou método HTTP, que define que tipo de operação o servidor vai realizar;
- Um header, com o cabeçalho da requisição que passa informações sobre a requisição;
- Um path (caminho ou rota) para o servidor;
- Informação no corpo da requisição, sendo esta informação opcional.

Em aplicação REST, os métodos HTTP mais utilizados são:

- GET: usado para solicitar que um servidor envie um determinado recurso/dado;
- POST: usado para enviar dados para o servidor;
- PUT: usado para editar e atualizar dados em um servidor;
- DELETE: usado para deletar dados do servidor.



## GRAPHQL e seus fundamentos

GraphQL é uma linguagem de consulta para APIs e um tempo de execução para preencher essas consultas com seus dados existentes. Basicamente, é usado para carregar dados de um servidor para um cliente. O GraphQL foi desenvolvido pelo Facebook em 2012, desde então, a linguagem de consulta cresceu constantemente em popularidade – em parte devido ao código aberto do Facebook em 2016.

O GraphQL tem três características principais:

- Ele permite que o cliente especifique exatamente quais dados ele precisa;
- Facilita a agregação de dados de várias fontes;
- Ele usa um sistema de tipos para descrever dados.

O GraphQL tem três operações de nível superior:

- Consulta: busca por somente leitura;
- Mutação: gravação, seguida por uma busca;
- Assinatura: conexão de longa duração para o recebimento de dados.

Como o GraphQL é um padrão de comunicação, existem muitas ferramentas para ajudá-lo a começar a trabalhar que suportam o GraphQL, por exemplo:

- GraphQL.js - JavaScript;
- Apollo Server - JavaScript;
- Express GraphQL - JavaScript;
- Graphene - Python;
- Strawberry - Python;
- Juniper - Rust;
- Entre outros (<https://graphql.org/code/>);