Desenvolvimento Web II

Aula 06 - GraphQL
Linguagem de Consulta para APIs

Prof. Fabricio Bizotto

Instituto Federal Catarinense

fabricio.bizotto@ifc.edu.br

Ciência da Computação
31 de março de 2024

Roteiro

Linguagem de Consulta para API

GraphQL

Definição

GraphQL - *Graph Query Language*

- Linguagem de consulta para APIs.
- Foi criada pelo Facebook em 2012 e tornou-se open-source em 2015.
- É uma alternativa ao REST. Permite que os clientes solicitem dados de forma flexível.
- O GraphQL fornece uma descrição completa e compreensível dos dados em sua API, facilitando a evolução das APIs ao longo do tempo.

Características

- **Consulta única** para recuperar apenas os dados necessários.
- **Tipagem forte** para definir a estrutura dos dados.
- **Introspecção** para consultar a estrutura da API. Isso permite que os clientes descubram quais operações podem ser realizadas.
- **Múltiplas operações** em uma única solicitação. Os clientes podem solicitar vários recursos em uma única solicitação.
- **Documentação** embutida. O GraphQL fornece uma descrição completa e compreensível dos dados em sua API.
- **Validação** de consulta. O GraphQL valida a consulta antes de executá-la.

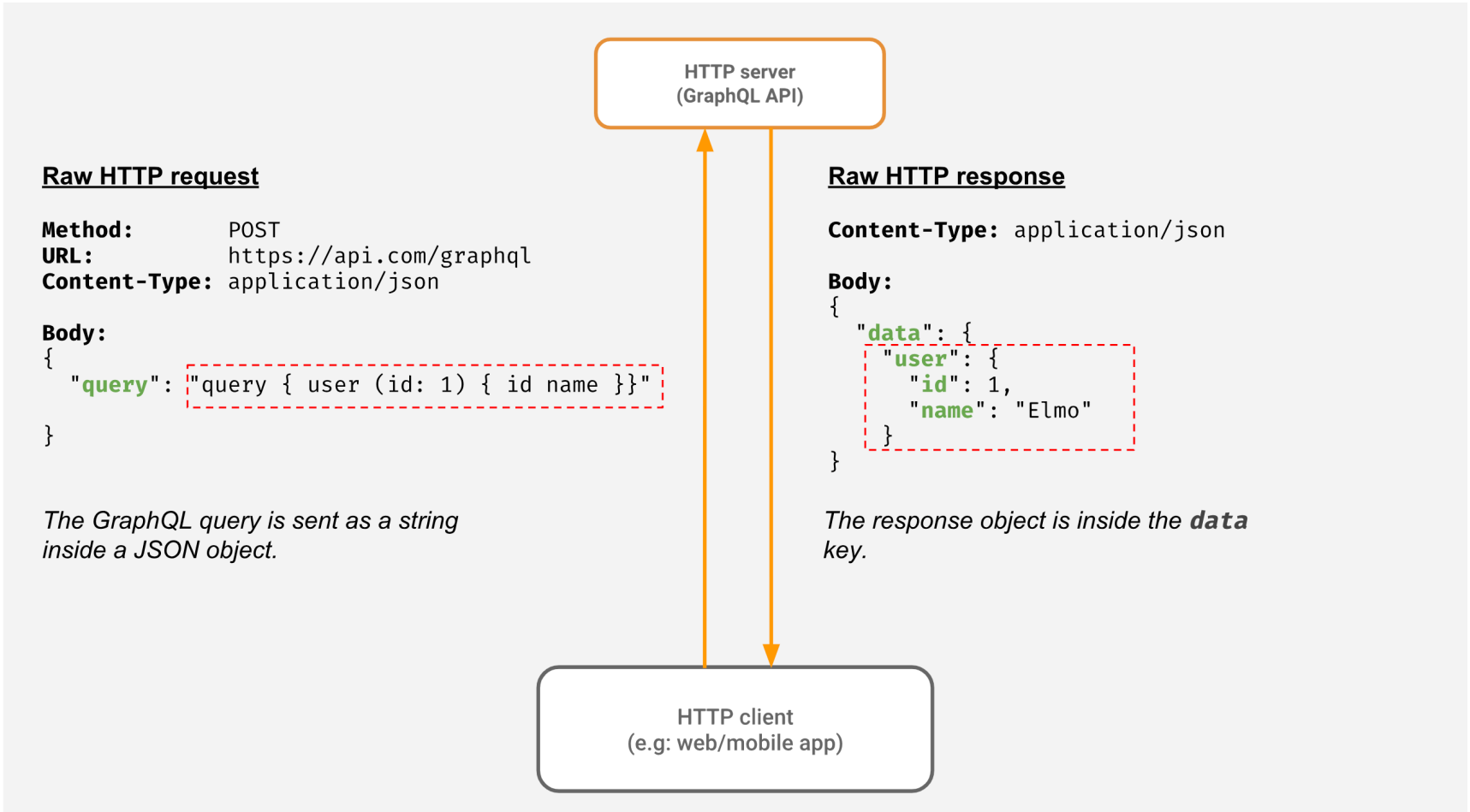

Figura: Estrutura GraphQL

REST	**GraphQL**
É uma arquitetura	É uma linguagem de consulta
Os dados são expostos como recursos	Os dados são expostos como um grafo
Overfetching ou underfetching são comuns.	Os clientes solicitam apenas os dados necessários, evitando overfetching e underfetching.
Múltiplos endpoints para diferentes recursos.	Um único ponto de entrada para todas as operações.
Operações de escrita (POST, PUT, DELETE) têm endpoints separados.	Mutações são tratadas no mesmo sistema de tipos e no mesmo endpoint que as consultas.
Utiliza diversos métodos HTTP (GET, POST, PUT, DELETE).	Usa o método POST para todas as operações. Pode ser usado com qualquer protocolo de transporte.
Pode exigir versionamento da API para adicionar ou modificar funcionalidades.	Não requer versionamento devido à flexibilidade nas consultas.

Tabela: REST vs GraphQL

Mutation

- É um tipo de operação que permite criar, atualizar ou excluir dados.
- É semelhante a uma operação de escrita no REST.
- As mutações são tratadas no mesmo sistema de tipos e no mesmo endpoint que as consultas.

```
mutation {
  createTask(
      title: "Estudar",
      description: "Estudar Web Service") {
    task {
      id
      title
      description
    }
  }
}
```

```
{
  "data": {
    "createTask": {
      "task": {
        "id": "3",
        "title": "Estudar",
        "description": "Estudar Web Service"
      }
    }
  }
}
```

Figura: Exemplo de Mutation

Query

- É um tipo de operação que permite recuperar dados.
- É semelhante a uma operação de leitura no REST.
- As consultas são tratadas no mesmo sistema de tipos e no mesmo endpoint que as mutações.

```
query {
  tasks {
    title
  }
}
```

```
{
  "data": {
    "tasks": [
      {
        "title": "Estudar"
      }
    ]
  }
}
```

Figura: Exemplo de Query

Material Complementar

- Documentação Oficial do GraphQL
- GraphQL: APIs for humans
- Apollo GraphQL
- How to GraphQL

Quem usa GraphQL?

- Repositório que lista algumas empresas brasileiras que utilizam GraphQL

Experimento 1

- Consultar a API GraphQL https://graphqlpokemon.favware.tech/v8
- Navegue pela documentação da API e teste os recursos disponíveis.

Experimento 2

Implementar uma aplicação frontend que permita consultar informações consultando uma API GraphQL.

- Utilizar uma biblioteca/framework de sua escolha para o desenvolvimento frontend (por exemplo, React, Vue, Angular, Python, VanillaJS, etc.).
- Utilizar a API GraphQL https://github.com/favware/graphql-pokemon ou
- Utilizar a API GraphQL https://github.com/trevorblades/countries ou
- Utilizar a API GraphQL https://rickandmortyapi.com/documentation ou
- Utilizar a API GraphQL https://studio.apollographql.com/public/SpaceX-pxxbxen/variant/current/home ou
- Utilizar a API GraphQL https://graphqlzero.almansi.me/ ou
- Outra API GraphQL de sua escolha.