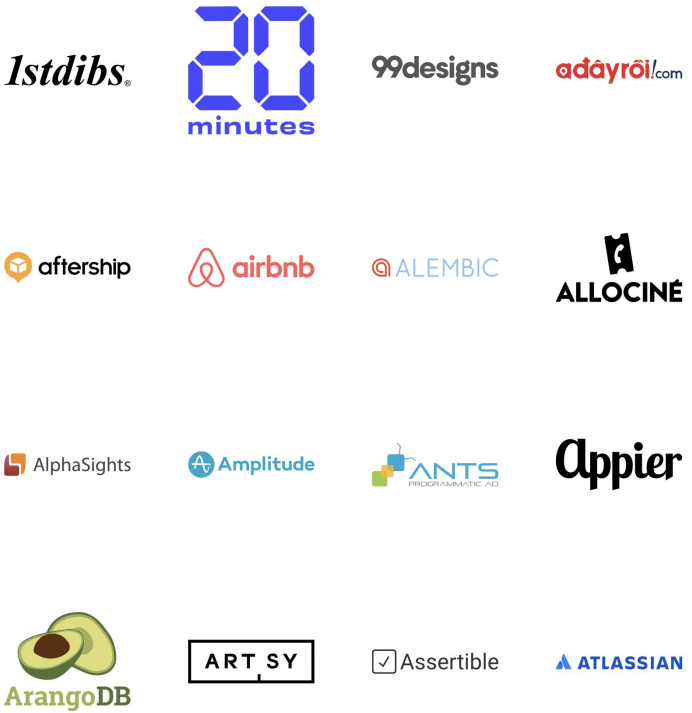
5 GraphQL vs REST

- ➔ REST và những hạn chế
- ➔ Giới thiệu GraphQL và Apollo
- ➔ Demo một vài Queries, Mutations
- ➔ Review Github Project UI

Tại sao?

- GraphQL đang là hype vài năm gần đây
- Rất nhiều công ty đã adopt GraphQL
- Vẫn khá là niche nên hiếm người có nhiều kinh nghiệm với nó



Nguồn: https://graphql.org/users/

Nói về REST

REST

- Viết tắt của Representational State Transfer
- Ra đời năm 2000
- Là một tiêu chuẩn để thiết kế API
- Map 1 URI với 1 resource nhất định

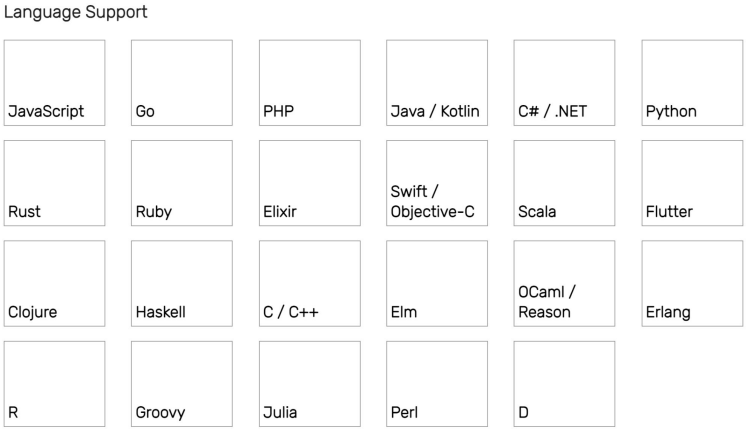
Hạn chế của REST

- Mỗi URI map với 1 response => Không linh hoạt cho nhiều trường hợp
- Khó mở rộng hoặc sử dụng lại 1 API
- Khó để có typed response
- Khó để quản lý version

GraphQL

GraphQL là gì

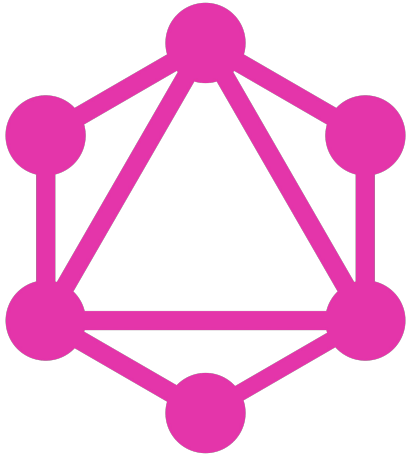
- Một “ngôn ngữ” query, một pattern hơn là một công nghệ
- Được hỗ trợ bởi nhiều ngôn ngữ
- Sinh ra để bù đắp cho những bất cập của REST
- Bắt nguồn từ Facebook năm 2012 và ra cộng đồng năm 2015



Nguồn: https://graphql.org/code/

GraphQL

- Thường sử dụng 1 method (thường là POST) cho mọi request
- Luôn trả về code 200 và error nằm chung trong response
- Chỉ có 1 URI duy nhất, thường thì kết thúc bằng “/graphql”
- Có 2 khái niệm chính:
  - Query: để GET dữ liệu
  - Mutation: để PATCH, PUT, DELETE (mutate: thay đổi) dữ liệu



Query

- Là cách để lấy dữ liệu về
- Được phép chọn field mình cần để không phải lấy dư data

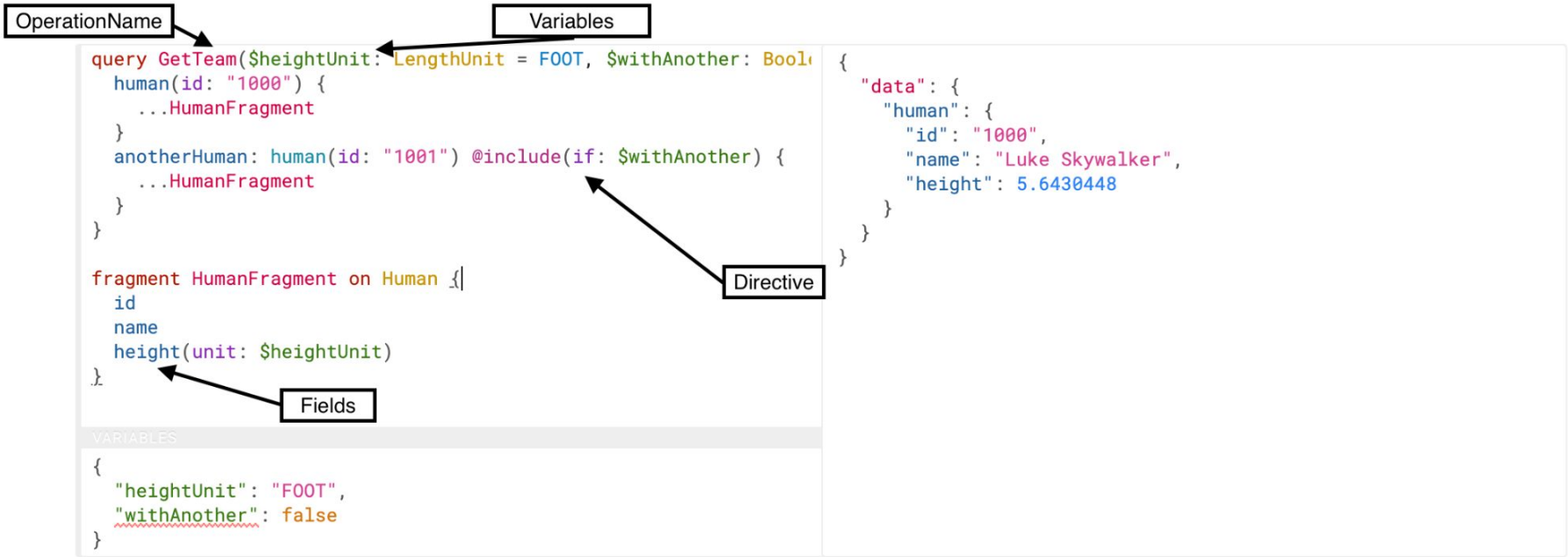
```
query HeroNameAndFriends($episode: Episode)
  hero(episode: $episode) {
    name
    friends {
      name
    }
  }
}
```

VARIABLES

```
{
  "episode": "JEDI"
}
```

```
{
  "data": {
    "hero": {
      "name": "R2-D2",
      "friends": [
        {
          "name": "Luke Skywalker"
        },
        {
          "name": "Han Solo"
        },
        {
          "name": "Leia Organa"
        }
```

Query (tiếp)


OperationName
Variables
query GetTeam($heightUnit: LengthUnit = FOOT, $withAnother: Bool
human(id: "1000") {
...HumanFragment
}
anotherHuman: human(id: "1001") @include(if: $withAnother) {
...HumanFragment
}
}
fragment HumanFragment on Human {
id
name
height(unit: $heightUnit)
}
Directive
Fields
VARIABLES
{
"heightUnit": "FOOT",
"withAnother": false
}
{
"data": {
"human": {
"id": "1000",
"name": "Luke Skywalker",
"height": 5.6430448
}
}
}

Mutation

- Tương tự như query
- Là cách để thay đổi dữ liệu

```
mutation CreateReviewForEpisode($ep: Episode!, $review: ReviewInp
  createReview(episode: $ep, review: $review) {
    stars
    commentary
  }
}
```

VARIABLES

```
{
  "ep": "JEDI",
  "review": {
    "stars": 5,
    "commentary": "This is a great movie!"
  }
}
```

```
{
  "data": {
    "createReview": {
      "stars": 5,
      "commentary": "This is a great movie!"
    }
  }
}
```

Các thư viện hỗ trợ GraphQL cho JS

- graphql-request
- urql
- Relay
- Apollo Client
- graphql-hooks

Apollo Client

- Apollo hỗ trợ cả về server lẫn client side
- Hệ thống caching mạnh mẽ
- Normalize và merge các objects tự động
- Thường được sử dụng cho các dự án lớn

Github GraphQL API

Github API

- REST: https://docs.github.com/en/rest
- GraphQL: https://docs.github.com/en/graphql

Tại sao Github sử dụng GraphQL

> GraphQL represents a massive leap forward for API development. Type safety, introspection, generated documentation, and predictable responses benefit both the maintainers and consumers of our platform. We're looking forward to our new era of a GraphQL-backed platform, and we hope that you do, too!

Theo: Github blog

Review Project UI

Github project UI

Review

- Review UI của từng page để tìm các phần UI có thể tái sử dụng.
- Phân chia thành các Container.
- Từ các Container phân chia thành các Component.

Cấu trúc Project

- Xây dựng Template cho Project.
- Phân chia các thư mục cho từng đối tượng (Page, Container, Styles, ...).
- Sử dụng các Library hỗ trợ (Build, Optimization, ...)

Implement Architecture

- Sử dụng các mô hình kiến trúc như Clean Architecture để dàng dàng maintain và phát triển Project.
- Dễ dàng chia sẻ với các thành viên trong team.

Bài tập

- Viết Query cho trang Pull Requests

Link tham khảo

Đọc vui:

- https://github.blog/2016-09-14-the-github-graphql-api/
- https://blog.api.rakuten.net/graphql-vs-rest/
- https://blog.logrocket.com/5-graphql-clients-for-javascript-and-node-js/