

GraphQL

Oriol Cuéllar Barrionuevo
David Valls Gonzalez-Roman
Salvador Grimalt

Introducción

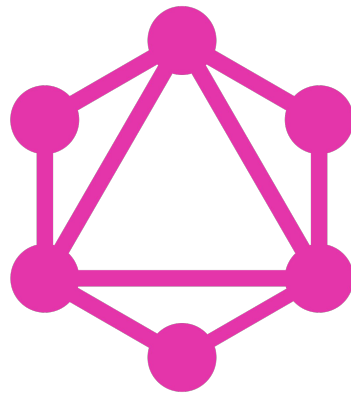
Lenguaje destinado a consultas.

Runtime para atenderles datos unos datos.

Proporciona una mayor velocidad y flexibilidad de desarrollo de APIs.

Desarrollado por Facebook en 2012 y hecho público en 2015, ha ido ganando popularidad entre las grandes compañías (Spotify, GitHub, Netflix,..).

Intuitivo y fácil de utilizar. Soluciona algunos problemas que tiene REST.



Versiones importantes (desde el punto de vista de Facebook)

2007: Apple presenta el iPhone. Competencia Safari-Google.

2010: Smartphones, Facebook comienza a utilizar estos navegadores. Utilizan API rest para crear una app nativa para iOS.

2011: API RESTful da problemas, necesidad nuevo llenguatge.

2012: Nace GraphQL.

2016: Versión estable más antigua. Github la comienza a utilizar.

2018: Se añade SDL. Lenguaje para a escribir esquemas de GraphQL.

271 VERSIONES DES DE FEBRERO 05, 2015.

Características principales

- Un solo EndPoint
- Fácil de actualizar
- Agrupar consultas
- GraphiQL IDE
- Schemas y Resolvers

Ejemplos

```
var schema = buildSchema(`
  type RandomDie {
    numSides: Int!
    rollOnce: Int!
    roll(numRolls: Int!): [Int]
  }

  type Query {
    getDie(numSides: Int): RandomDie
  }
`);
```

```
// This class implements the RandomDie GraphQL type
class RandomDie {
  constructor(numSides) {
    this.numSides = numSides;
  }

  rollOnce() {
    return 1 + Math.floor(Math.random() * this.numSides);
  }

  roll({numRolls}) {
    var output = [];
    for (var i = 0; i < numRolls; i++) {
      output.push(this.rollOnce());
    }
    return output;
  }
}
```

```
// The root provides the top-level API endpoints
var root = {
  getDie: ({numSides}) => {
    return new RandomDie(numSides || 6);
  }
}
```

Ejemplos

```
1 {  
2   getDie(numSides: 8){  
3     roll(numRolls: 6)  
4   }  
5  
6 }
```

```
{  
  "data": {  
    "getDie": {  
      "roll": [  
        8,  
        4,  
        2,  
        4,  
        7,  
        8  
      ]  
    }  
  }  
}
```

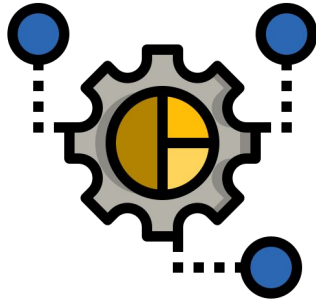
Puntos fuertes



Diseño



Agrupar consultas



Obtención precisa de datos



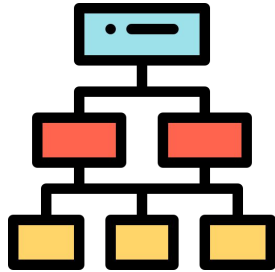
Rápido



Puntos fuertes



Estructura jerárquica



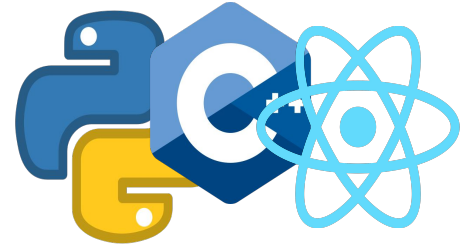
Lenguaje tipado

`int`
`#...`

Documentacion
autogenerada



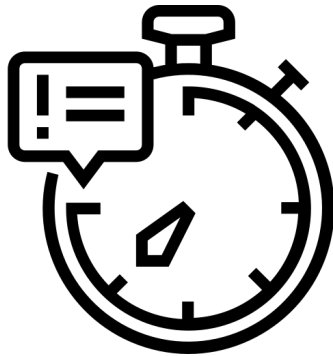
Multilenguaje



Puntos débiles



Velocidad de consultas



Lenguaje nuevo



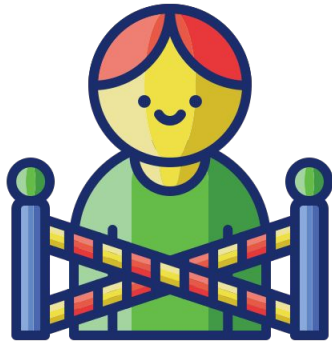
Más tarea



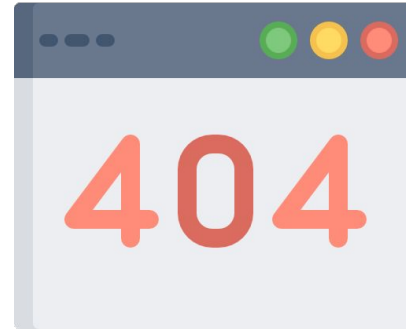
Puntos débiles



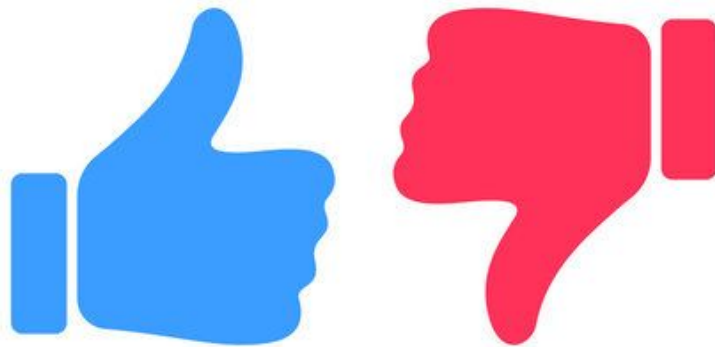
Comportamientos maliciosos



Manejo de errores



Conclusión



WebGrafia

<https://graphql.org/>

<https://en.wikipedia.org/wiki/GraphQL>

<https://www.howtographql.com/>

<https://www.redhat.com/es/topics/api/what-is-graphql>

<https://morioh.com/p/d531b40b22c6>

Repartición de tareas

- Preparar PowerPoint
Oriol, David, Salvador



- Revisar PowerPoint
Oriol, David



- Presentar a classe
Oriol, David

