

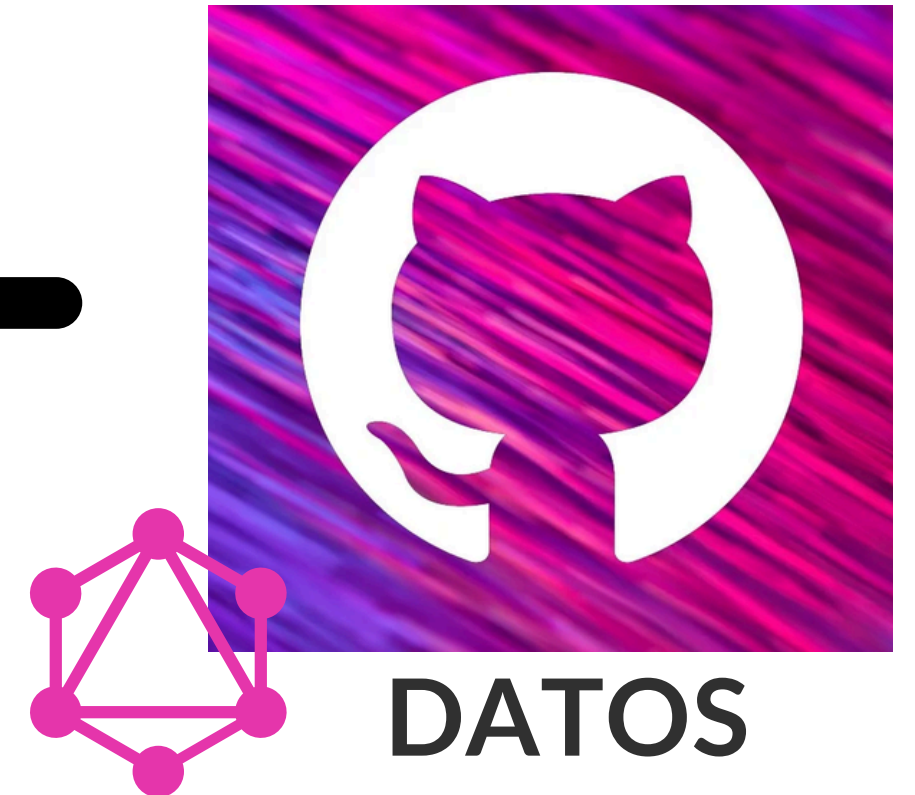
POTENCIANDO GITHUB-GQL CON NEO4J

Manuel Otero

Javier Fernández

1 INTRODUCCIÓN: CONTEXTO

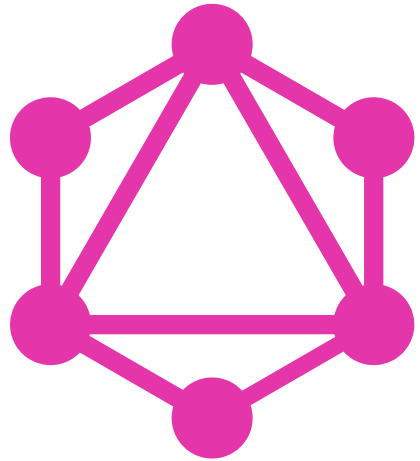
CONSULTAS
COMPLEJAS



DATOS
GITHUB

neo4j

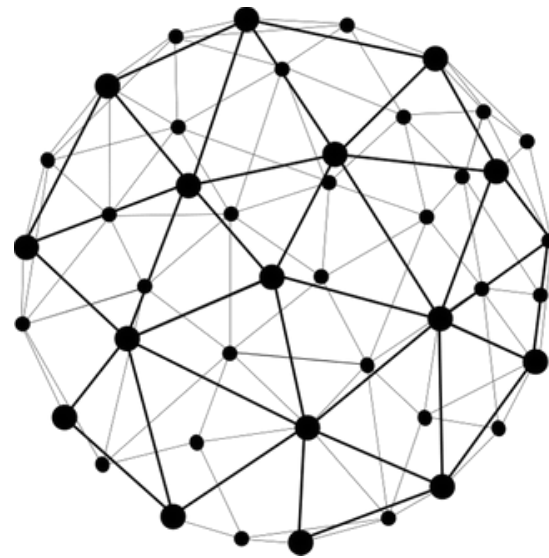
1 INTRODUCCIÓN: OBJETIVOS



USAR GQL



DESARROLLAR API

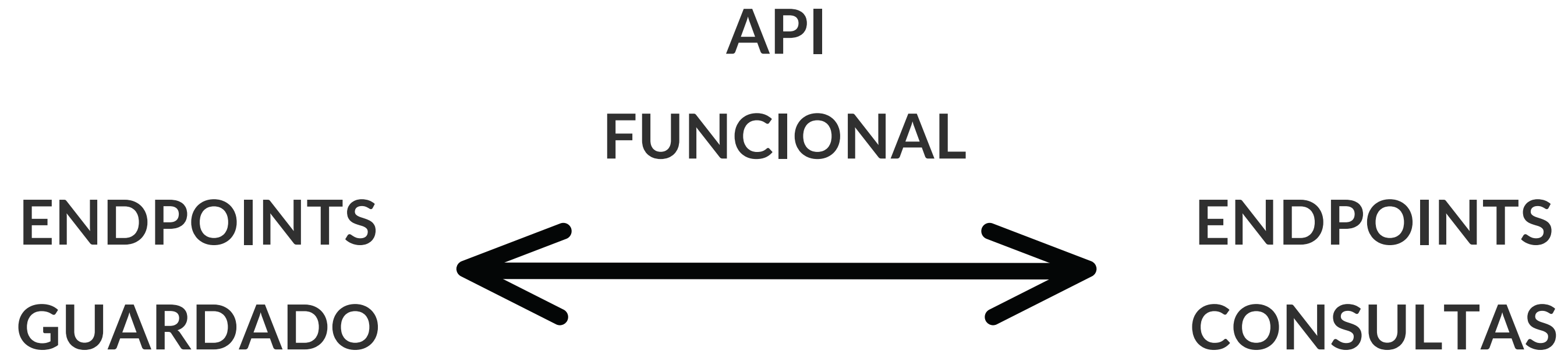


APROVECHAR
GRAFOS



CONOCER NEO4J

1 INTRODUCCIÓN: ALCANCE



DOCUMENTACIÓN



2 TECNOLOGÍAS: EXPRESS.JS

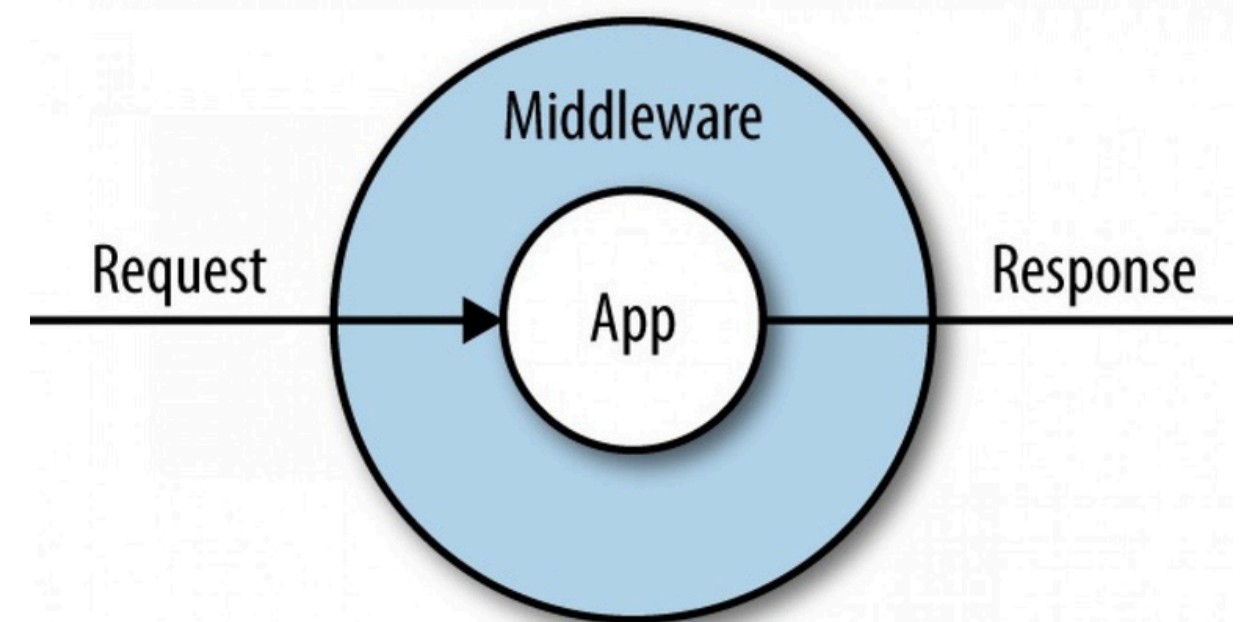
ENDPOINTS

/api/v1

Express

JS

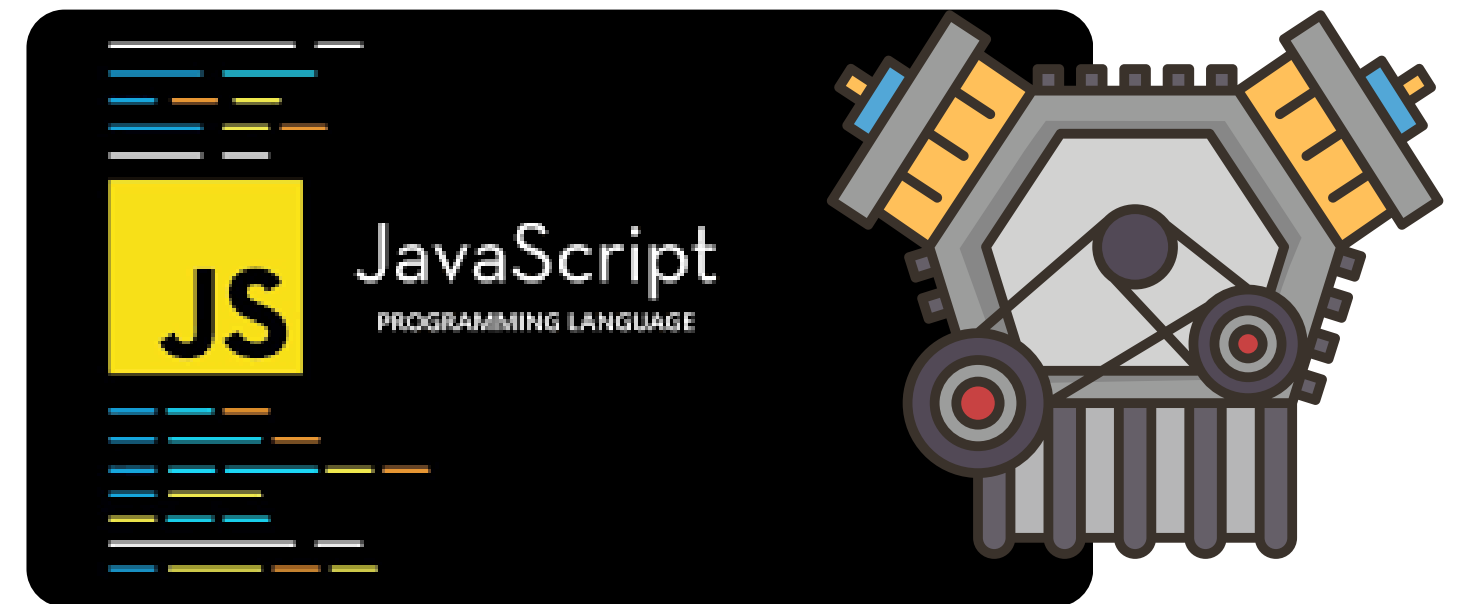
MIDDELWARES



2 TECNOLOGÍAS: NODE.JS

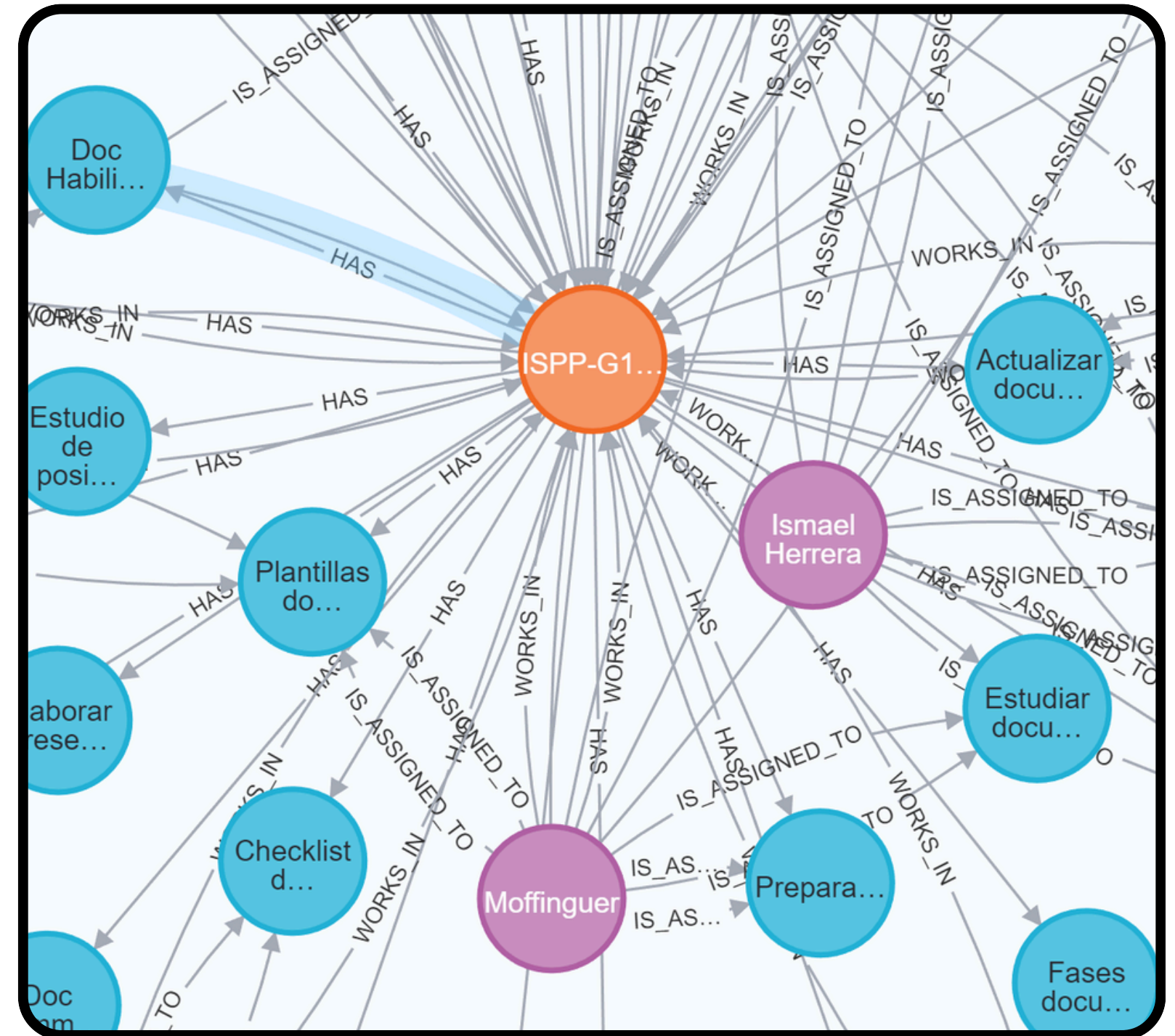


DEPENDENCIAS



MOTOR

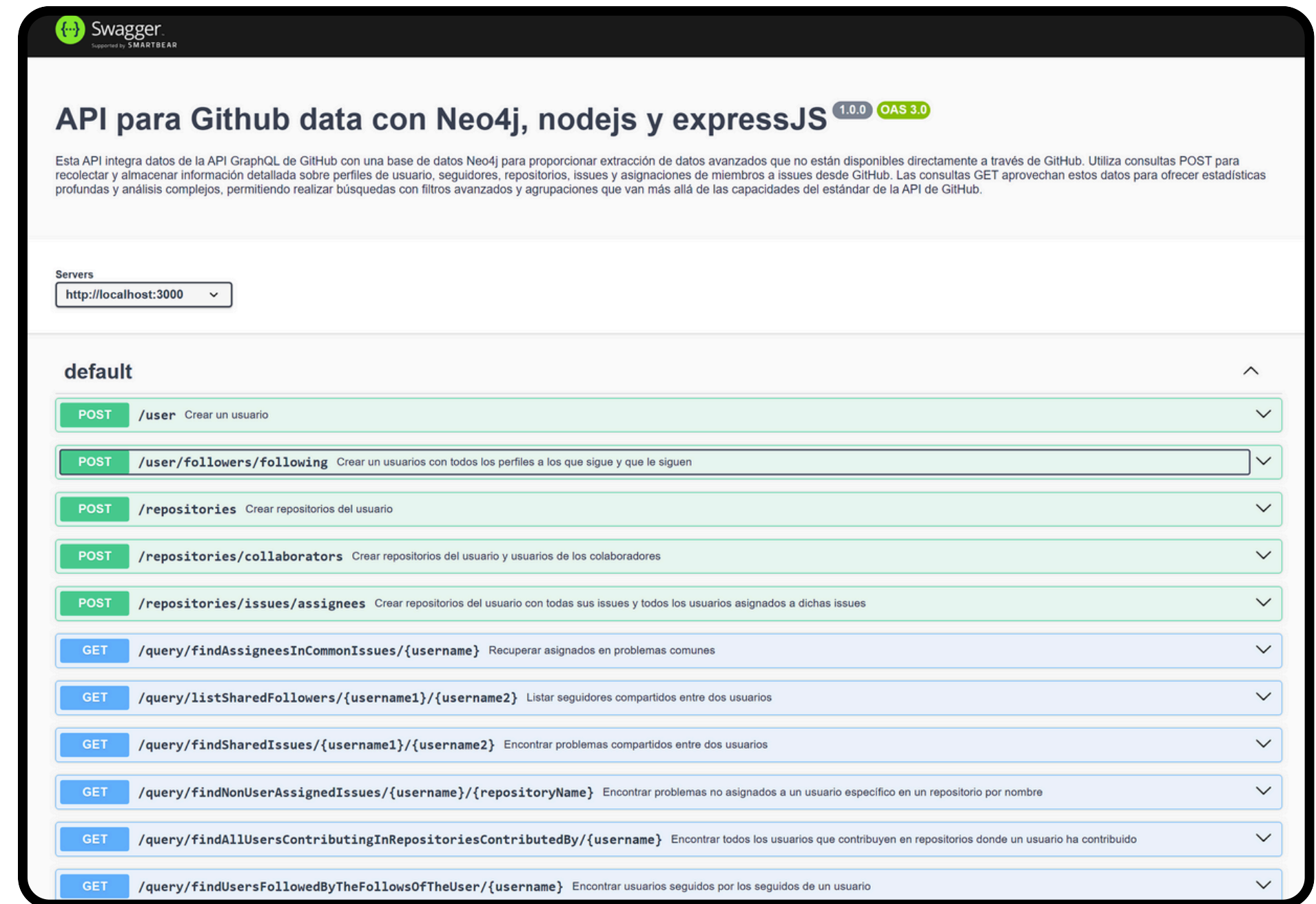
2 TECNOLOGÍAS: NEO4J



2 TECNOLOGÍAS: SWAGGER UI



Swagger UI
Supported by SMARTBEAR




2 TECNOLOGÍAS: GITHUB GQL API


GitHub GraphQL API

Start exploring GraphQL API queries using your account's data now. [Sign in with GitHub](#)





Heads up! GitHub's GraphQL Explorer makes use of your real, live, production data.



<untitled> <untitled> <untitled> x +



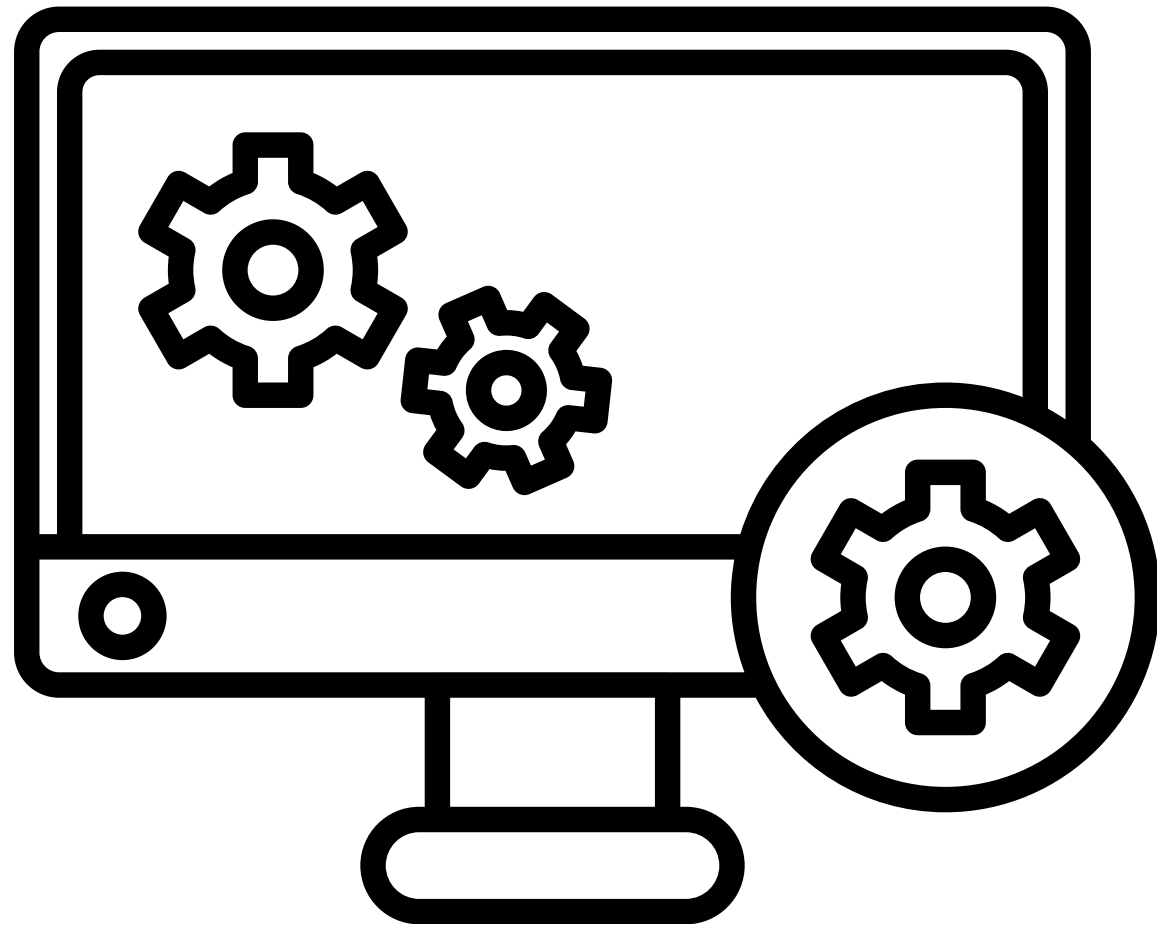
```
1 # Type queries here, and you will s
2 # aware of GitHub's current GraphQL
3 # syntax and validation errors high
4 # get you started with a simple que
5
6 query {
7   viewer {
8     login
9   }
10 }
```



GraphiQL

```
{
  "data": null,
  "errors": [
    {
      "message": "Log in to try the GraphQL
Explorer!"
    }
  ]
}
```

3 ARQUITECTURA: INTRODUCCIÓN



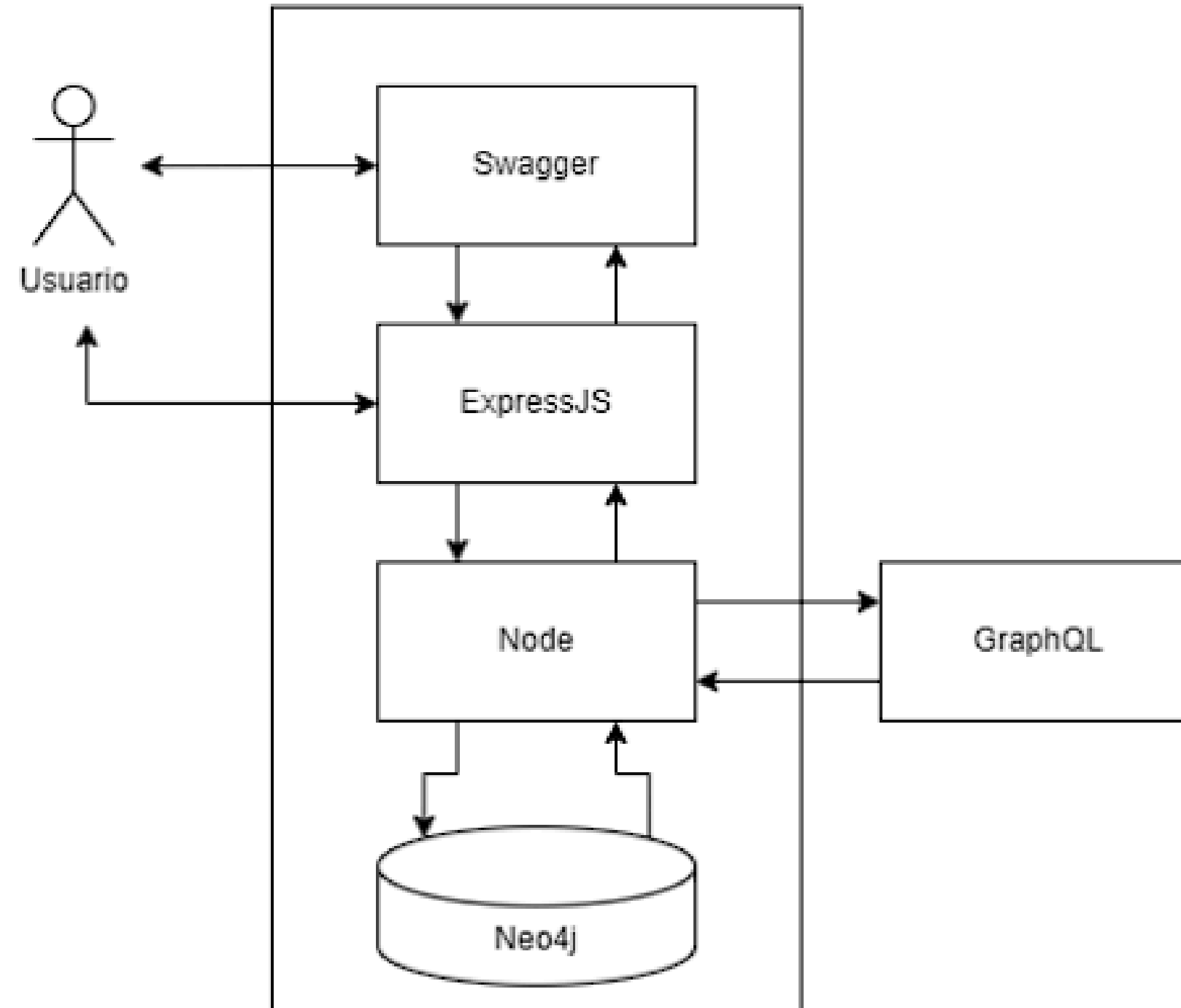
BASE DE DATOS



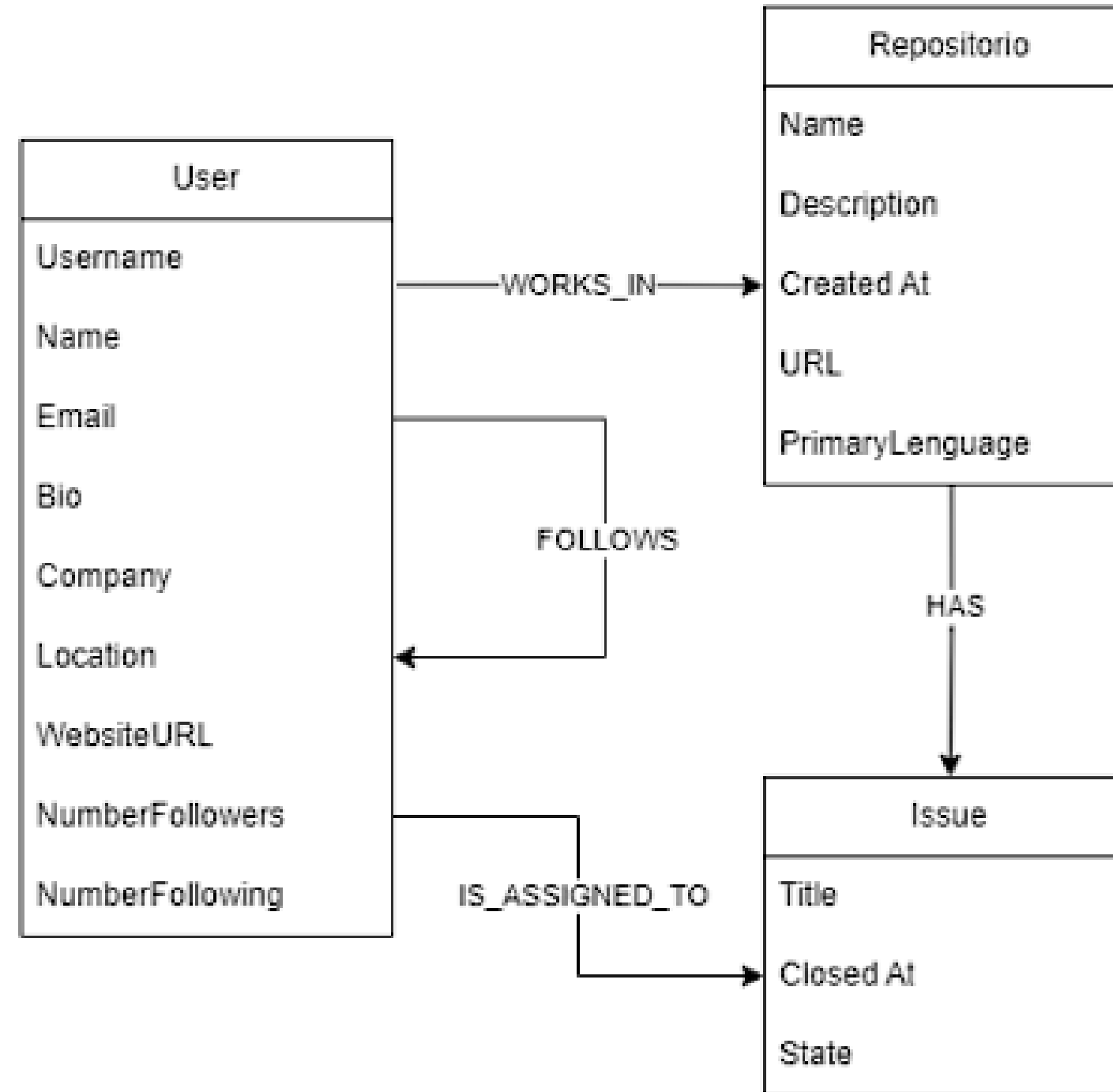
SERVIDOR



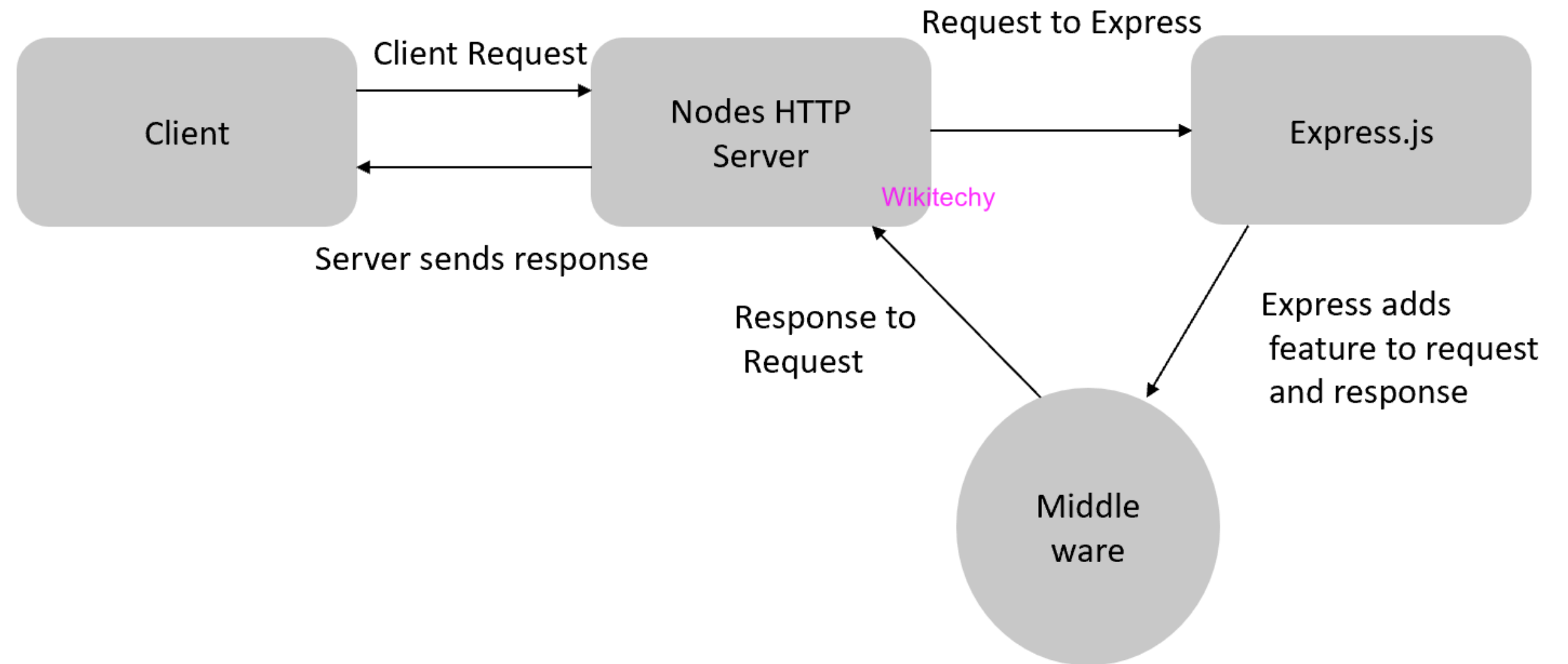
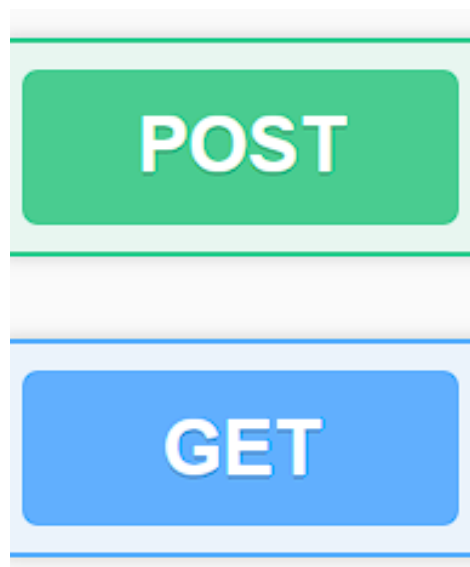
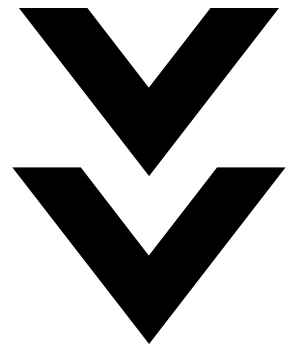
3 ARQUITECTURA: FLUJO



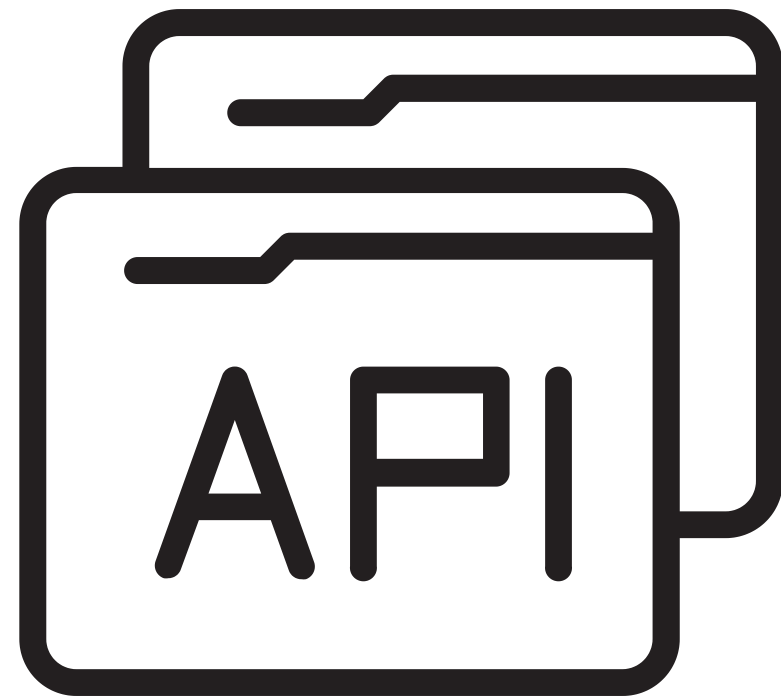
3 ARQUITECTURA: BASE DE DATOS



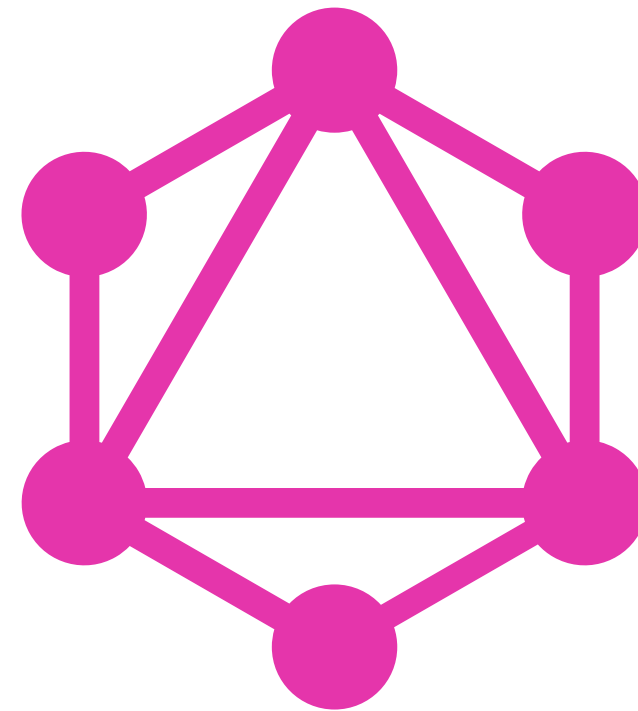
3 ARQUITECTURA: SERVIDOR



4 IMPLEMENTACIÓN: EXTRACCIÓN DE DATOS

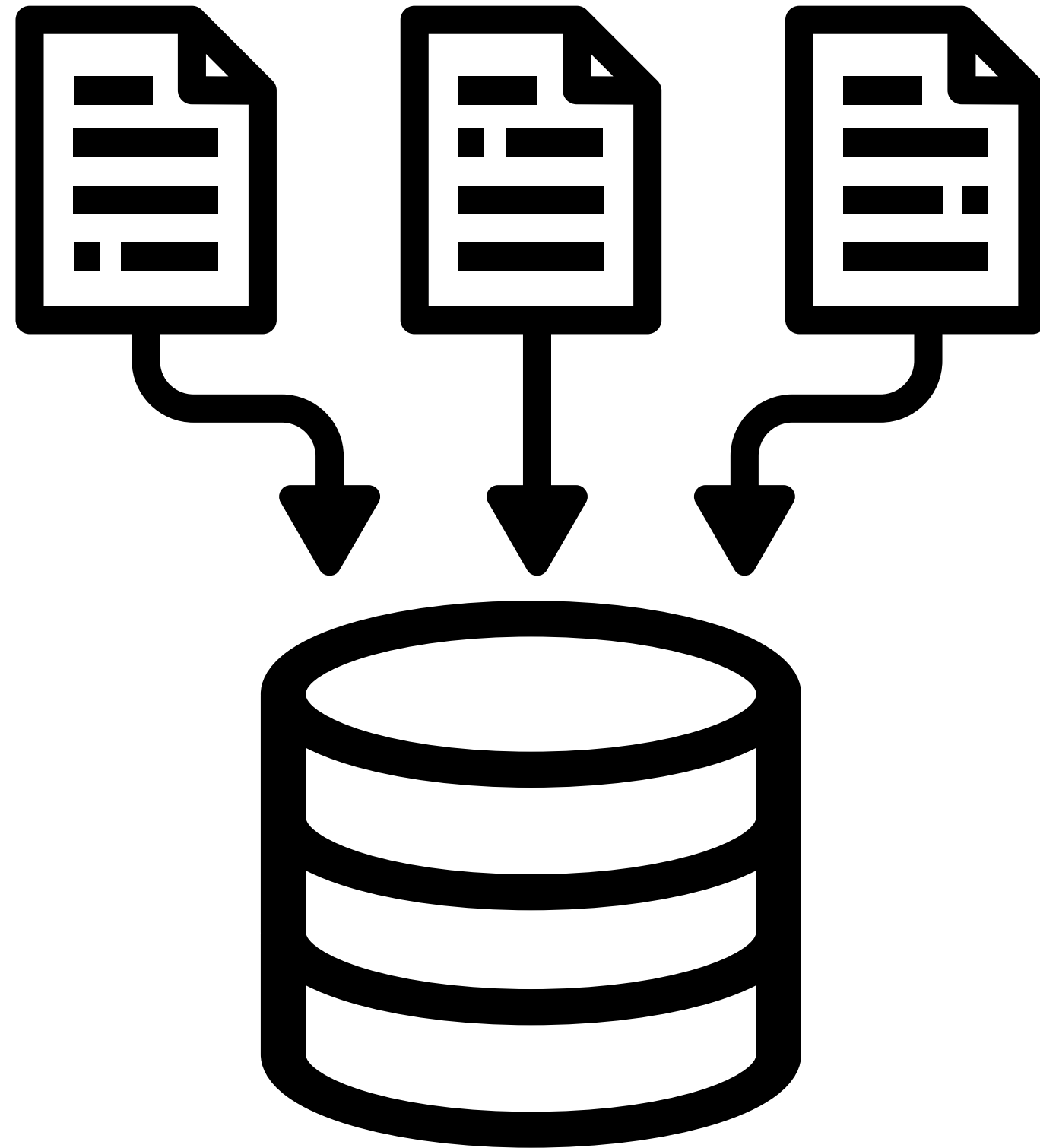


API

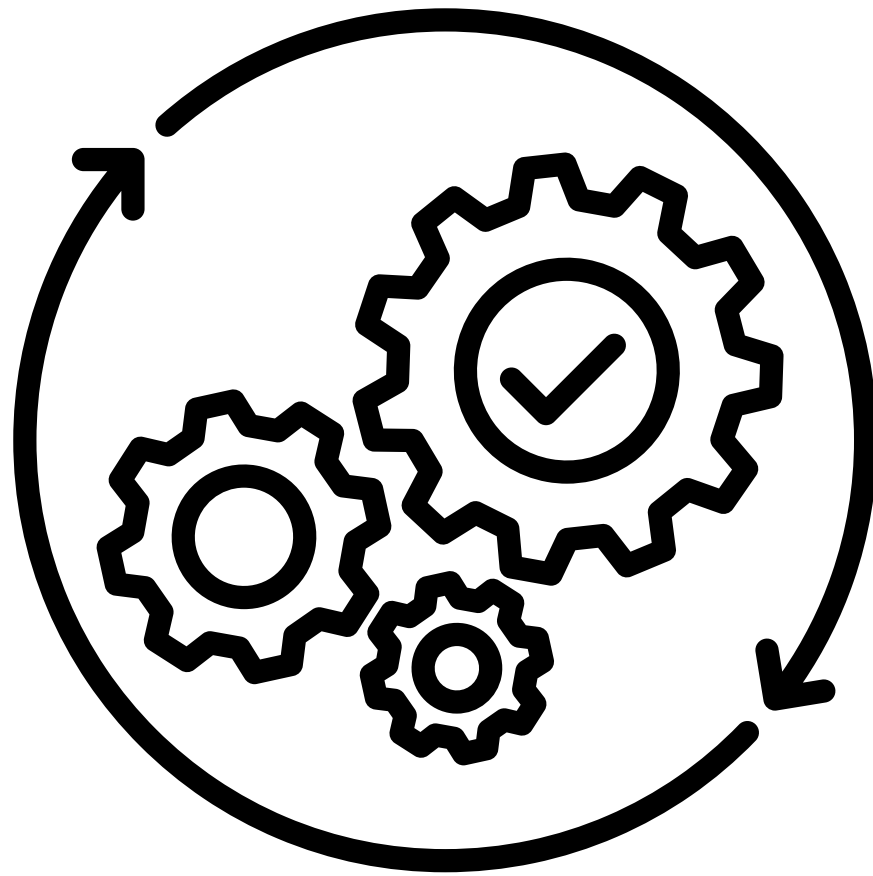


GITHUB GQL

4 IMPLEMENTACIÓN: POBLACION DE LA DB



4 IMPLEMENTACIÓN: PROCESAMIENTO



PROCESAMIENTO



ANÁLISIS

5 MANUAL DE INSTALACIÓN



6 MANUAL DE USUARIO

6.2. Ejemplos de consultas

A continuación, se proporcionan ejemplos de consultas para cada uno de los endpoints mencionados anteriormente. Estos ejemplos ilustran cómo se deben estructurar las solicitudes para interactuar efectivamente con la API.

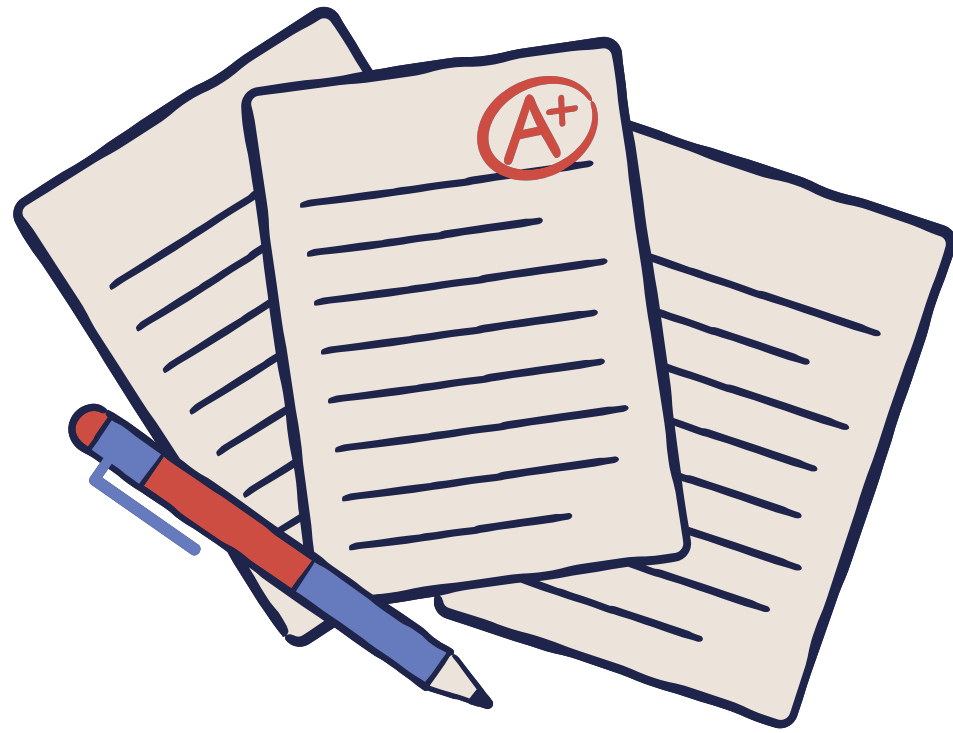
```
POST /user {  
    "username": "JaviFdez7"  
}
```

```
POST /user/followers/following {  
    "username": "JaviFdez7"  
}
```

```
POST /repositories {  
    "username": "JaviFdez7"  
}
```

```
POST /repositories/collaborators {  
    "username": "JaviFdez7"  
}
```

7 VALIDACIÓN



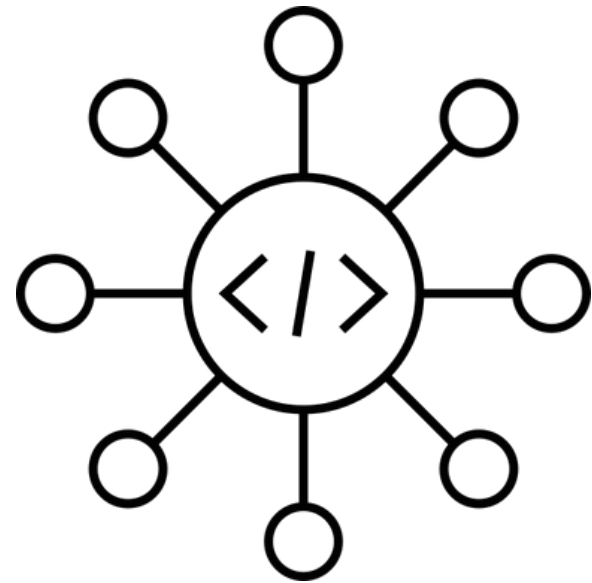
TESTS



RESULTADOS

DEMO

8 CONCLUSIONES



**ENFOQUE
RELACIONAL**



INVESTIGACIÓN



GRAPHQL & NEO4J

MUCHAS
GRACIAS