

Sprint 2 – Grupo 6

WHEELS

la app de viajes compartidos

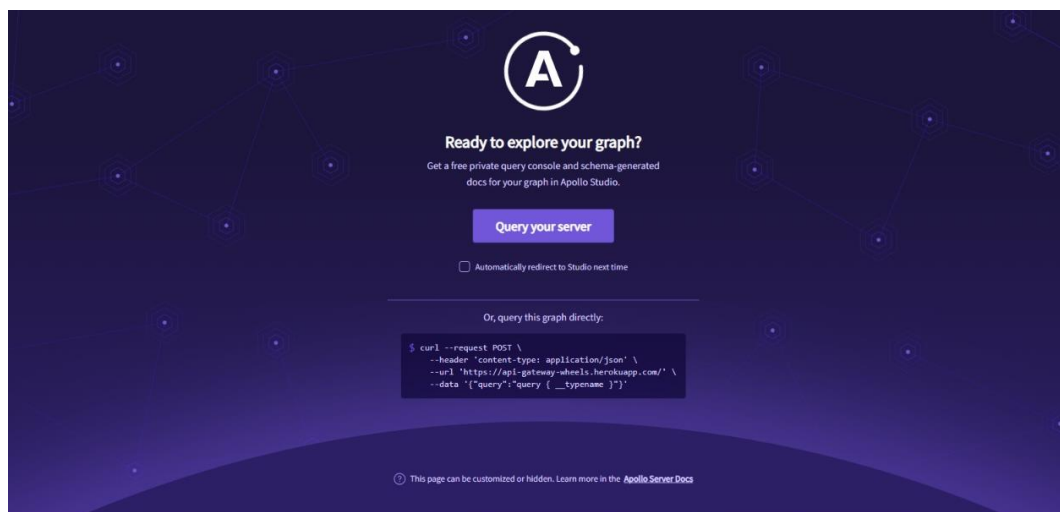
¿Qué es Wheels?

Wheels es un aplicativo web que tiene como finalidad facilitar a sus usuarios, la búsqueda y reserva de viajes compartidos a destinos locales o nacionales. Contamos con una plataforma intuitiva y dinámica, ofreciendo una experiencia única en su clase.


Creación y despliegue del microservicio de consumo API-REST

Para el sprint cuatro se creo el microservicio de consumos API-REST el cual permite realizar consultas, que permiten realizar operaciones post, put, delete, de cada uno de los módulos que se plantearon al inicio del proyecto, consultas guiadas por los id propios de cada módulo y sus propios atributos.

1. Asignación de tareas y flujo de trabajo en Jira <https://proyecto-wheels.atlassian.net/jira/software/projects/WGP/boards/1>
2. Despliegue del microservicio en GraphQL <https://api-gateway-wheels.herokuapp.com/>



- Repositorio en GitHub del microservicio <https://github.com/Grupo6-P53-MinTIC/appigate-wheels>

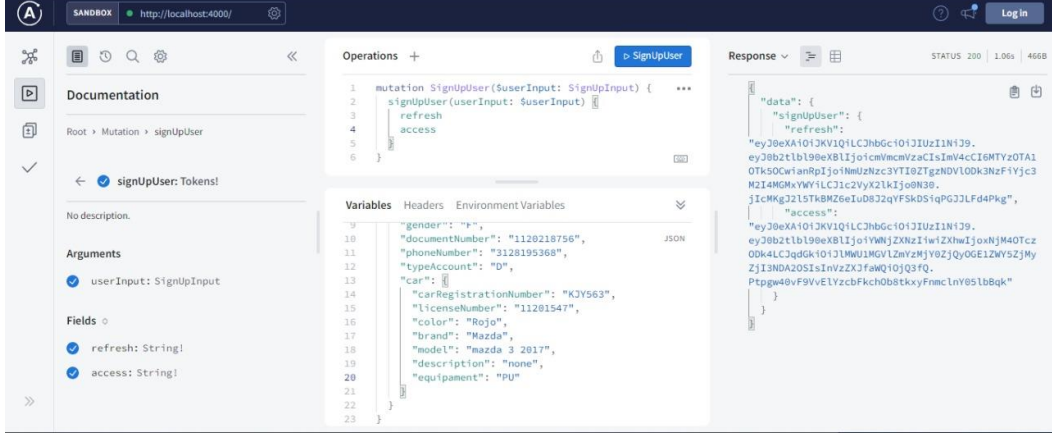
	BayronDavid Merge pull request #4 from Grupo6-P53-MinTIC/feat/adriana	c420c7b 5 days ago	🕒 17 commits
📁	src	Merge branch 'main' into feat/adriana	5 days ago
📄	.gitignore	Merge branch 'main' into feat/william	6 days ago
📄	Dockerfile	Despliegue - solo auth_ms	5 days ago
📄	package-lock.json	first commit	8 days ago
📄	package.json	first commit	8 days ago

- Consultas en Apollo

```

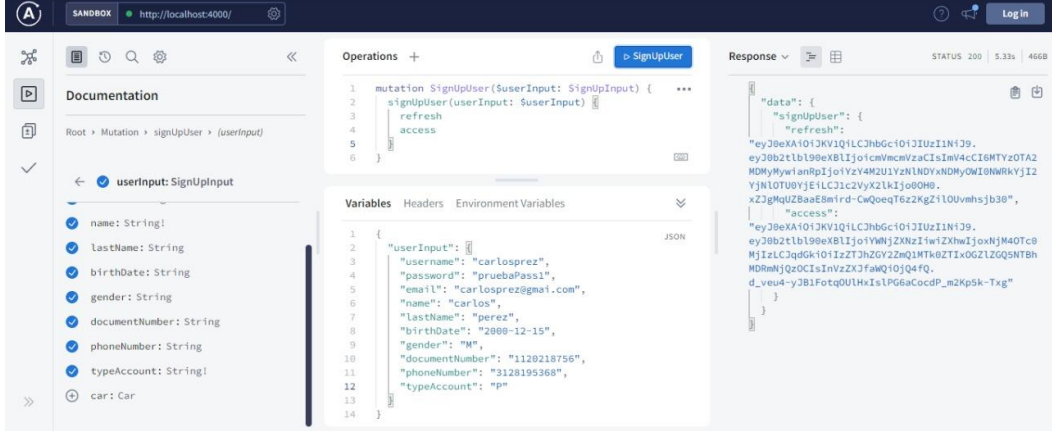
Crear un nuevo usuario
[9:29 a. m., 8/12/2021] +57 312 8192394: mutation SignUpUser($userInput:
SignUpInput) {
  signUpUser(userInput: $userInput) {
    refresh
    access
  }
}
{
  "userInput": {
    "username": "danireve",
    "password": "pruebaPass1",
    "email": "danireve@gmail.com",
    "name": "Daniela",
    "lastName": "Alejandra",
    "birthDate": "2000-12-15",
    "gender": "F",
    "documentNumber": "1120218756",
    "phoneNumber": "3128195368",
    "typeAccount": "D",
    "car": {
      "carRegistrationNumber": "KJY563",
      "licenseNumber": "11201547",
      "color": "Rojo",
      "brand": "Mazda",
      "model": "mazda 3 2017",
      "description": "none",
      "equipament": "PU"
    }
  }
}

```



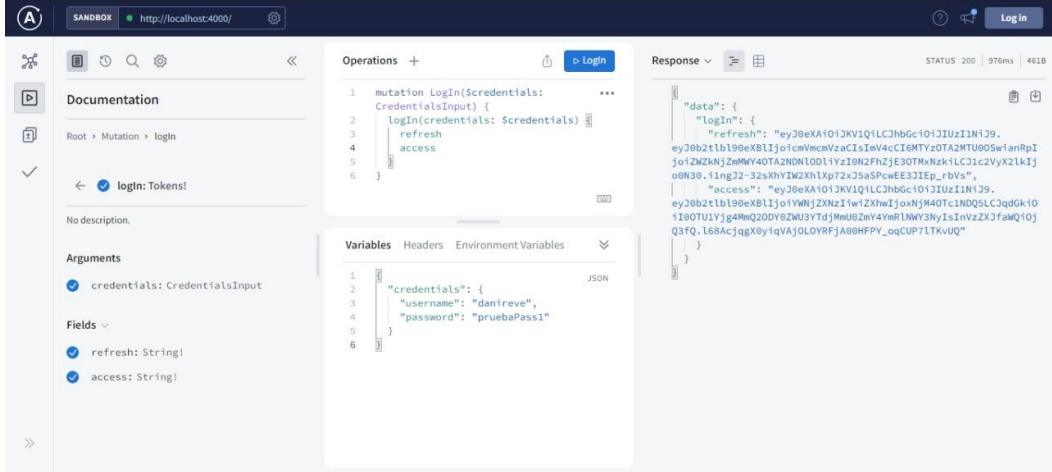
Obtener todos los usuarios

```
[9:32 a. m., 8/12/2021] +57 312 8192394: mutation SignUpUser($userInput:
SignUpInput) {
  signUpUser(userInput: $userInput) {
    refresh
    access
  }
}
{
  "userInput": {
    "username": "carlosprez",
    "password": "pruebaPass1",
    "email": "carlosprez@gmail.com",
    "name": "carlos",
    "lastName": "perez",
    "birthDate": "2000-12-15",
    "gender": "M",
    "documentNumber": "1120218756",
    "phoneNumber": "3128195368",
    "typeAccount": "P"
  }
}
```



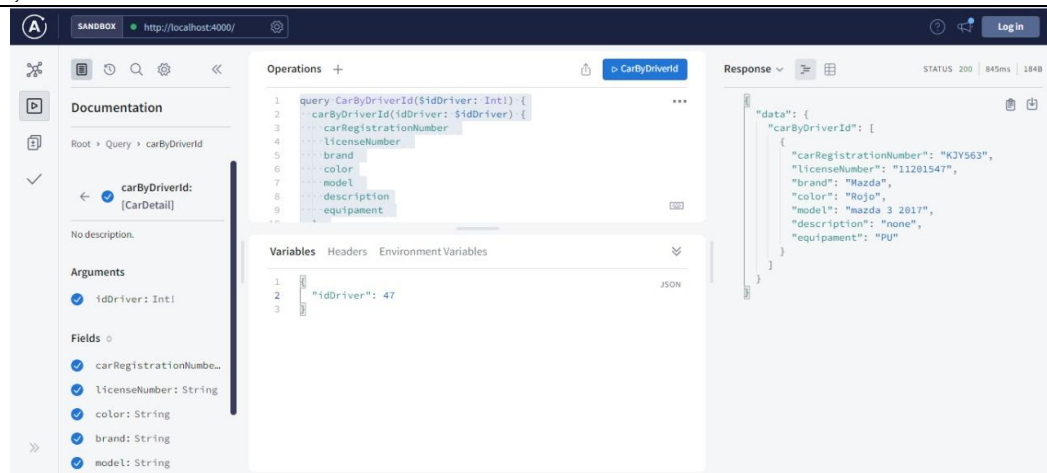
LogIn

```
[9:53 a. m., 8/12/2021] +57 312 8192394: mutation LogIn($credentials:
CredentialsInput) {
  logIn(credentials: $credentials) {
    refresh
    access
  }
}
{
  "credentials": {
    "username": "danireve",
    "password": "pruebaPass1"
  }
}
```



Obtener un carro por el id del conductor

```
query CarByDriverId($idDriver: Int!) {
  carByDriverId(idDriver: $idDriver) {
    carRegistrationNumber
    licenseNumber
    brand
    color
    model
    description
    equipament
  }
}
```



Consultar viaje por Id

```
query GetTravelById($getTravelByIdId: String!) {
  getTravelById(id: $getTravelByIdId) {
    idTravel
    idDriver
    nameDriver
    fromPlace
    toPlace
    passThrough
    published
    dateTravel
    seats
    price
  }
}
```

Documentation

Root > Query > getTravelById

← **getTravelById: Travel!**

- ✓ idTravel: String!
- ✓ idDriver: String!
- ✓ nameDriver: String!
- ✓ fromPlace: String!
- ✓ toPlace: String!
- ✓ passThrough: String!
- ✓ published: String!
- ✓ dateTravel: String!
- ✓ seats: String!
- ✓ price: String!

Operations

GetTravelById

```

3   idTravel
4   idDriver
5   nameDriver
6   fromPlace
7   toPlace
8   passThrough
9   published
10  dateTravel
11  seats
12  price
  
```

Variables

```

1   {
2     "getTravelById": null
3   }
  
```

Response

```

{
  "data": {
    "getTravelById": {
      "idTravel": "002",
      "idDriver": "102",
      "nameDriver": "Pedro"
    }
  }
}
  
```

STATUS 200 | 2.89s | 84B

Consultar viajes que pasan por un lugar

```

query GetTravelByToPlace($passThrough: String!) {
  getTravelByPassThrough(passThrough: $passThrough) {
    idTravel
    idDriver
    nameDriver
    fromPlace
    toPlace
    passThrough
    published
    dateTravel
    seats
    price
  }
}
  
```

Documentation

Root > Query > getTravelByPassThrough

← **getTravelByPassThrough: [Travel!]**

- ✓ idTravel: String!
- ✓ idDriver: String!
- ✓ nameDriver: String!
- ✓ fromPlace: String!
- ✓ toPlace: String!
- ✓ passThrough: String!
- ✓ published: String!
- ✓ dateTravel: String!
- ✓ seats: String!
- ✓ price: String!

Operations

GetTravelByToPlace

```

3   idTravel
4   idDriver
5   nameDriver
6   fromPlace
7   toPlace
8   passThrough
9   published
10  dateTravel
11  seats
12  price
  
```

Variables

```

1   {
2     "passThrough": "Garzon"
3   }
  
```

Response

```

{
  "data": {
    "getTravelByPassThrough": [
      {
        "idTravel": "003",
        "idDriver": "103",
        "nameDriver": "Maria",
        "fromPlace": "Netiva",
        "toPlace": "Pitalito",
        "passThrough": "Garzon",
        "published": true,
        "dateTravel": "2021-11-29T05:07:15.741+00:00",
        "seats": "4",
        "price": "15000"
      }
    ]
  }
}
  
```

STATUS 200 | 1.38s | 276B