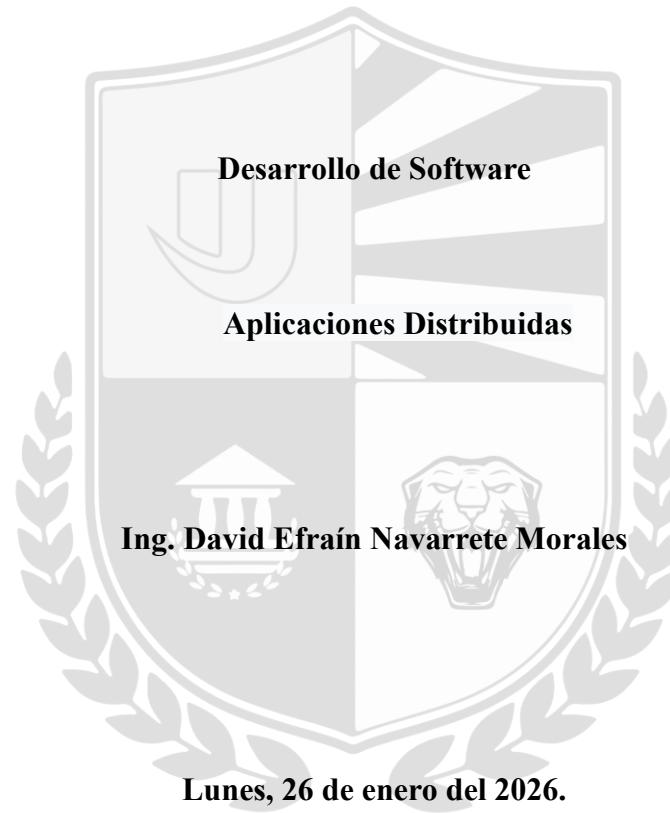




## TAREA TEÓRICA 5 U1

Luis A. Simbaña

Universitario Japón



- 📍 **SEDE MATRIZ QUITO** Pomasqui - Barrio Marieta de Veintimilla ☎ 022 356 368
- 📍 **CAMPUS QUITO SUR** La Magdalena, frente al colegio Paulo Sexto ☎ 022 103 555
- 📍 **SEDE SANTO DOMINGO** Calle Cuenca y Galápagos esquina ☎ 022 754 009
- 📍 **CAMPUS VÍA CHONE SANTO DOMINGO** Vía a Chone Km 2,1/2 ☎ 022 754 009
- 📍 **SEDE AMBATO** Av. Pedro Vásconez y Enrique Sánchez ☎ 099 400 4839



## Desarrollo de un microservicio en Spring Boot

Primero ejecute el backend con:

```
mvn spring-boot:run
```

Mi proyecto quedó activo en esta ruta:

<http://localhost:8080>

Cree un Representante en Postman con:

**POST**

<http://localhost:8080/api/representantes>

**Body (JSON):**

```
{  
    "nombre": "Luis",  
    "telefono": "098132383"  
}
```

Esto guarda un representante en la base de datos y generó su id.

Cree una Ruta, luego probe:

**POST**

<http://localhost:8080/api/rutas>



- 📍 **SEDE MATRIZ QUITO** Pomasqui - Barrio Marieta de Veintimilla ☎ 022 356 368
- 📍 **CAMPUS QUITO SUR** La Magdalena, frente al colegio Paulo Sexto ☎ 022 103 555
- 📍 **SEDE SANTO DOMINGO** Calle Cuenca y Galápagos esquina ☎ 022 754 009
- 📍 **CAMPUS VÍA CHONE SANTO DOMINGO** Vía a Chone Km 2,1/2 ☎ 022 754 009
- 📍 **SEDE AMBATO** Av. Pedro Vásconez y Enrique Sánchez ☎ 099 400 4839

**Body:**

```
{
```

```
    "nombre": "Ruta Centro"
```

```
}
```

Cree un estudiante con relaciones, finalmente probe:

**POST**

<http://localhost:8080/api/estudiantes>

**Body:**

```
{
```

```
    "nombre": "Luis",
```

```
    "direccion": "Latacunga",
```

```
    "representante": { "id": 1 },
```

```
    "ruta": { "id": 1 }
```

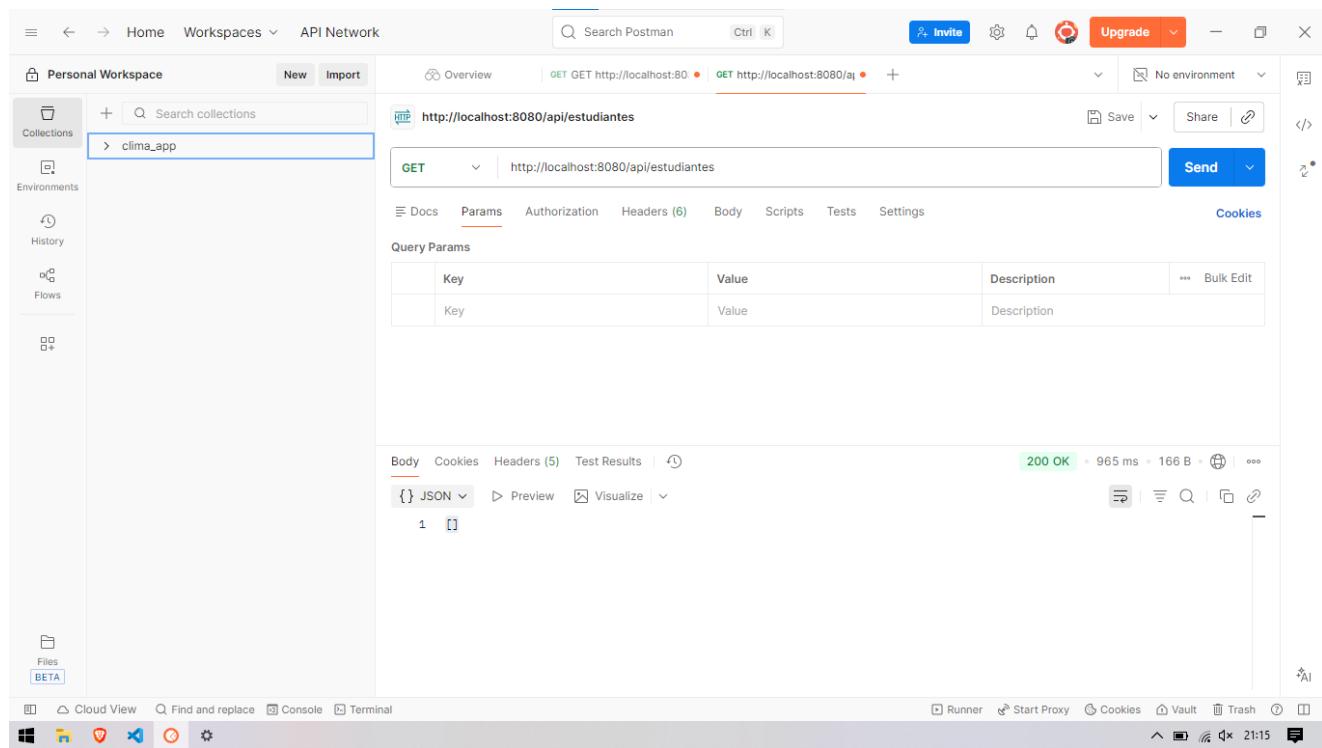
```
}
```

**Aquí use los IDs ya creados para relacionar:**

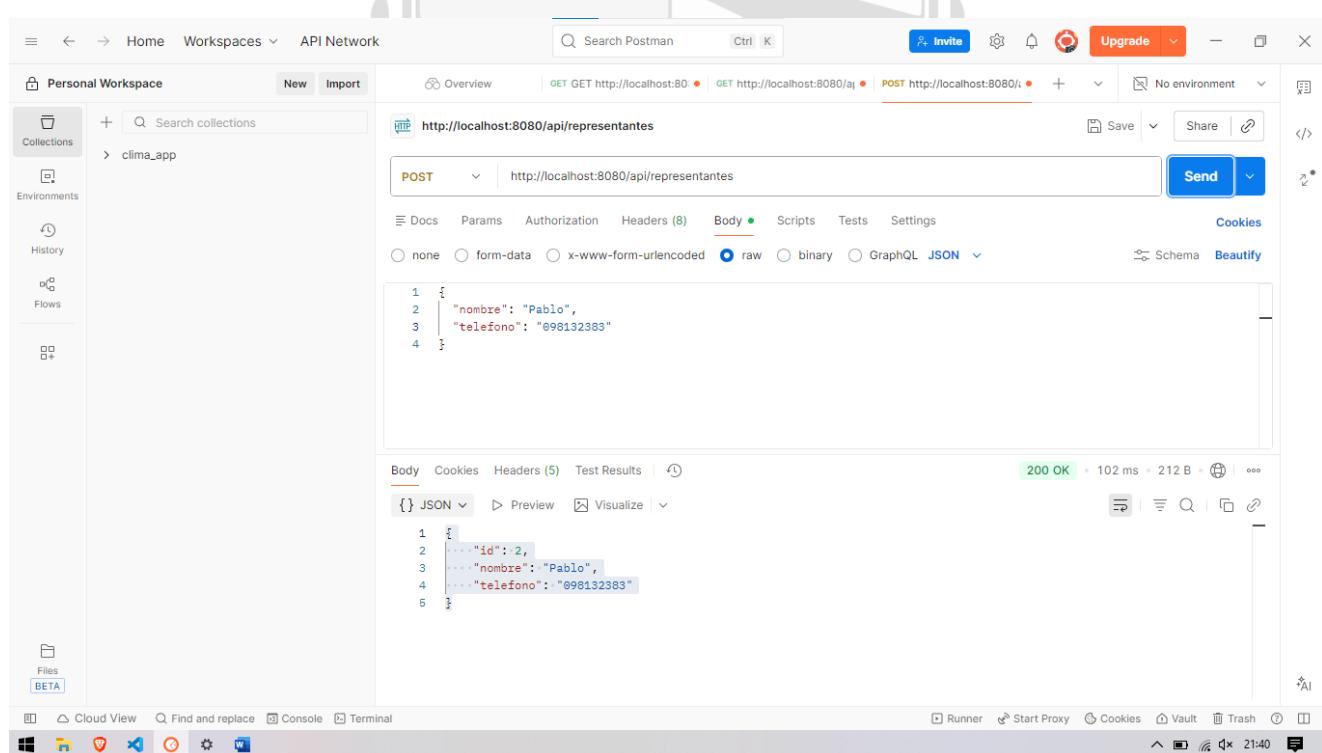
- Estudiante → Representante
- Estudiante → Ruta

Y el backend guardó todo correctamente.





The screenshot shows the Postman interface with a successful GET request to `http://localhost:8080/api/estudiantes`. The response status is `200 OK` with a duration of `965 ms` and a size of `166 B`. The response body is empty, indicated by the number `1` and an empty JSON object `[]`.



The screenshot shows the Postman interface with a successful POST request to `http://localhost:8080/api/representantes`. The response status is `200 OK` with a duration of `102 ms` and a size of `212 B`. The response body contains a JSON object with fields `id`, `nombre`, and `telefono`.

```

1 {
2   "nombre": "Pablo",
3   "telefono": "098132383"
4 }
      
```

```

1 {
2   ...
3   "id": 2,
4   ...
5 }
      
```



Personal Workspace

HTTP <http://localhost:8080/api/rutas>

POST http://localhost:8080/api/rutas

Body (raw)

```
1 {
2   "nombre": "Ruta Centro"
3 }
4 
```

200 OK

{} JSON

```
1 {
2   "id": 1,
3   "nombre": "Ruta Centro"
4 }
```

Cloud View Find and replace Console Terminal

Personal Workspace

HTTP <http://localhost:8080/api/estudiantes>

POST http://localhost:8080/api/estudiantes

Body (raw)

```
1 {
2   "nombre": "Luis",
3   "direccion": "Latacunga",
4   "representante": { "id": 1 },
5   "ruta": { "id": 1 }
6 }
7 
```

200 OK

{} JSON

```
1 {
2   "direccion": "Latacunga",
3   "id": 6,
4   "nombre": "Luis",
5   "representante": {
6     "id": 1,
7     "nombre": null,
8     "telefono": null
9   },
10   "ruta": { 
```

Cloud View Find and replace Console Terminal



