

**Caso de Estudio (Informe Etapa 2): Microservicios de “EduTech Innovators SPA”**

**Alumnos:**

Franco Pisani

Vicente Riquelme

Ignacio López

**Docente:**

Eduardo Baeza

**Asignatura:**

Desarrollo Fullstack I

26-05-2025

**ÍNDICE**

**1 .Introducción**

**2. Diagrama de Arquitectura de Microservicios**

**3. Estructura del Proyecto**

3.1. Dependencias

3.2. Componentes implemetandos

**4. Base de Datos**

4.1. Motor

4.2. Estructura

**5. Implementación de los Servicios**

**6. Git / Github**

6.1. Comandos utilizados

**7. Conclusión**

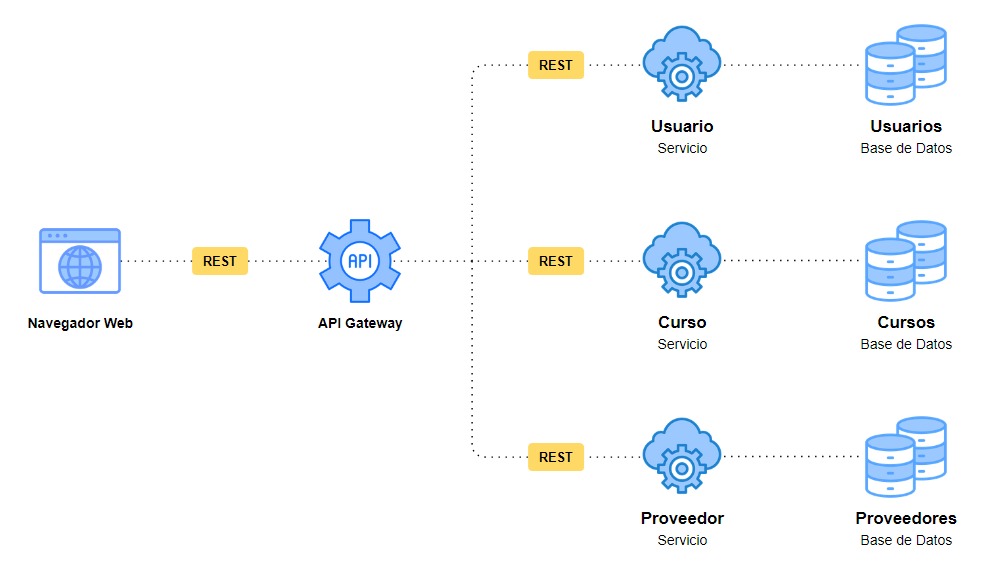
**1. INTRODUCCIÓN**

En este trabajo, correspondiente a la segunda etapa de la experiencia de aprendizaje, se desarrolla la implementación de microservicios siguiendo una arquitectura basada en API Full REST. A partir del caso trabajado en la etapa anterior, el equipo ha continuado con el diseño e implementación del sistema, enfocándose en tres servicios específicos, iniciando por el servicio de gestión de usuarios.

Este trabajo tiene como propósito evidenciar la aplicación de conocimientos técnicos adquiridos sobre el framework Spring, el gestor de dependencias Maven, el control de versiones mediante Git, y la validación de servicios con herramientas como Postman. A través de esta evaluación, se busca no solo demostrar competencias técnicas en la construcción de microservicios, sino también la correcta aplicación de buenas prácticas de diseño y desarrollo de software modular, escalable y mantenible.

El informe detalla aspectos clave como la estructura del proyecto, la implementación de servicios, la interacción con la base de datos atraves de Postman, y la validación de las funcionalidades desarrolladas. Además, se incluye evidencia gráfica y técnica del trabajo realizado en equipo, lo que permitirá evaluar tanto el dominio de herramientas específicas como la capacidad de colaboración en un entorno de desarrollo ágil.

**2. DIAGRAMA DE ARQUITECTURA DE MICROSERVICIOS**

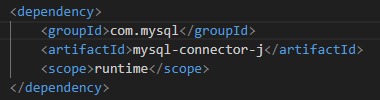
****

**3. ESTRUCTURA DEL PROYECTO**

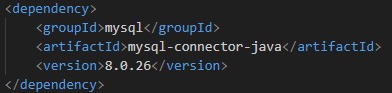
**3.1. Dependencias**

Despues de establecidas las dependencias a traves de Spring Initializr, se modifica pom.xml para utilizar la librería apropiada para MySQL Server.

**ANTES:**

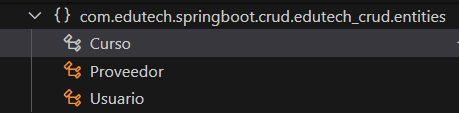


**CON LA MODIFICACIÓN:**

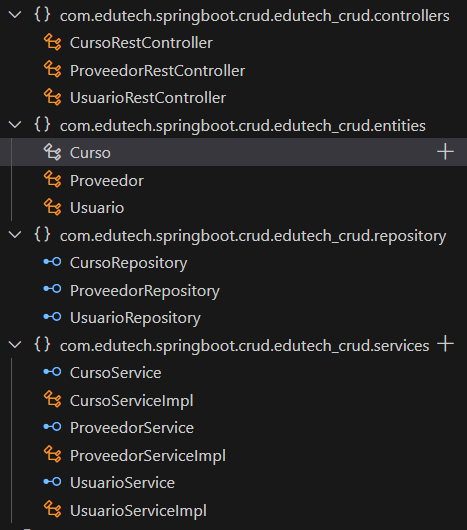


**3.2. Componentes Implementados**

Se implementaron 3 clases para este proyecto, las cuales estan dentro de su package llamado Entities, el que alberga todo el codigo de las clases, junto con los atributos. Estas clases son: **Usuario** (que almacenará informacion tanto de los alumnos como la de los instructores), **Curso** (que almacena la asignatura a la que corresponde cada curso, descripcion y el instructor que la imparte) y **Proveedor,** que almacena la información importante de estos, como el rut, la razón social, el giro y el email para contacto.



Ademas, se crearon 3 packages, **Controllers**, componente encargado de recibir, procesar y responder a las solicitudes del usuario o del cliente, **Services**, que contiene la logica de la aplicación y **Repository**, encargado de acceder a la base de datos. Estos existen para cada una de las clases, como se verá en la siguiente captura.



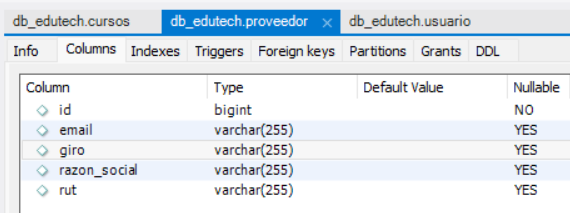
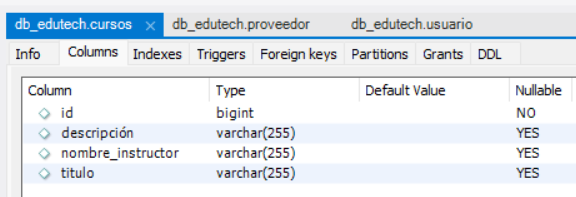
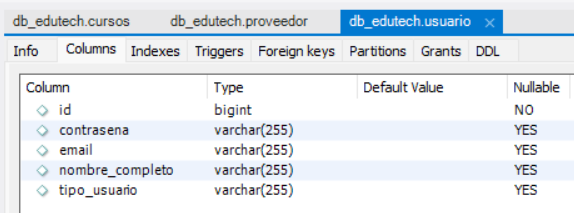
**4. BASE DE DATOS**

**4.1. Motor**

El motor de base de datos utilizado en el codigo es MySQL.

**4.2. Estructura**

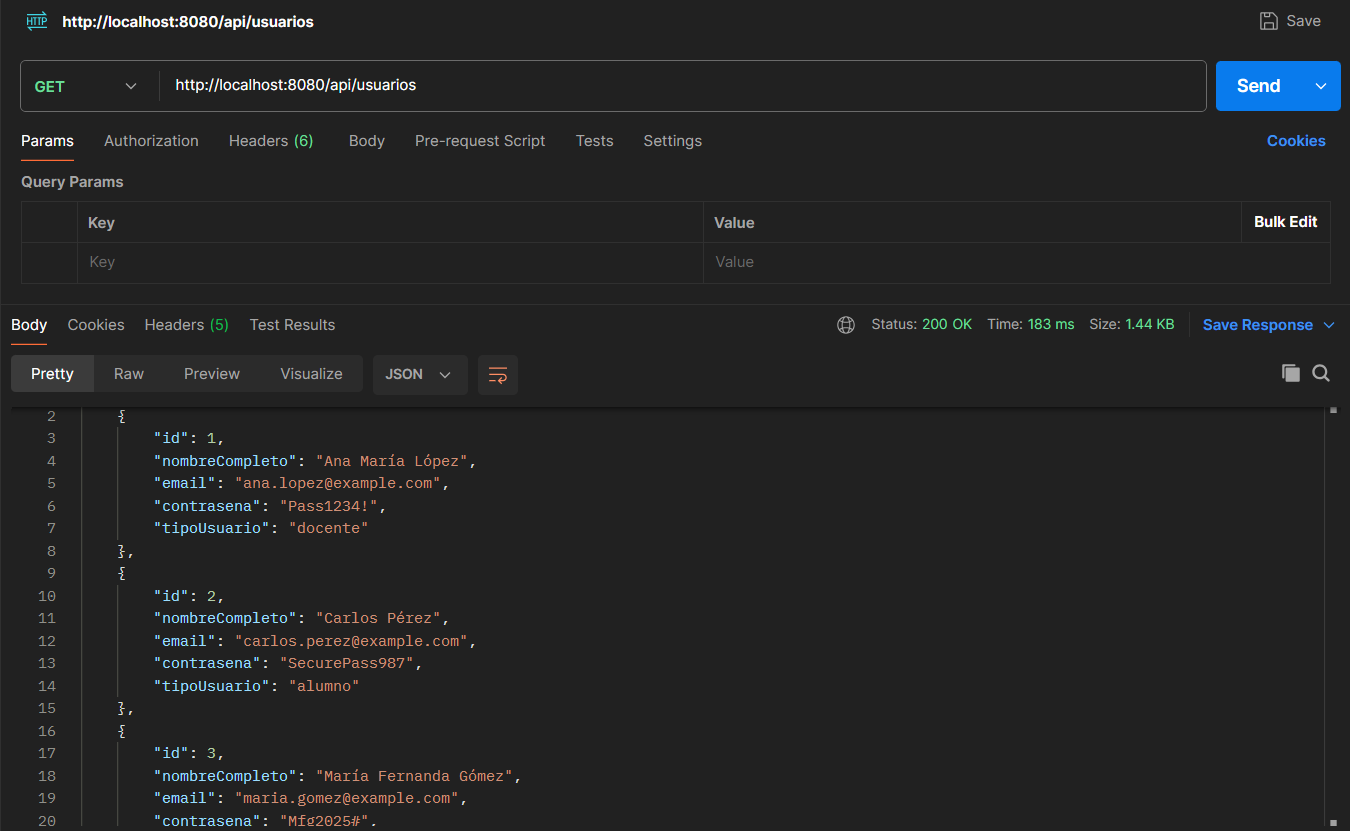
A continuacion se mostrará la estructura de la base de datos, junto con sus tablas y atributos.



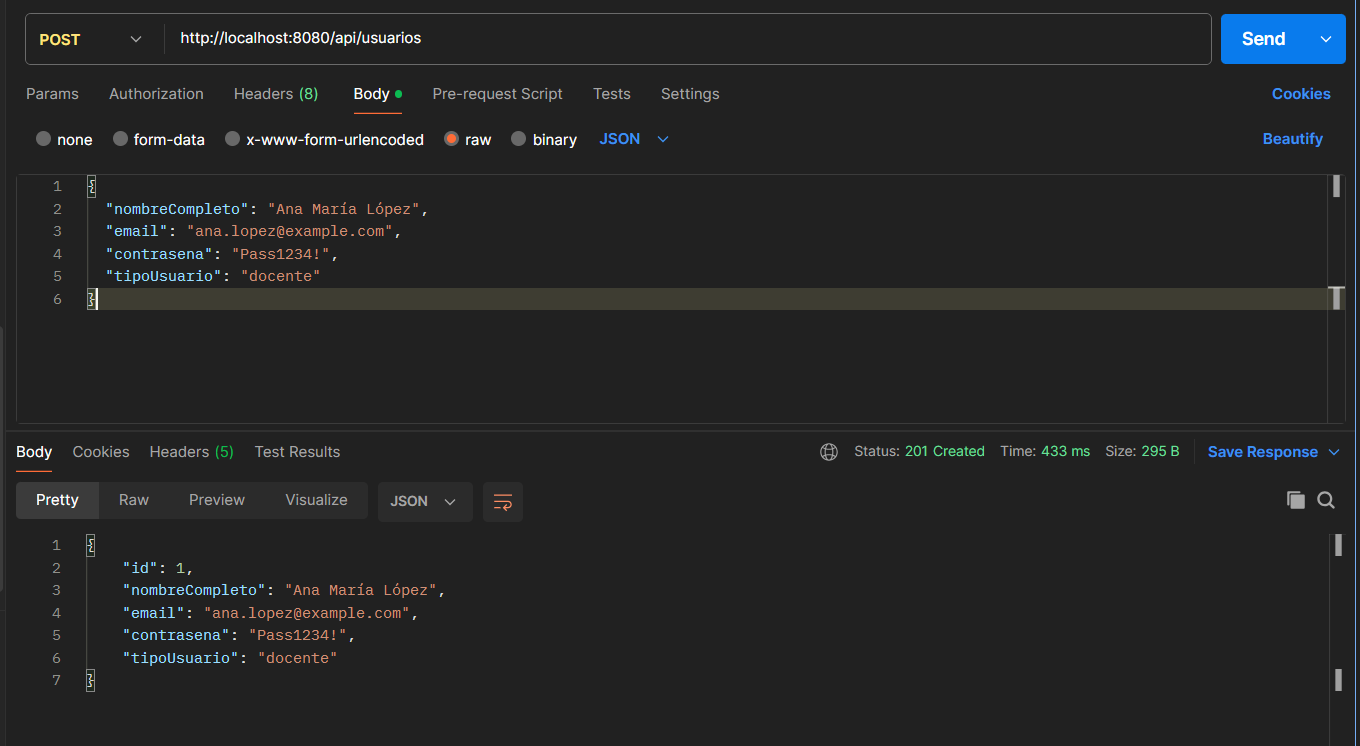
**5. IMPLEMENTACIÓN DE LOS SERVICIOS**

**USUARIO:**

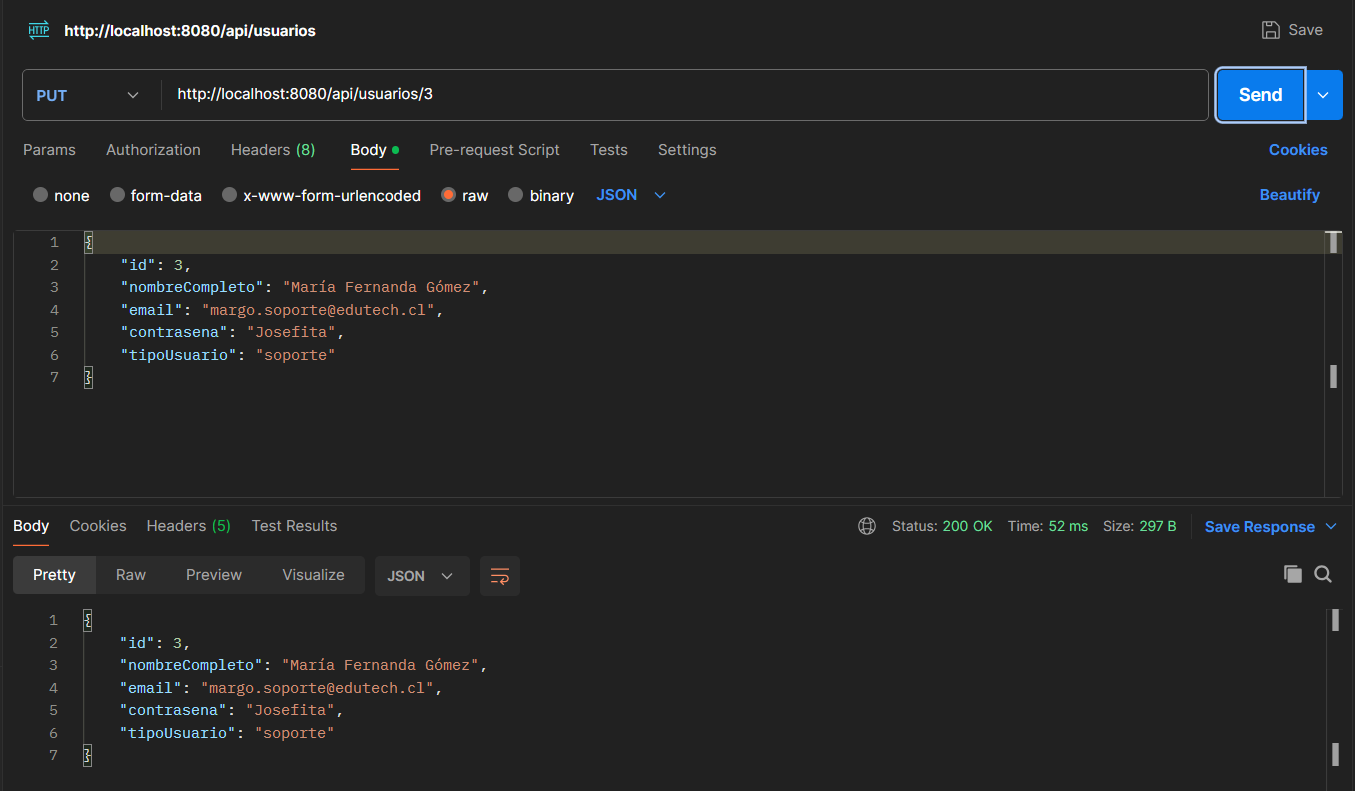
**GET**



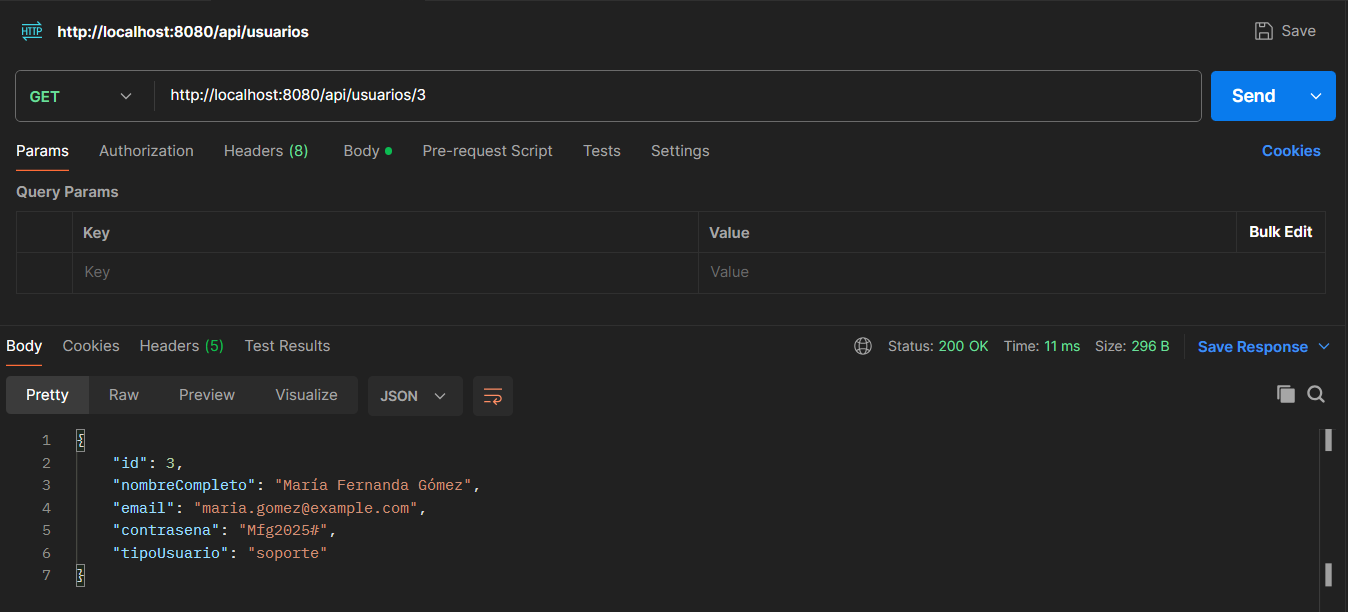
**POST**



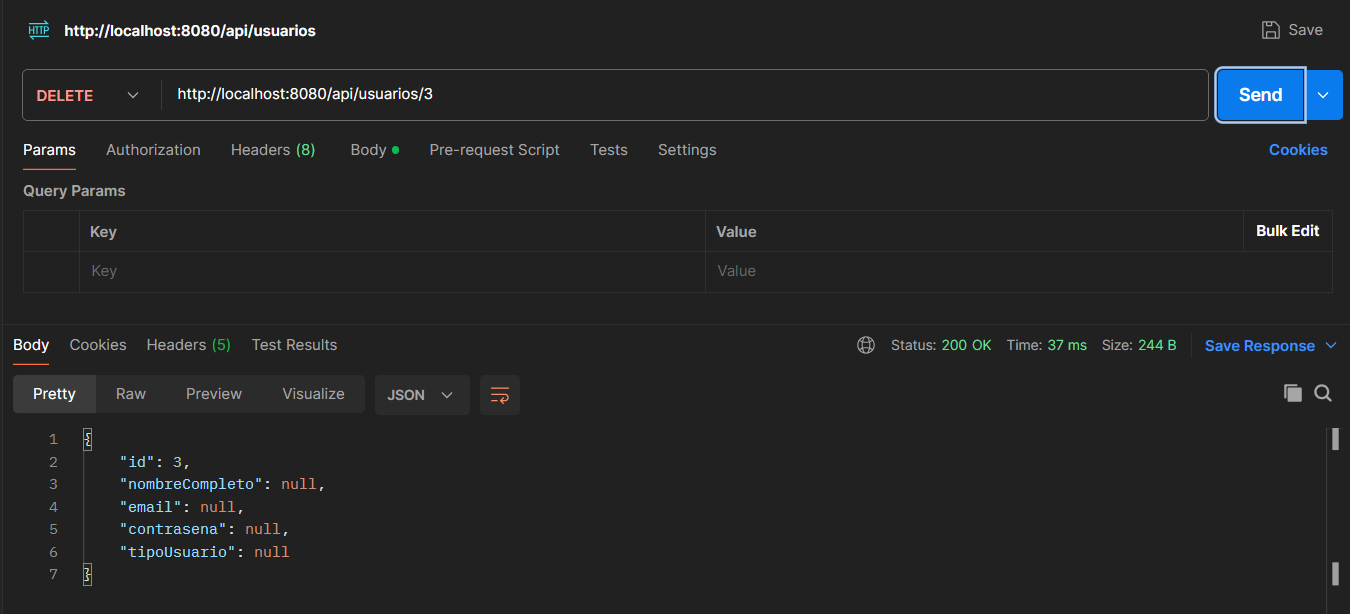
**PUT**



**SE MUESTRA LA MODIFICACIÓN:**

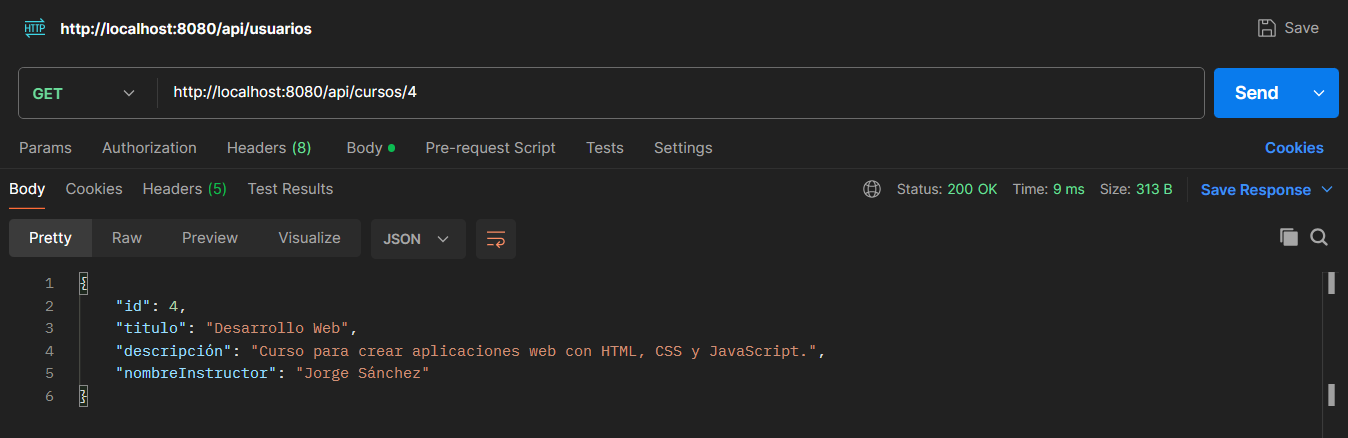
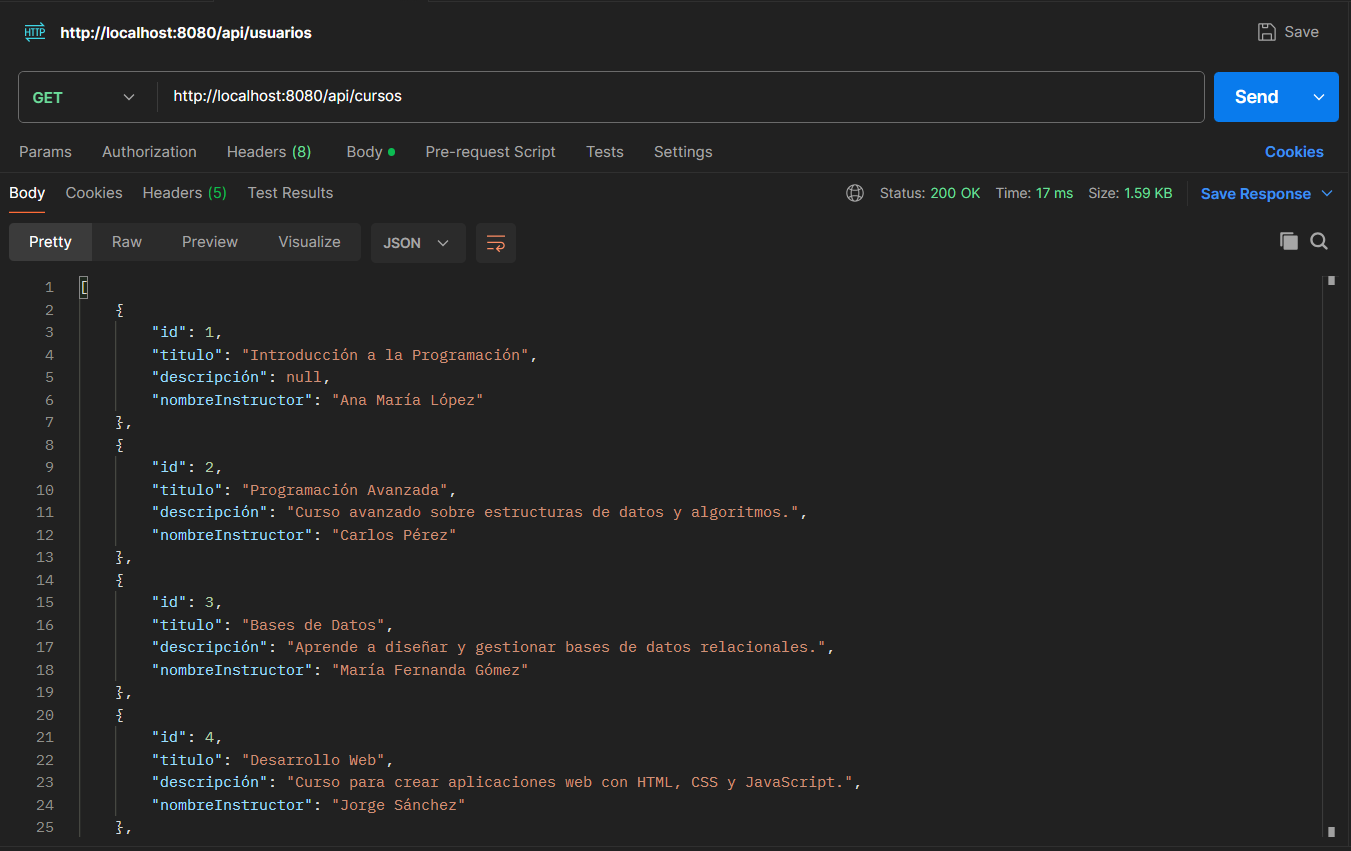


**DELETE**

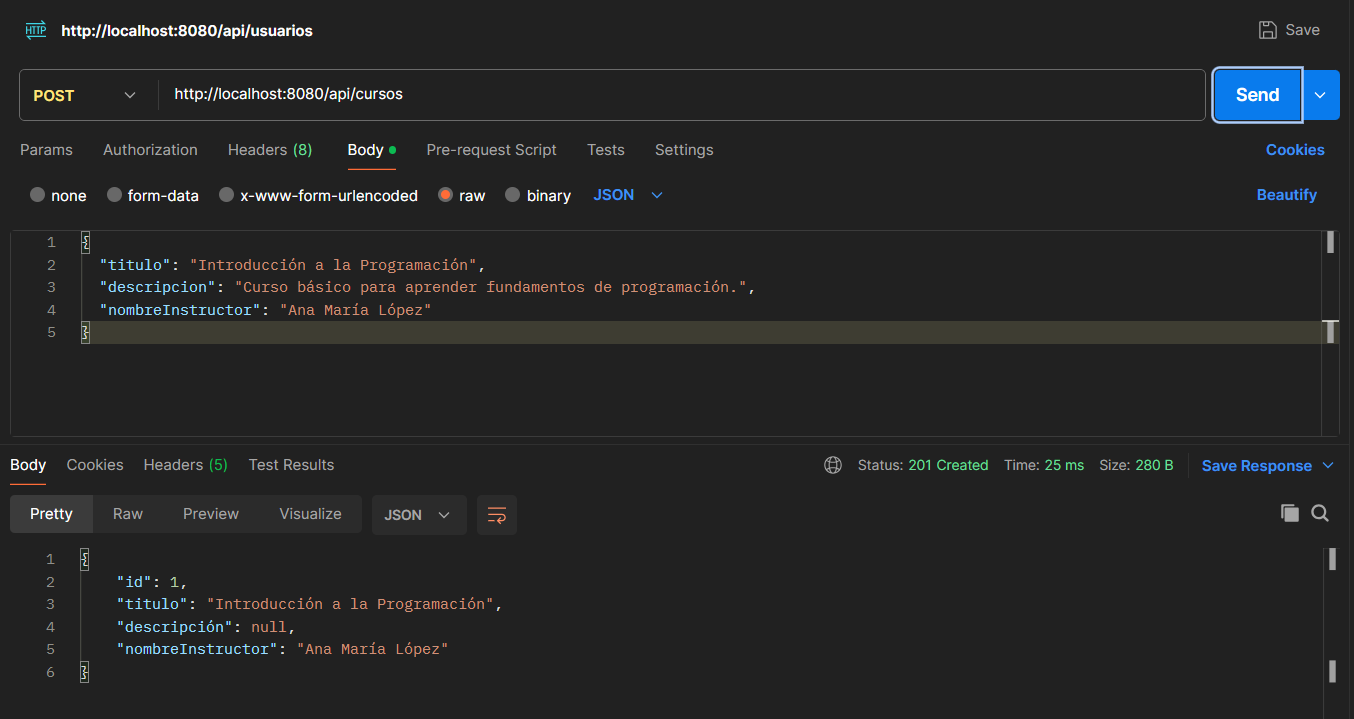


**CURSO:**

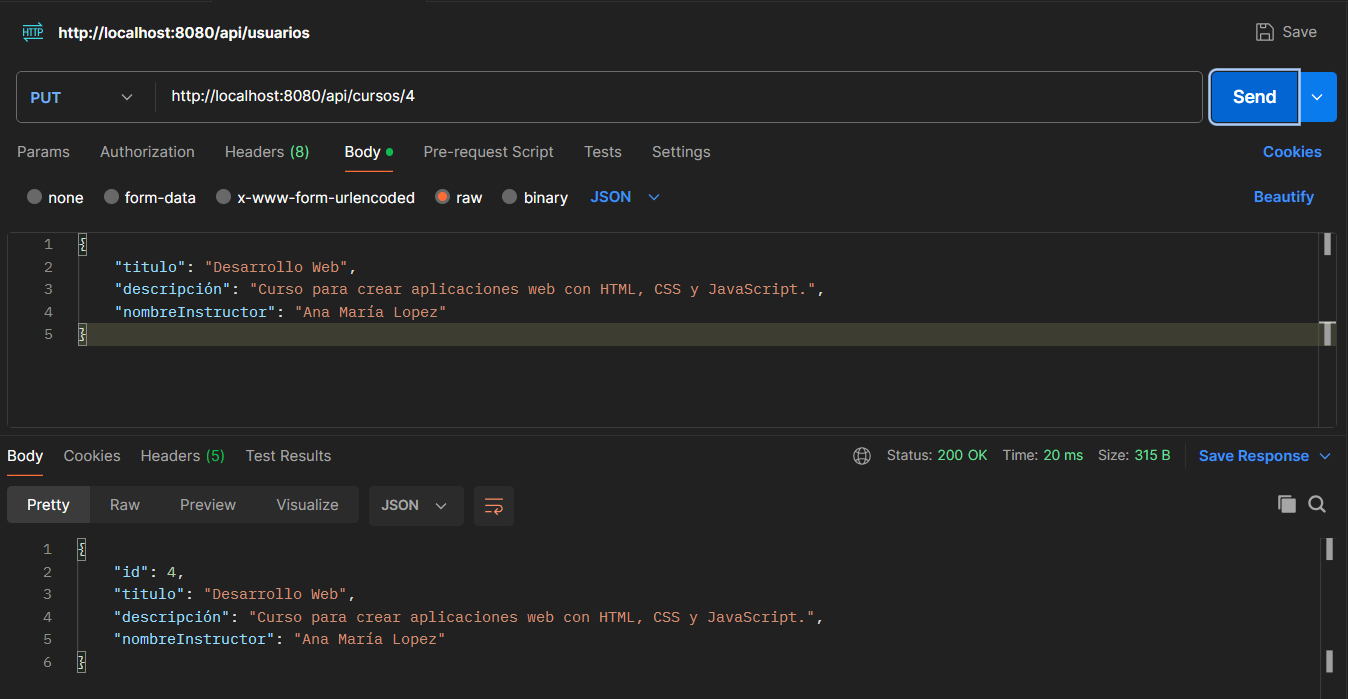
**GET**



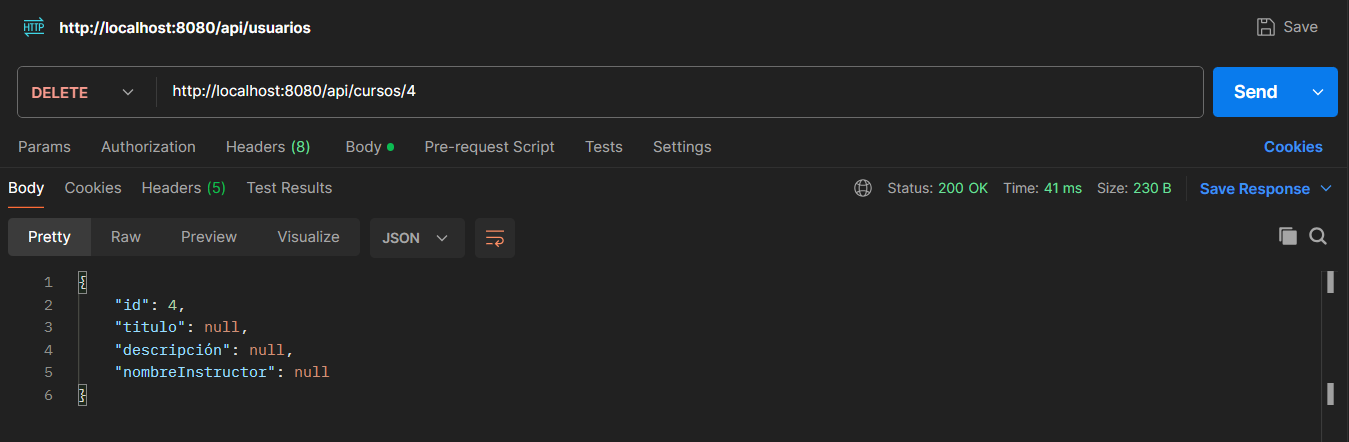
**POST**



**PUT**

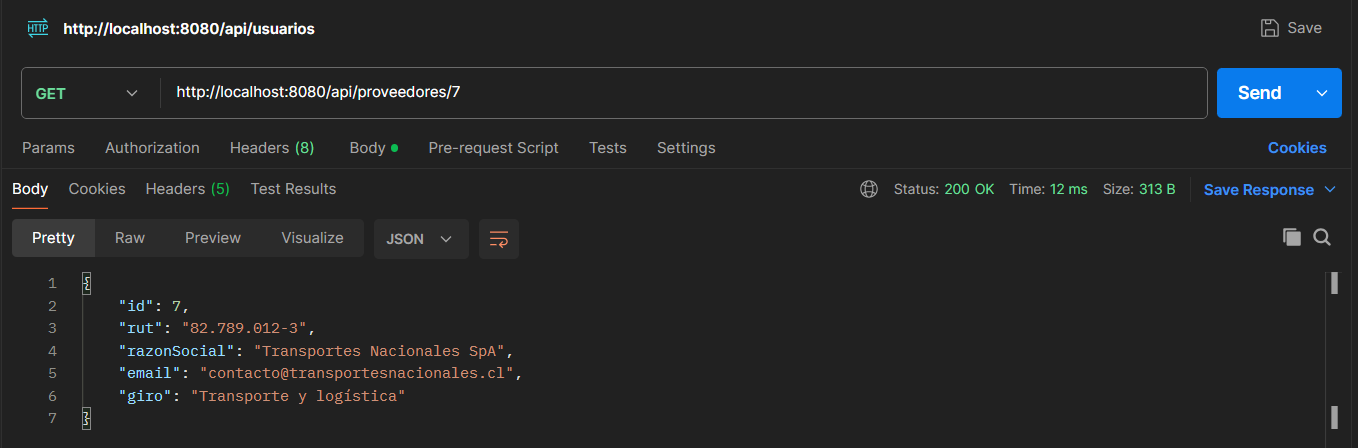
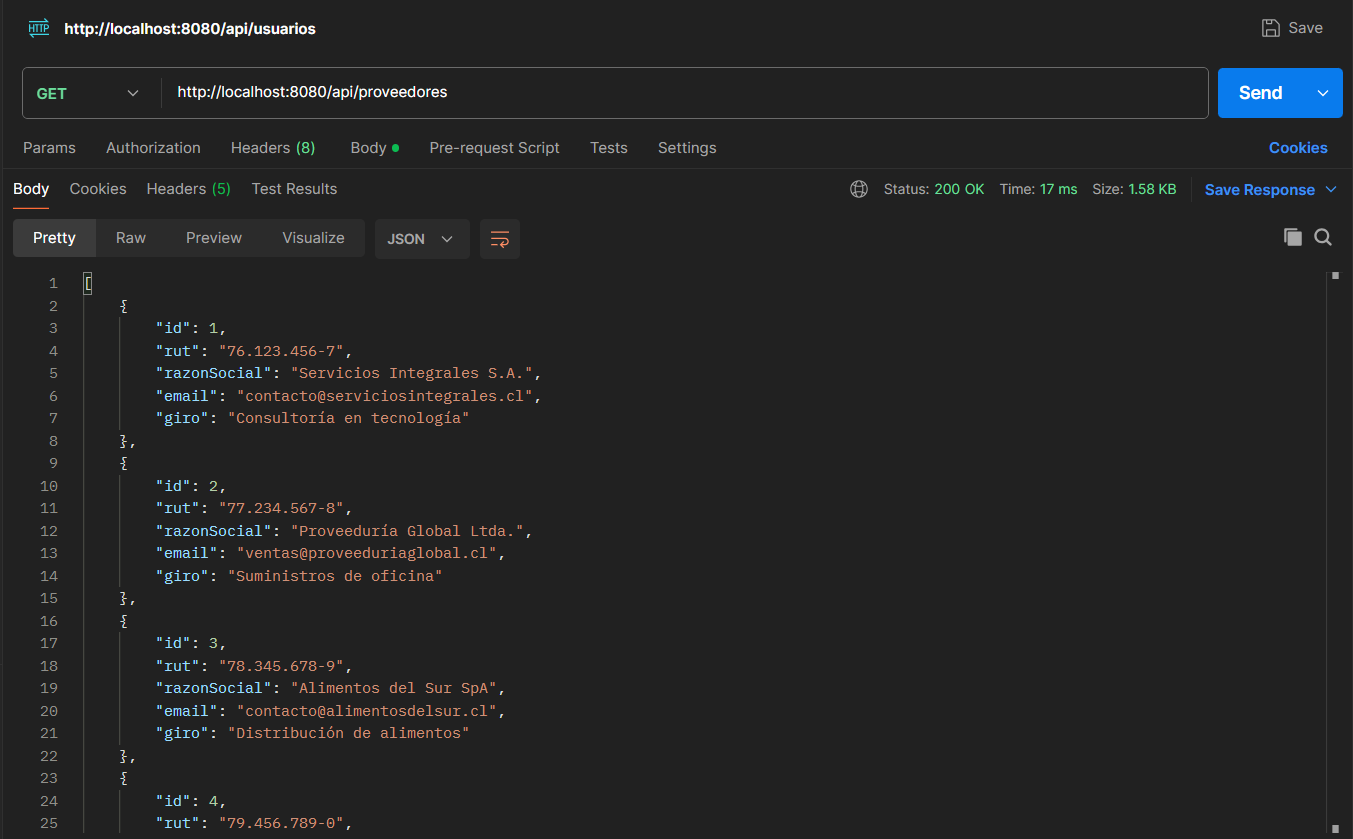


**DELETE**

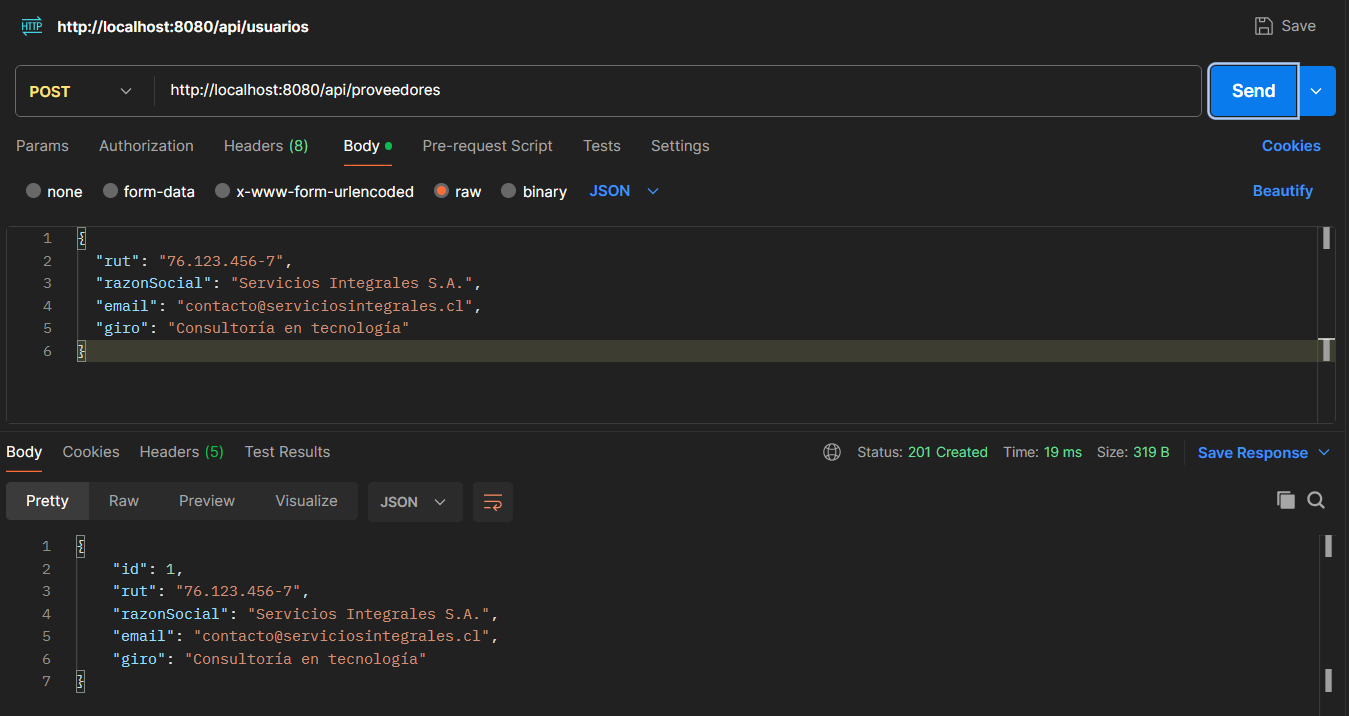


**PROVEEDOR:**

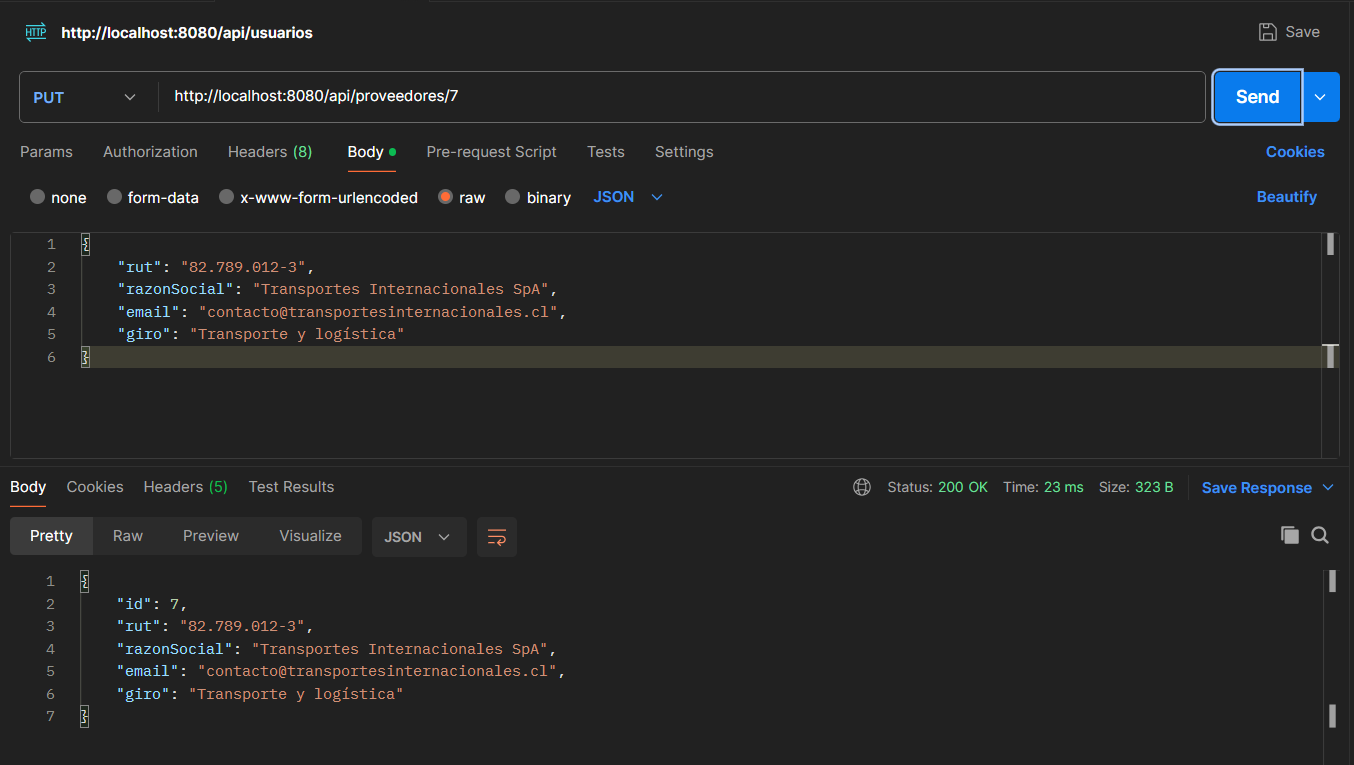
**GET**



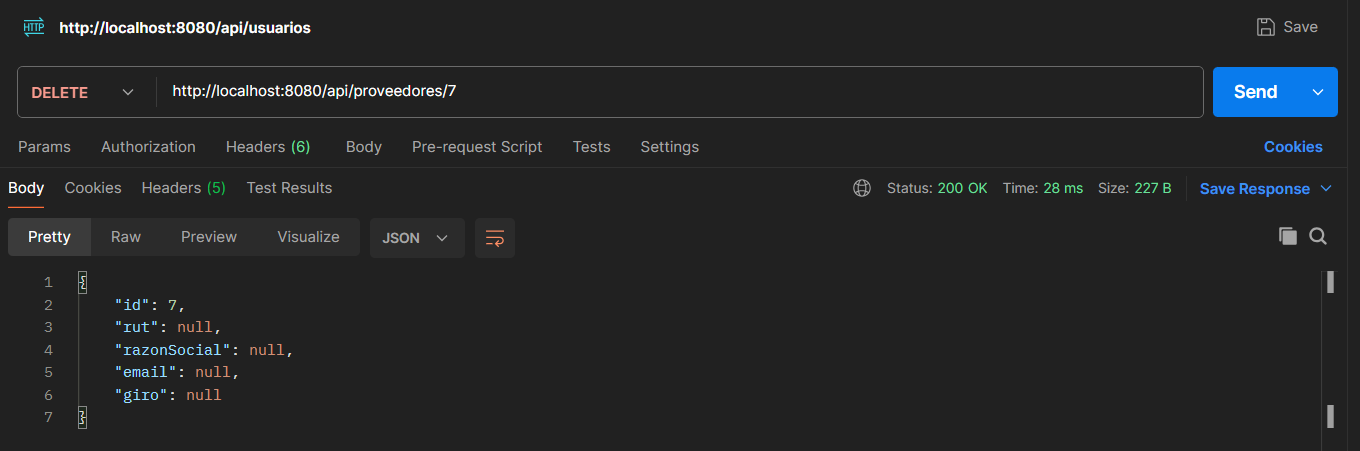
**POST**



**PUT**



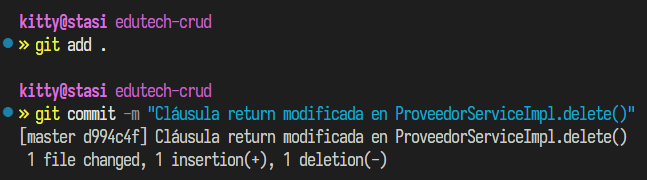
**DELETE**



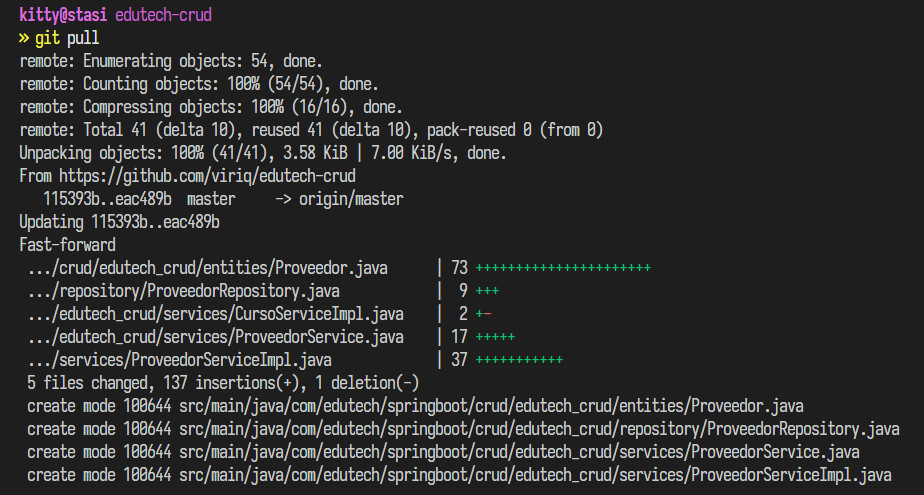
**6. GIT / GITHUB**

**6.1. Comandos utilizados**

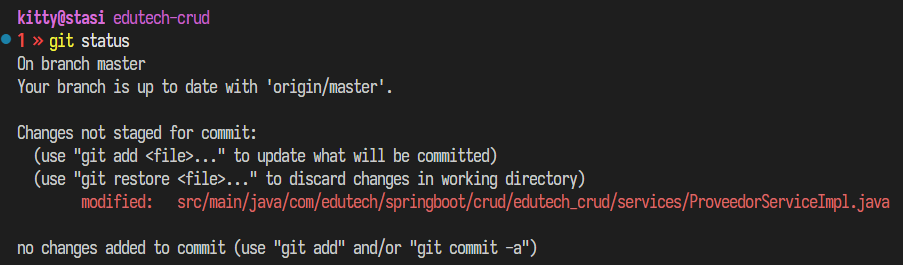
GIT ADD COMMIT



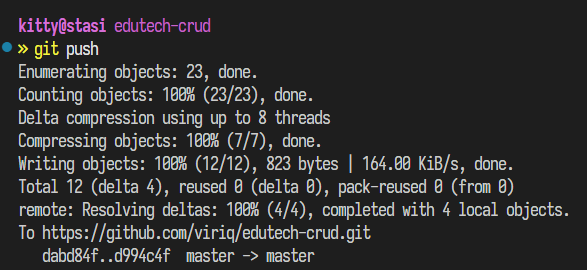
GIT PULL



GIT STATUS



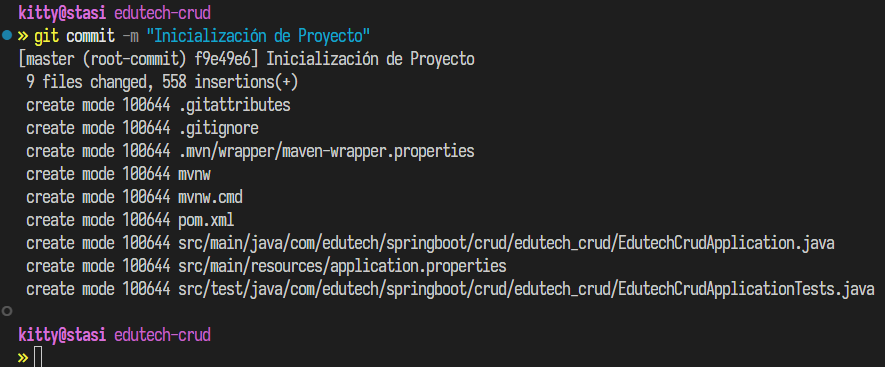
GIT PUSH



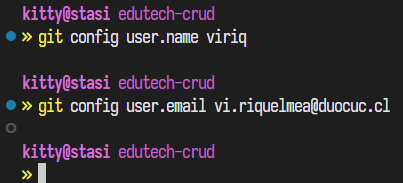
INIT GIT ADD



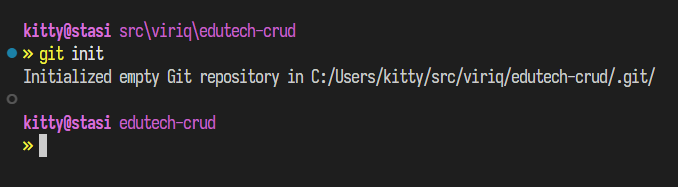
INIT GIT COMMIT



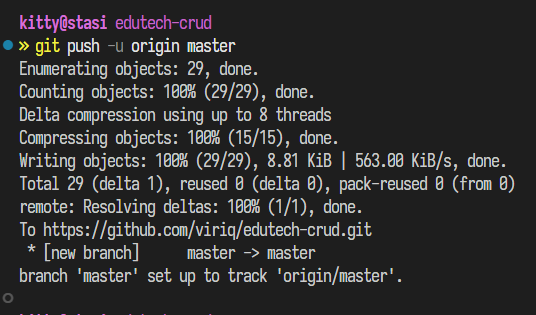
INIT GIT CONFIG



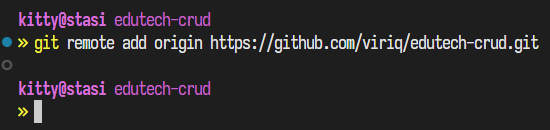
INIT GIT INIT



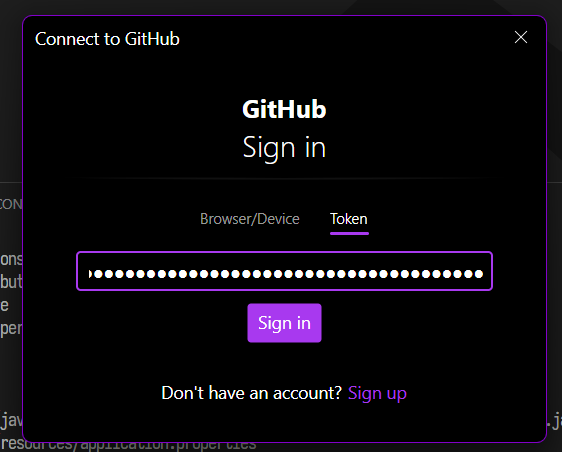
INIT GIT PUSH U



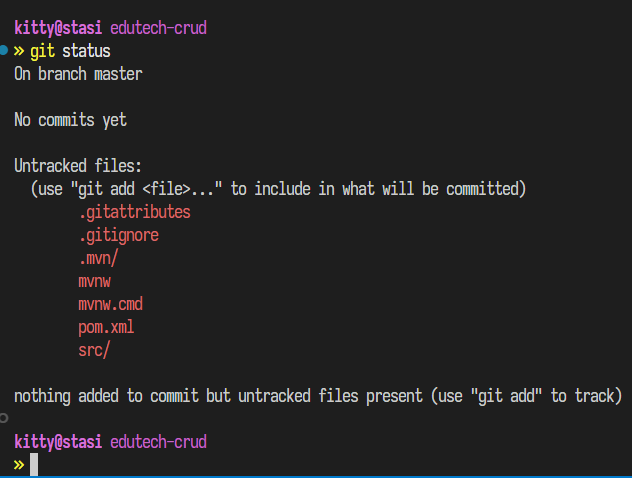
INIT GIT REMOTE ADD ORIGIN



INIT GIT SIGNIN TOKEN



INIT GIT STATUS



**CONCLUSIÓN**

El desarrollo de esta segunda etapa permitió consolidar los conocimientos adquiridos en torno a la arquitectura de microservicios y su implementación práctica mediante el uso del framework Spring, el gestor de proyectos Maven y herramientas de validación como Postman. A través del diseño e implementación de tres servicios REST, incluyendo el servicio de usuarios, se logró demostrar una comprensión sólida de la estructura y funcionalidad del backend en un entorno de desarrollo fullstack.

Se aplicaron buenas tecnicas de diseño, asi como tambien el uso de Git como nuestro sistema de control de versiones, el que facilitó el trabajo colaborativo y la correcta gestión del proyecto en equipo, permitiendo un seguimiento ordenado de los avances y mejoras.

En definitiva, esta experiencia práctica fortaleció las habilidades técnicas y colaborativas del equipo, preparándonos para enfrentar desafíos similares en contextos profesionales reales. La implementación de los servicios, su documentación y validación constituyen una base sólida para continuar desarrollando soluciones orientadas a servicios con estándares actuales de calidad y eficiencia.