Desarrollar software a partir de la integración de sus módulos componentes
GA8-220501096-AA1-EV01

Aprendiz:
Miguel Angel Pineda Vega

Instructor
Juan Manuel Aldana Zambrano

Servicio Nacional de aprendizaje
Tecnologia en analisis y desarrollo de software
2024

Introducción

En el contexto del proyecto educativo en el que estamos inmersos, el desarrollo del backend constituye una fase crítica que implica la integración y codificación de diversos módulos que conforman la estructura fundamental de nuestra aplicación web. Este proceso no solo requiere una comprensión profunda de los requisitos del sistema y de las especificaciones técnicas detalladas en los diagramas de clases y componentes, sino también una implementación efectiva utilizando las herramientas y tecnologías adecuadas.

Para nuestro proyecto, hemos elegido utilizar Node.js y Express como tecnologías centrales para el servidor debido a su eficiencia y capacidad para manejar múltiples solicitudes simultáneamente, lo que es esencial para el rendimiento de nuestra aplicación. La estructura del proyecto se organiza en una arquitectura MEAN (MongoDB, Express, Angular, Node.js), centrando en esta fase la atención exclusivamente en el backend.

El backend de nuestra aplicación se encarga de manejar la lógica de negocios, la interacción con la base de datos y la autenticación de usuarios, entre otras funciones críticas. Para asegurar que el sistema es robusto y seguro, se aplicarán prácticas de codificación estándar y se utilizarán patrones de diseño adecuados para la arquitectura del software por componentes que estamos desarrollando.

Además, se hará uso de GIT para el control de versiones, asegurando que cada cambio en el código sea rastreable y reversible si es necesario. Esto es parte de las buenas prácticas de desarrollo que integramos en nuestro flujo de trabajo para minimizar los errores y mejorar la colaboración entre los desarrolladores.

En resumen, este documento guiará el desarrollo del backend, detallando cada paso necesario para transformar los requisitos y modelos en un sistema funcional y bien integrado, preparado para interactuar eficientemente con el frontend y otros servicios de nuestra infraestructura de software.

Objetivo General

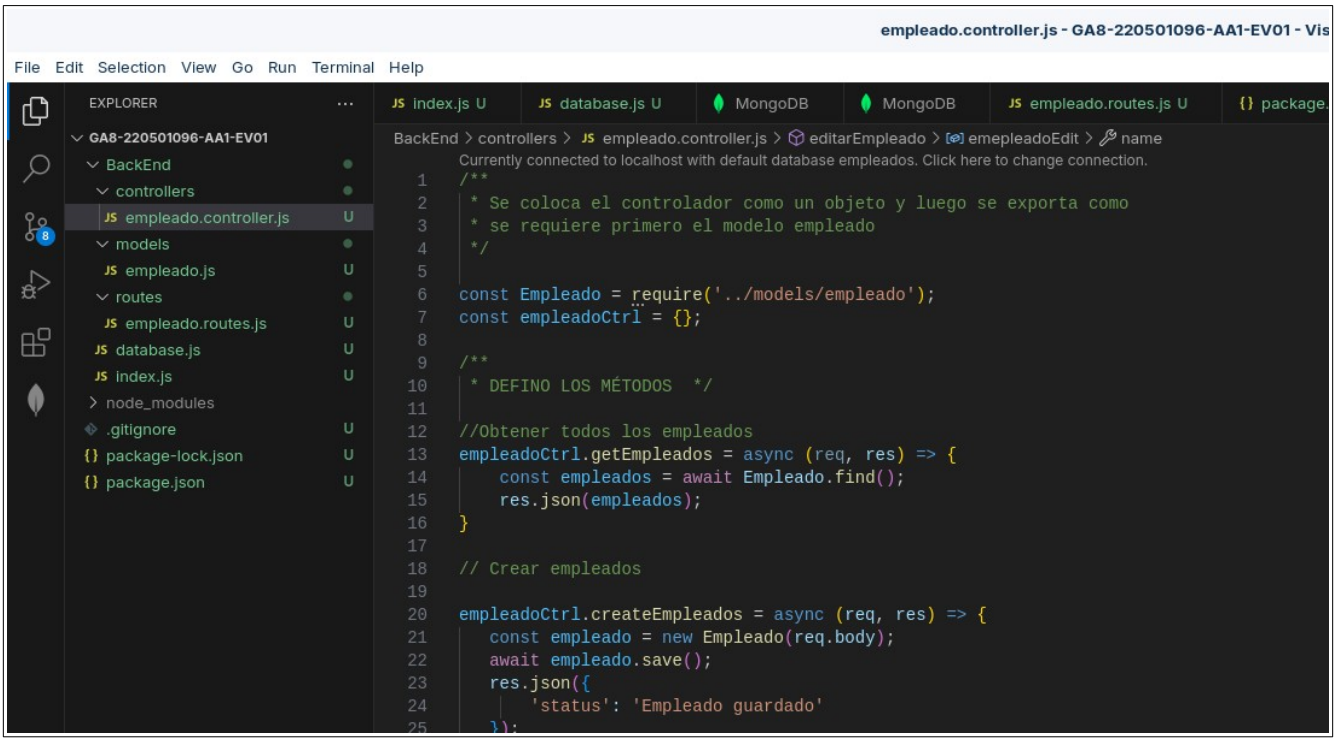
Desarrollar un backend robusto y escalable para una aplicación web orientada a la gestión de empleados, utilizando Node.js y Express, que integre eficientemente los módulos componentes según los requerimientos especificados, asegurando alta disponibilidad y seguridad en las operaciones de la aplicación.

Objetivos Específicos

- Implementar la estructura de base de datos y los modelos en MongoDB para gestionar eficazmente los datos de los empleados, incluyendo sus detalles personales, datos de empleo y registros de actividad, utilizando Mongoose para facilitar la interacción con la base de datos y garantizar la integridad de los datos.

- Desarrollar y probar APIs RESTful que permitan la interacción segura y eficiente entre el frontend y el backend, incluyendo autenticación de usuarios y manejo de sesiones, aplicando pruebas unitarias para asegurar el correcto funcionamiento de cada componente antes de su integración final.

Desarrollo de la evidencia



En la imagen se muestra la estructura de un proyecto típico de backend utilizando Node.js y Express, estructurado de manera clara y organizada en el entorno de desarrollo Visual Studio Code (VSCode). Aquí está la descripción de cada parte de la estructura del proyecto:

1. Backend (Directorio Raíz)

controllers: Este directorio contiene los controladores de la aplicación. Los controladores gestionan las solicitudes entrantes y devuelven respuestas al cliente. En este caso, tienes un archivo:

> **empleado.controller.js:** Maneja las operaciones CRUD (Crear, Leer, Actualizar, Eliminar) para los "empleados". Contiene métodos para obtener todos los empleados y crear un nuevo empleado.

models: Contiene los modelos de datos utilizados en la aplicación. Los modelos definen la estructura de los datos que se guardan en la base de datos.

> **empleado.js**: Define el modelo de datos para un "empleado", utilizando Mongoose trabajando con MongoDB.

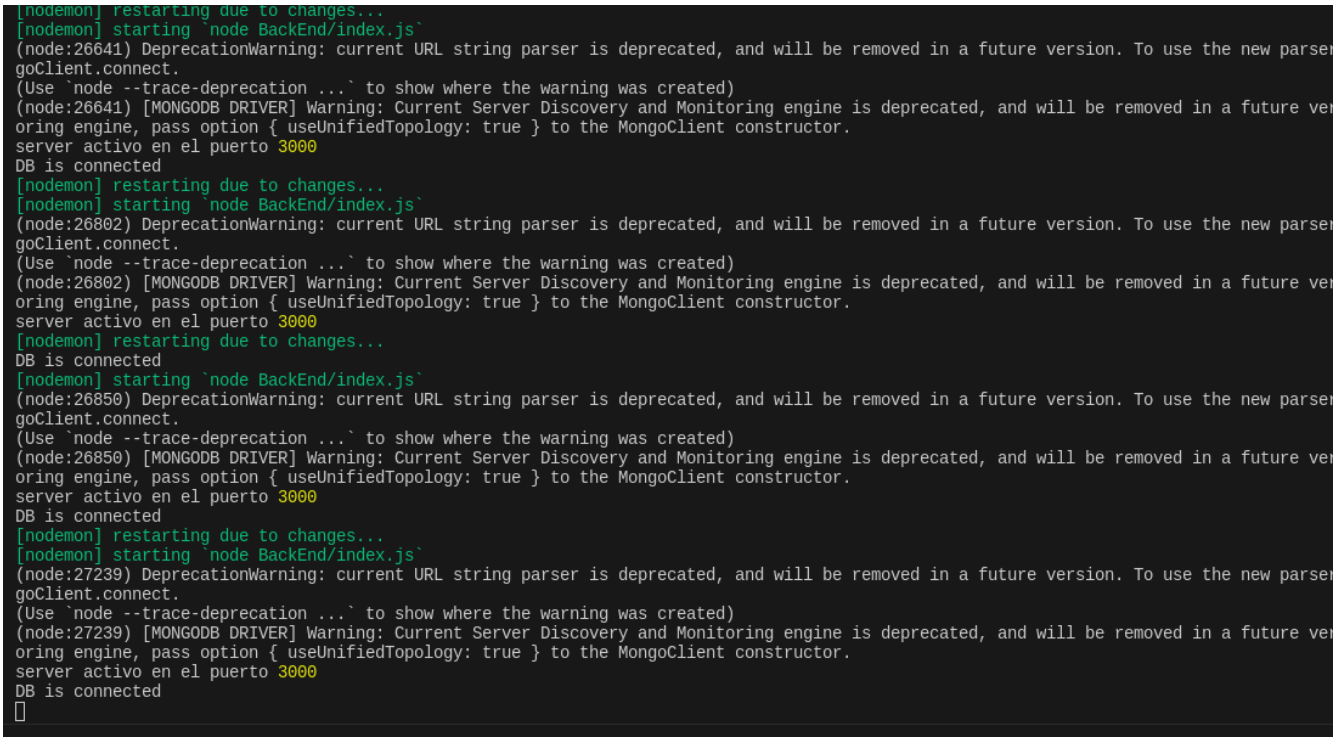
routes: Este directorio aloja las rutas de la aplicación, que definen endpoints URI y métodos HTTP específicos para acceder a los recursos.

> **empleado.routes.js:** Define las rutas para acceder a las funciones definidas en empleado.controller.js.

database.js: Este archivo contiene la configuración de la base de datos, como la conexión a MongoDB. Se encarga de establecer la conexión con la base de datos utilizando las credenciales y la URI proporcionadas.

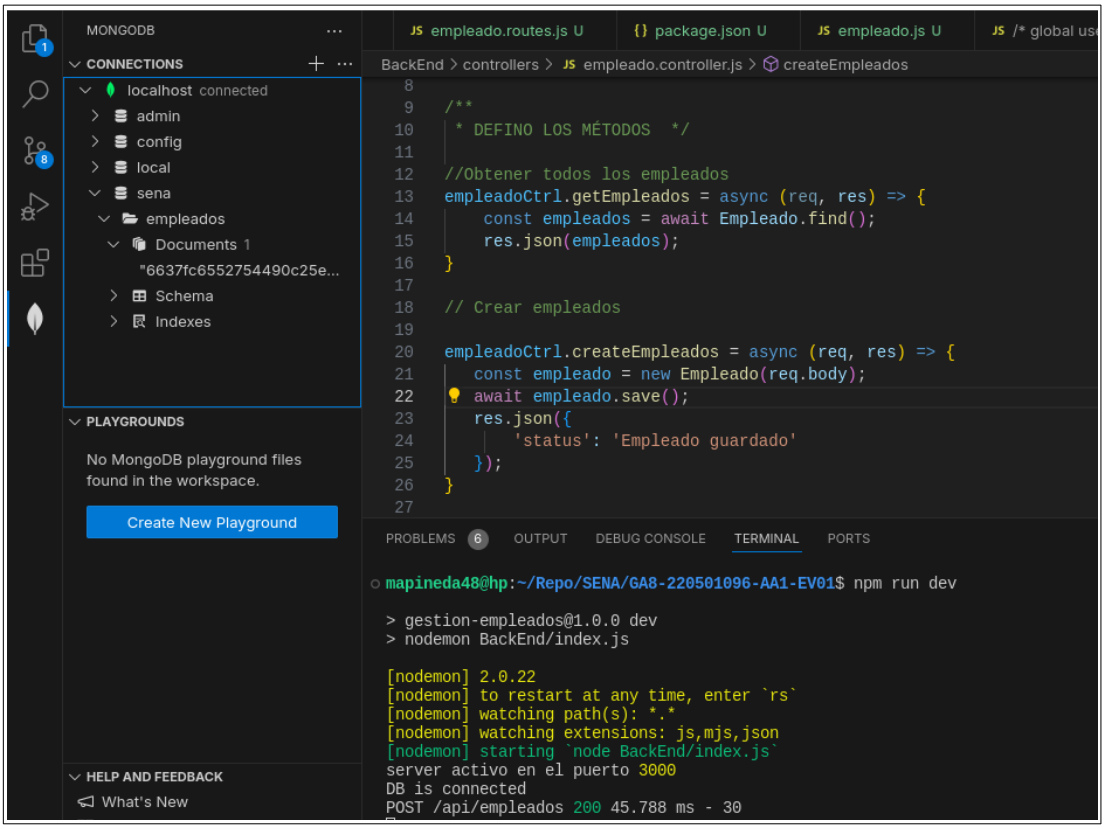
index.js: Este es el archivo principal de la aplicación, que inicia el servidor y agrupa todas las configuraciones, rutas y middlewares. Aquí es donde configuras el puerto del servidor, aplicas middlewares como JSON body-parser y conectas todas las rutas.

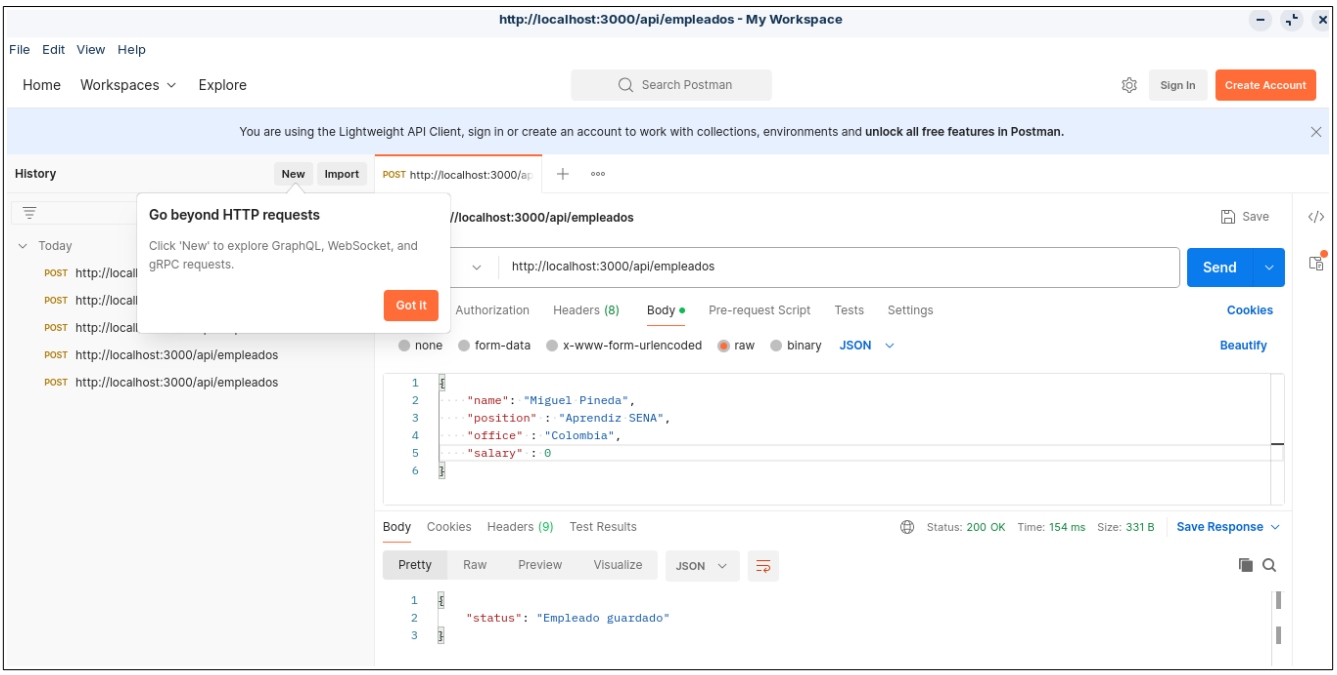
Iniciar el proyecto con **npm run dev**



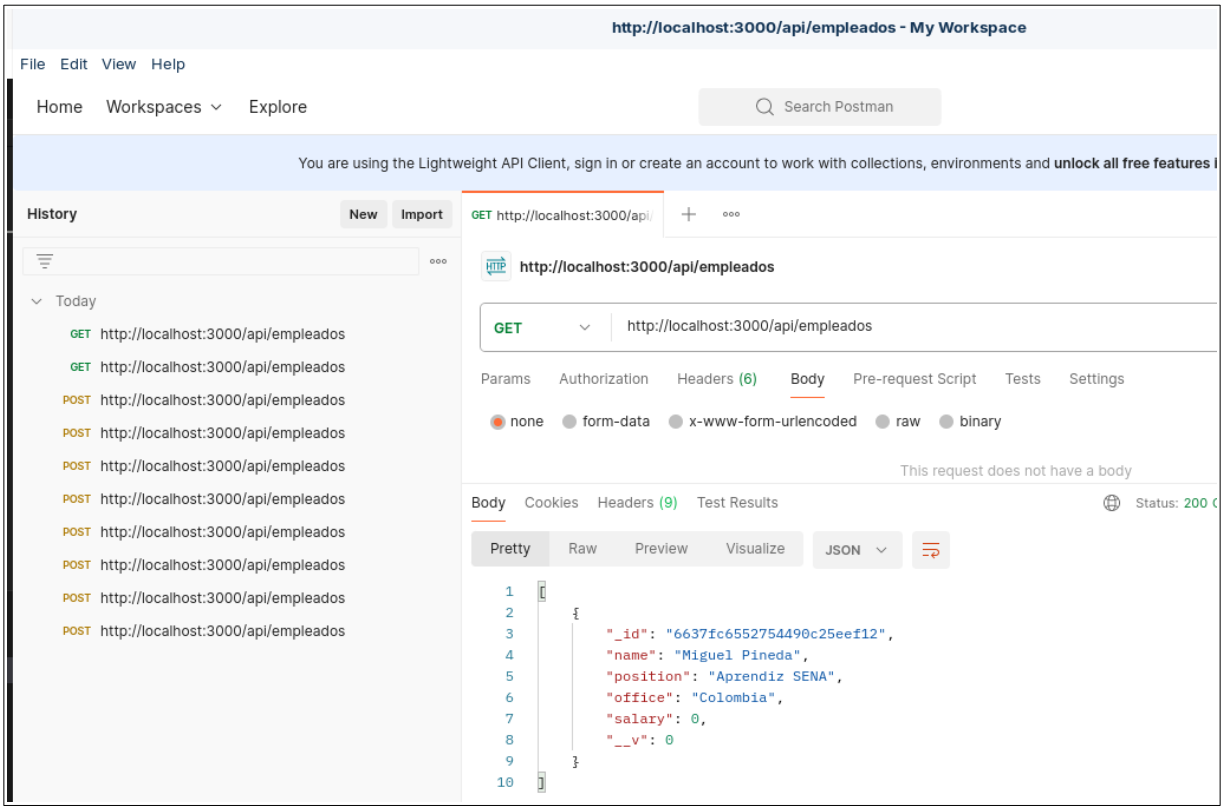
Postman

Crear un nuevo empleado

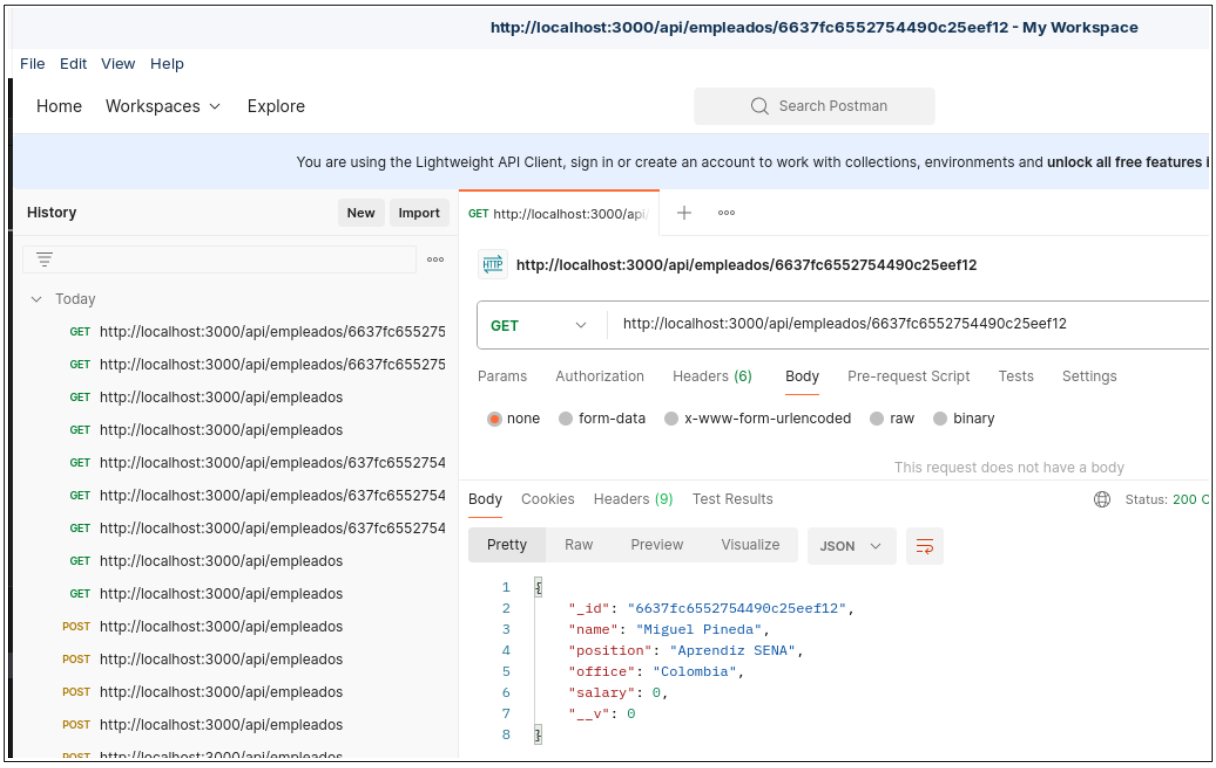




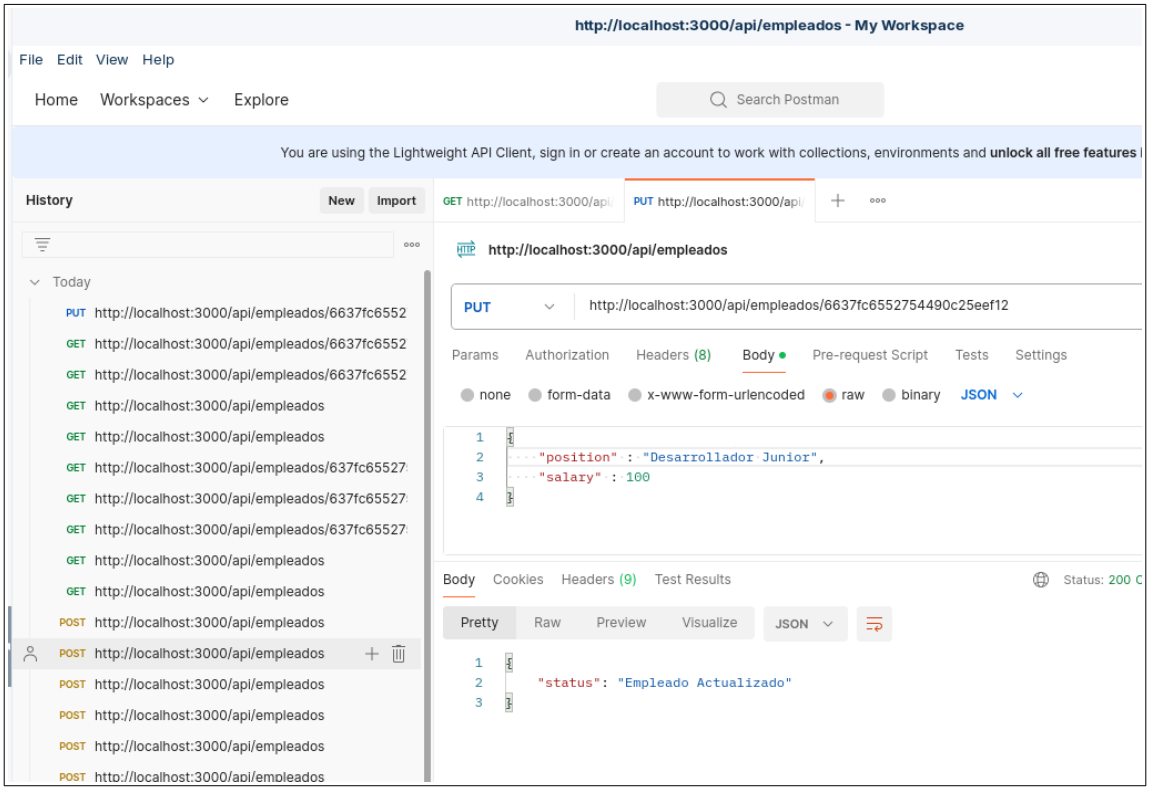
Consultar registros



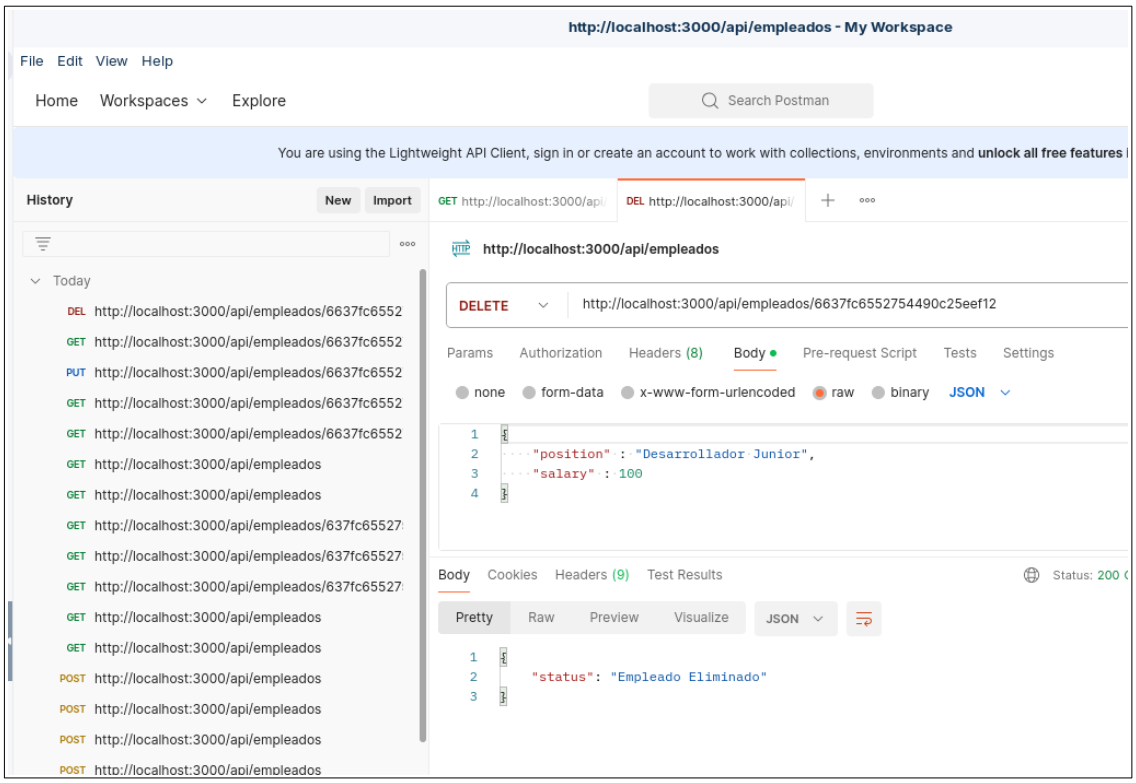
Consultar registro



Actualizar registro



Eliminar registro



Conclusión

Este proyecto no solo ha reforzado la importancia de seguir buenas prácticas de desarrollo y arquitectura de software, sino que también ha demostrado la eficacia de utilizar control de versiones y metodologías de testing para mantener la calidad del software. La implementación de autenticación y seguridad en las operaciones de la base de datos ha sido crucial para proteger la integridad y privacidad de la información manejada.

Mirando hacia el futuro, el proyecto está bien posicionado para futuras expansiones y mejoras. El siguiente paso será integrar este backend con interfaces de usuario diseñadas específicamente para los distintos roles de usuarios, asegurando una experiencia fluida y funcional para todos los empleados. Además, se contempla la incorporación de más funcionalidades como reportes avanzados y análisis de datos para mejorar la toma de decisiones dentro de la empresa.