

SENA

Centro Nacional de hotelería, turismo y alimentos

**Ficha:
2627076**

Trabajo:

GA7-220501096-AA5-EV04 API del proyecto

Aprendices

**Milton Herrera
Talía Yurani López Gualcala
María Camila Carreño Maldonado
Jorge Orlando Rincón Ortiz
Miguel Angel Torres Díaz**

Servicio Nacional de Aprendizaje

2024

Introducción

Este documento se centra en el proyecto de creación de usuarios utilizando Node.js y Express, con un enfoque específico en las pruebas realizadas con **Postman** y las validaciones implementadas en la aplicación. Mientras que la implementación del **CRUD** de usuarios utilizando estas tecnologías es el núcleo del proyecto, el énfasis principal aquí recae en la robustez y fiabilidad de la aplicación a través de las pruebas exhaustivas y las validaciones implementadas.

Al utilizar Node.js y Express, el proyecto proporciona una plataforma sólida para el desarrollo de aplicaciones web escalables y de alto rendimiento. Sin embargo, la calidad de la aplicación no solo se basa en su funcionalidad, sino también en su capacidad para manejar entradas de usuarios de manera segura y consistente. Es por eso que se han realizado pruebas exhaustivas utilizando Postman, una herramienta ampliamente utilizada para probar y documentar APIs web.

Además, se han implementado diversas validaciones en la aplicación para garantizar la integridad de los datos y prevenir posibles errores o ataques de seguridad. Estas validaciones abarcan desde la verificación de datos de entrada hasta la autenticación de usuarios y la protección contra ataques de inyección SQL.

A lo largo de este documento, se explorarán en detalle las pruebas realizadas con Postman, incluyendo los escenarios de prueba, los resultados obtenidos y cualquier acción correctiva tomada en respuesta a los hallazgos de las pruebas. También se proporcionará información detallada sobre las diferentes validaciones implementadas en la aplicación y cómo contribuyen a su robustez y seguridad general.

El proyecto se basa en varias tecnologías clave para su implementación exitosa:

Qué es Postman

Postman es una plataforma completa para el desarrollo y la gestión de APIs, que ofrece una interfaz fácil de usar para crear, ejecutar y analizar pruebas de API. Su versatilidad y flexibilidad lo convierten en una herramienta invaluable para desarrolladores y equipos de pruebas que desean asegurar la calidad y el rendimiento de sus APIs.

Cómo Postman nos ayudó

Utilizando Postman, pudimos realizar una variedad de pruebas, incluyendo:

-
- GET: Para recuperar datos de usuarios existentes.
- PUT: Para actualizar la información del usuario.
- POST: Para agregar nuevos usuarios al sistema.
- DELETE: Para eliminar usuarios del sistema.
-

Postman nos permitió enviar fácilmente diferentes tipos de solicitudes HTTP y verificar las respuestas recibidas. Esto nos permitió identificar y corregir cualquier problema o error en la funcionalidad de la aplicación de manera eficiente.

Funcionalidades adicionales de Postman

Postman ofrece características adicionales, como la capacidad de automatizar pruebas, generar documentación de API y colaborar en equipo, lo que lo convierte en una herramienta integral para el desarrollo de APIs web.

En resumen, Postman desempeñó un papel fundamental en la validación y verificación de la funcionalidad de nuestra aplicación, ayudándonos a garantizar su calidad y fiabilidad.

Node.js

Es un entorno de ejecución de JavaScript construido sobre el motor V8 de Chrome, que permite ejecutar JavaScript en el lado del servidor. Node.js proporciona una arquitectura de eventos asíncronos y no bloqueantes, lo que lo hace ideal para aplicaciones web escalables y de alto rendimiento.

Express.js

Es un marco de aplicación web para Node.js que simplifica el desarrollo de aplicaciones web mediante la provisión de una capa de abstracción sobre Node.js, facilitando la gestión de rutas, middleware y peticiones HTTP.

Arquitectura Limpia (Clean Architecture)

También conocida como arquitectura hexagonal, es un enfoque arquitectónico que promueve la separación de preocupaciones y la independencia de capas. En este proyecto, la arquitectura limpia se utiliza para organizar el código en capas claras y definidas, como la capa de dominio, la capa de aplicación y la capa de infraestructura, facilitando la modularidad y el mantenimiento del código.

Documentación del Código

Se ha prestado especial atención a la documentación del código en todas las partes del proyecto. Esto incluye comentarios claros y descriptivos en el código fuente, así como documentación adicional para funciones, métodos y clases. La documentación adecuada del código es fundamental para mejorar la comprensión y el mantenimiento del mismo a lo largo del tiempo.

Link del repositorio:

- <https://github.com/migueltorresd/bank>

Link del video:

- <https://youtu.be/lANKjqrfkml>