**QUALITY CONTROL ENGINEER PRUEBA TÉCNICA – DANIEL MERCHAN**

**Documentación:** Pruebas de API "Your Store"

**Fecha:** 06 de septiembre de 2025

**Elaborado por:** Daniel Felipe Merchan Godoy

**Versión:** 1.0

**Objetivo:** Validar funcionalidad de la API relacionada en el requerimiento de la prueba técnica.

**Configuración del Entorno**

**URL Base:** https://fakestoreapi.com

**Herramienta:** Postman v11.0

**API TESTING**

1. **Diseño de casos de Prueba**

Se adjunta Excel con el diseño de los casos de prueba propuestos para el requerimiento.



1. **Ejecución Casos de Prueba**
   1. **CASO 1 API DISPONIBLE**

Objetivo: Verificar que la API esta en funcionamiento y el endpoint base responde correctamente

Endpoint: /products

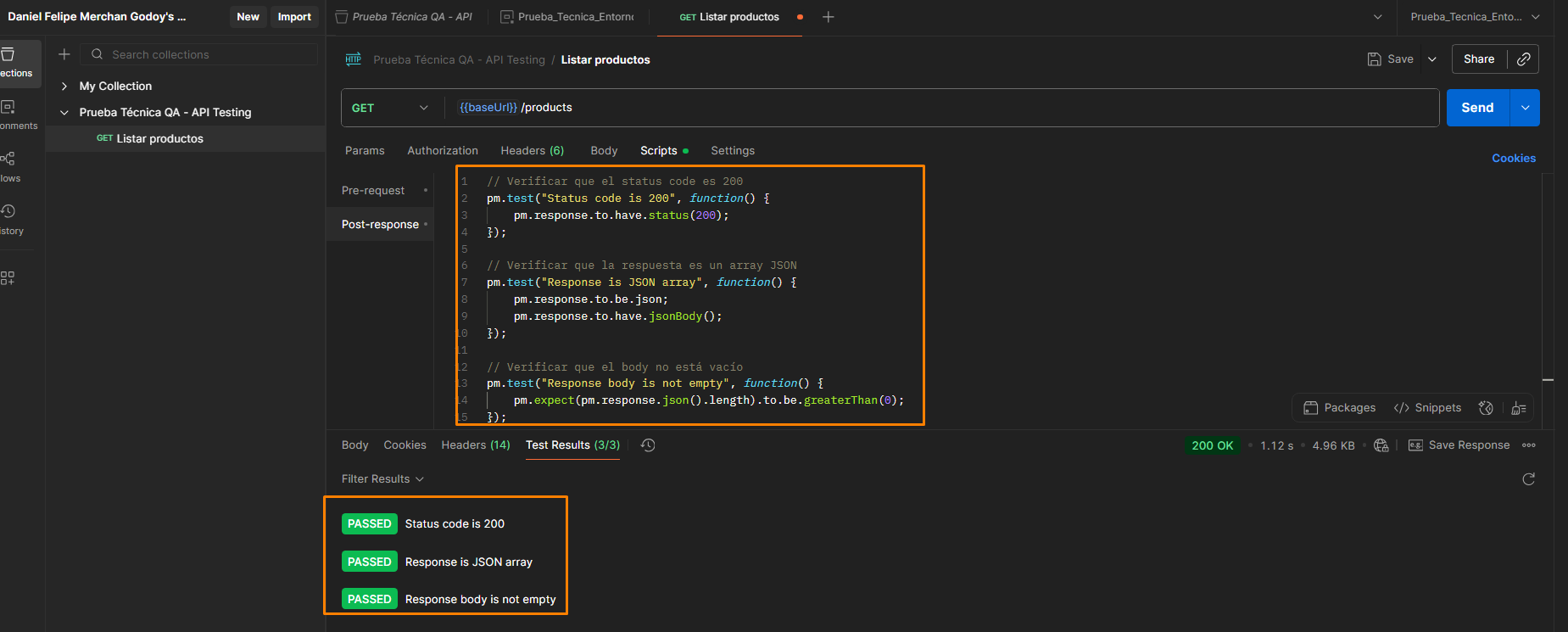
Metodo: GET

Datos de Entrada: N/A

Resutlado Esperado: Código de respuesta HTTP 200

Paso a Paso:

* Request y estado de los test ejecutados.



* Codigo de Respuesta 200

Texto

El contenido generado por IA puede ser incorrecto.

* Tiempo de respuesta 1090 ms

Captura de pantalla de un celular

El contenido generado por IA puede ser incorrecto.

* Tamaño del Response

Texto

El contenido generado por IA puede ser incorrecto.

RESULTADOS OBTENIDOS:

• Status Code: 200 OK

• Response Time: 1090ms

• Tamaño Response: 4.95 KB

• Tests Ejecutados: 3/3 passed

• Resultado Final: APROBADO

* 1. **CASO 2 Listar electronics**

Objetivo: Verificar que la API filtra correctamente los productos de acuerdo a la categoría, para este caso tomaremos la categoría “electronics”

Endpoint: /products/category/electronics

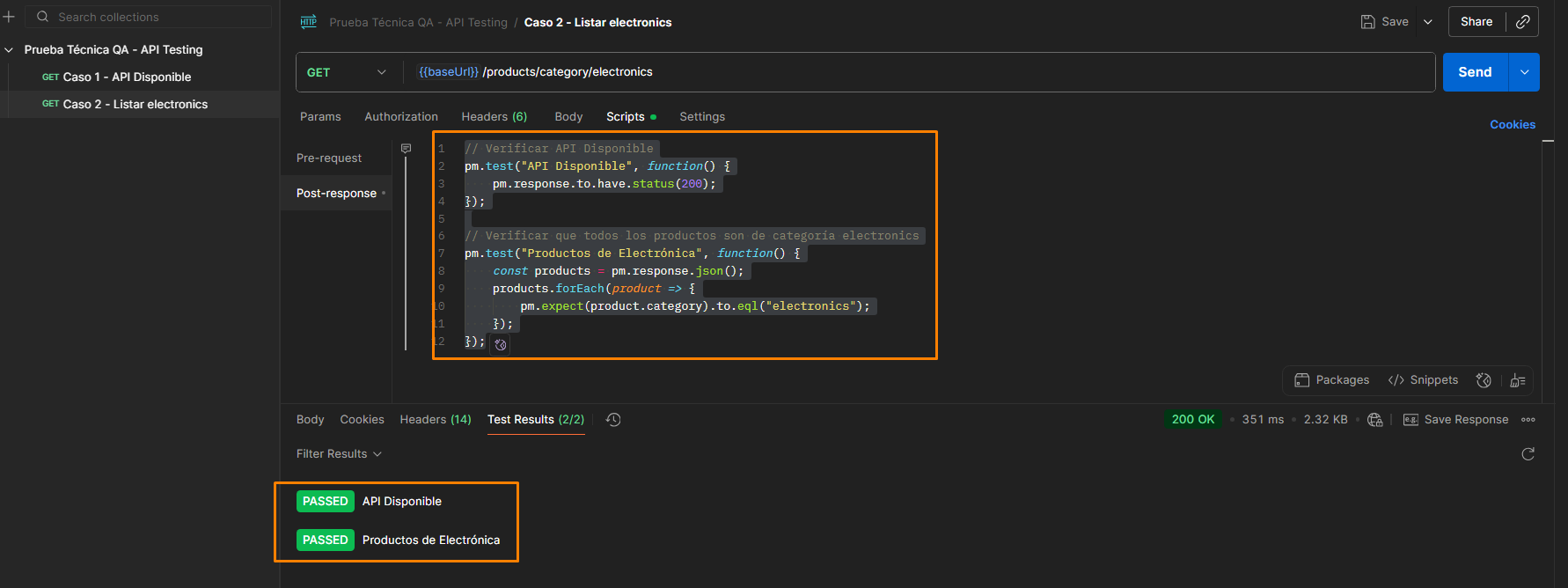
Metodo: GET

Datos de Entrada: N/A

Resutlado Esperado: Código de respuesta HTTP 200 y array con productos de la categoría.

Paso a Paso:

* Request y estado de los test ejecutados.



* Array de productos de la categoría “electronics”

Interfaz de usuario gráfica, Texto

El contenido generado por IA puede ser incorrecto.

* Respuesta OK, tiempo y tamaño de la respuesta

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

RESULTADOS OBTENIDOS:

• Status Code: 200 OK

• Response Time: 351 ms

• Tamaño Response: 2.32 KB

• Tests Ejecutados: 3/3 passed

• Resultado Final: Se listan correctamente los productos de la categoría “electronics”

* 1. **CASO 3 Obtener producto por ID**

Objetivo: Verificar que la API filtra correctamente los productos de acuerdo al ID de cada uno.

Endpoint: /products/5

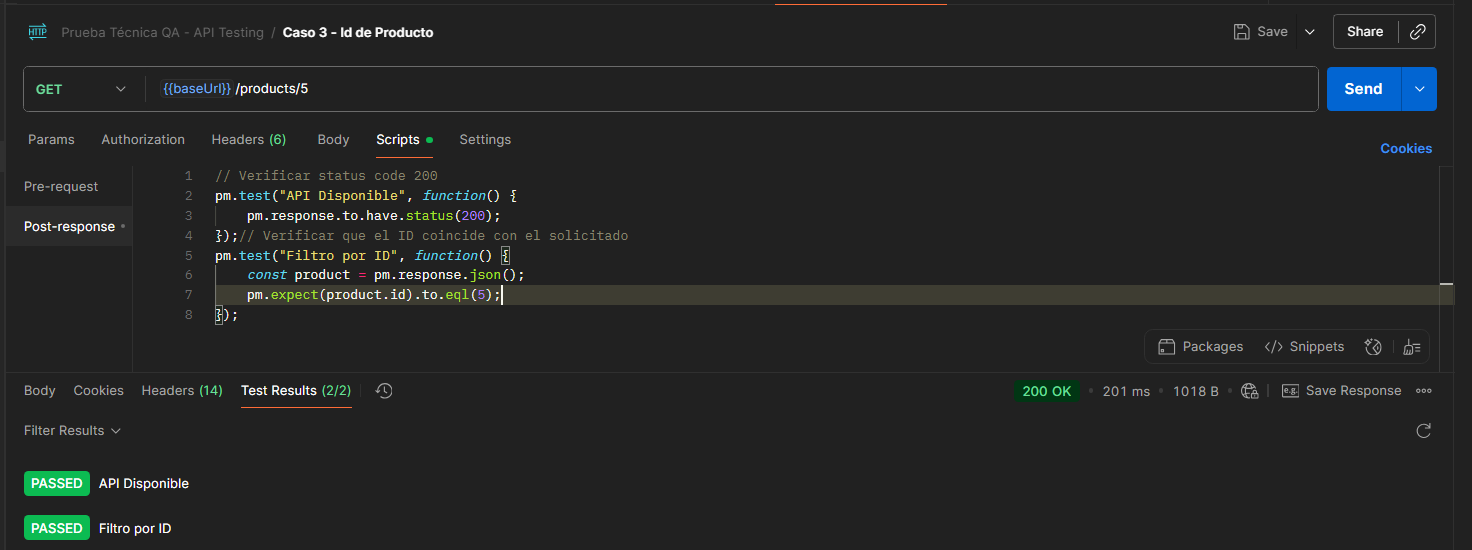
Metodo: GET

Datos de Entrada: N/A

Resutlado Esperado: Código de respuesta HTTP 200 y array de producto con ID = 5.

Paso a Paso:

* Request y estado de los test ejecutados.



* Array de producto cuyo ID = 5

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

* Respuesta OK, tiempo y tamaño de la respuesta

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

RESULTADOS OBTENIDOS:

• Status Code: 200 OK

• Response Time: 201 ms

• Tamaño Response: 1009 B

• Tests Ejecutados: 2/2 passed

• Resultado Final: Se logra filtrar exitosamente el listado de productos por el ID.

* 1. **CASO 4 Crear nuevo producto**

Objetivo: Verificar la correcta creación del nuevo producto.

Endpoint: /products

Metodo: POST

Datos de Entrada: Datos del nuevo producto.

Resutlado Esperado: Código de respuesta HTTP 201 y array de producto creado.

Paso a Paso:

* Metodo, URL y Body del nuevo producto

Interfaz de usuario gráfica

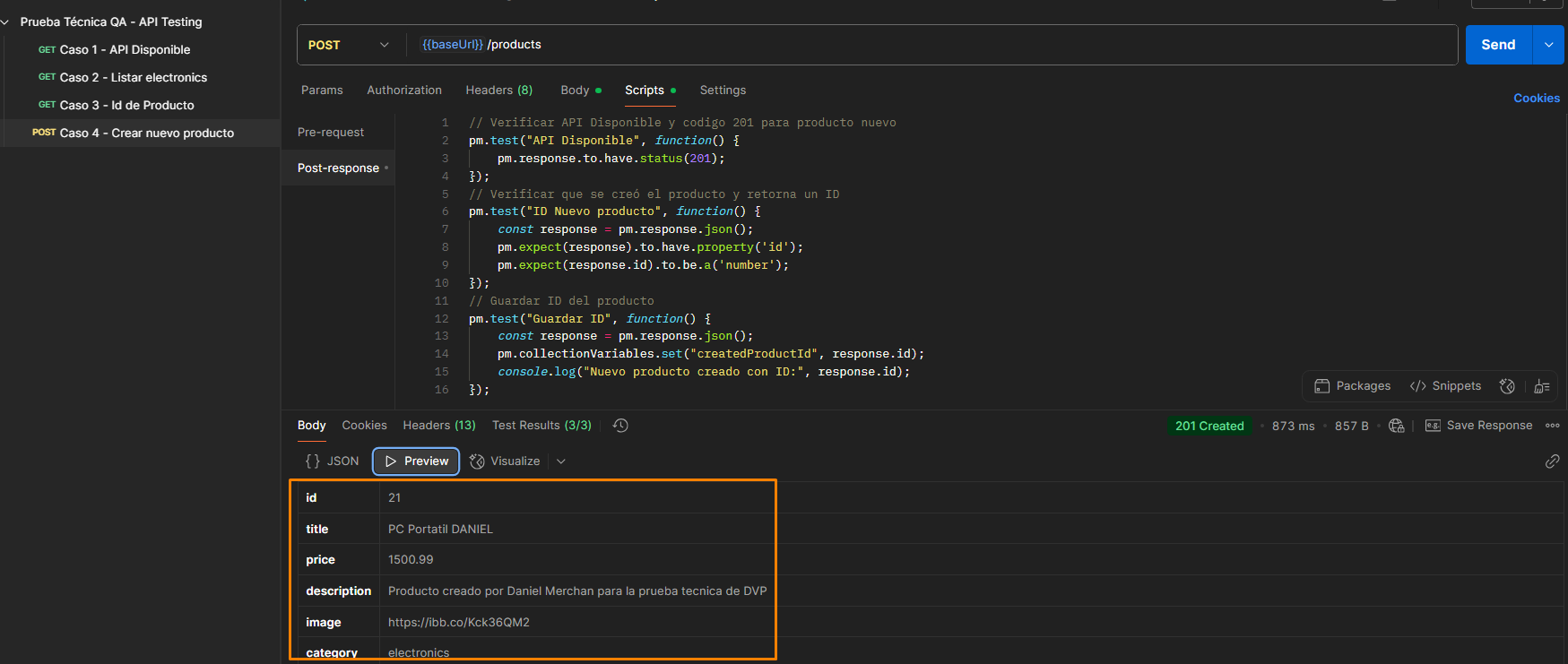
El contenido generado por IA puede ser incorrecto.

* Request y estado de los test ejecutados.

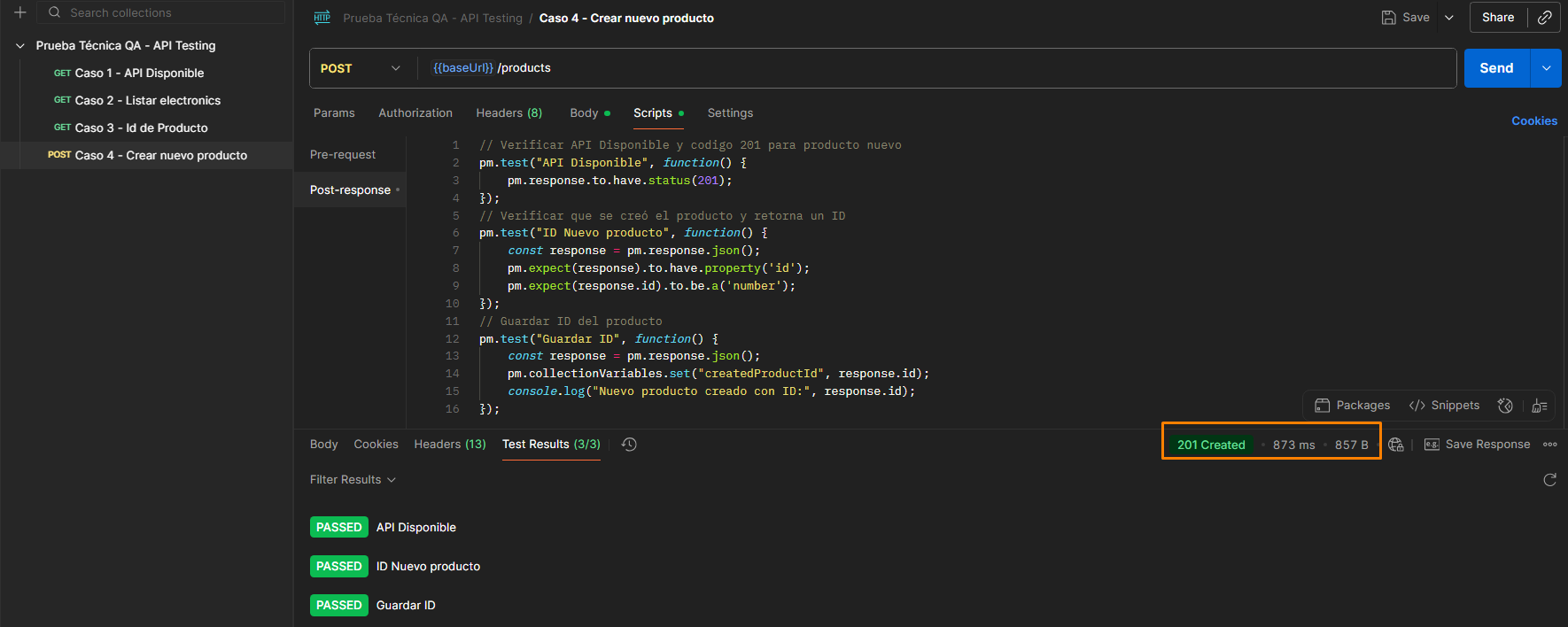
Captura de pantalla de un videojuego

El contenido generado por IA puede ser incorrecto.

* Array de producto creado



* Respuesta OK, tiempo y tamaño de la respuesta



RESULTADOS OBTENIDOS:

• Status Code: 201 CREATED

• Response Time: 873 ms

• Tamaño Response: 857 B

• Tests Ejecutados: 3/3 passed

• Resultado Final: Se realiza la creación exitosa del nuevo producto en la API.

* 1. **CASO 5 Validar Correcta Creación del producto creado**

Objetivo: Verificar que el producto creado en el Caso 4 persiste en el sistema.

Endpoint: /products/21

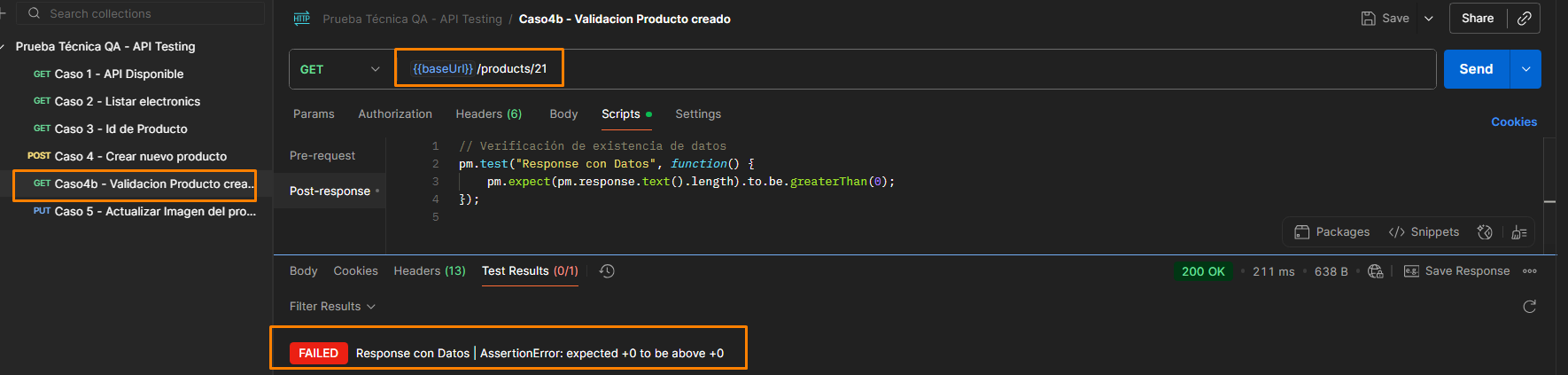
Metodo: GET

Datos de Entrada: N/A

Resutlado Esperado: Código de respuesta HTTP 200 y array de producto con ID = 21.

Paso a Paso:

* Consultar el estado actual del producto creado



RESULTADOS OBTENIDOS:

• Status Code: 200 OK

• Response Time: 211 ms

• Tamaño Response: 638 B

• Tests Ejecutados: 0/1 passed

• Resultado Final: Al enviar la petición para consultar el producto, Se recibió la respuesta "AssertionError: expected +0 to be above +0", lo que indica que el producto creado no persiste en la base de datos.

• Bug Asociado: 001

* 1. **CASO 6 Actualizar imagen del producto creado**

Objetivo: Verificar la correcta actualización de la imagen del producto creado.

Endpoint: /products/1

Metodo: PUT

Datos de Entrada: Nueva URL de imagen del producto

Resutlado Esperado: Código de respuesta HTTP 201 y array de producto actualizado.

Limitación técnica: Debido al BUG-001 (persistencia inexistente), el producto creado en el Caso 4 no existe realmente para operaciones posteriores.

Solución implementada:

Para demostrar la funcionalidad de actualización, se utilizará el producto existente ID 1, verificando que:

- El endpoint PUT /products/1 funciona correctamente

- La actualización de imagen se realiza exitosamente

- Los datos persisten después de la actualización

Paso a Paso:

* Metodo, URL y Body del nuevo producto

Captura de pantalla de computadora

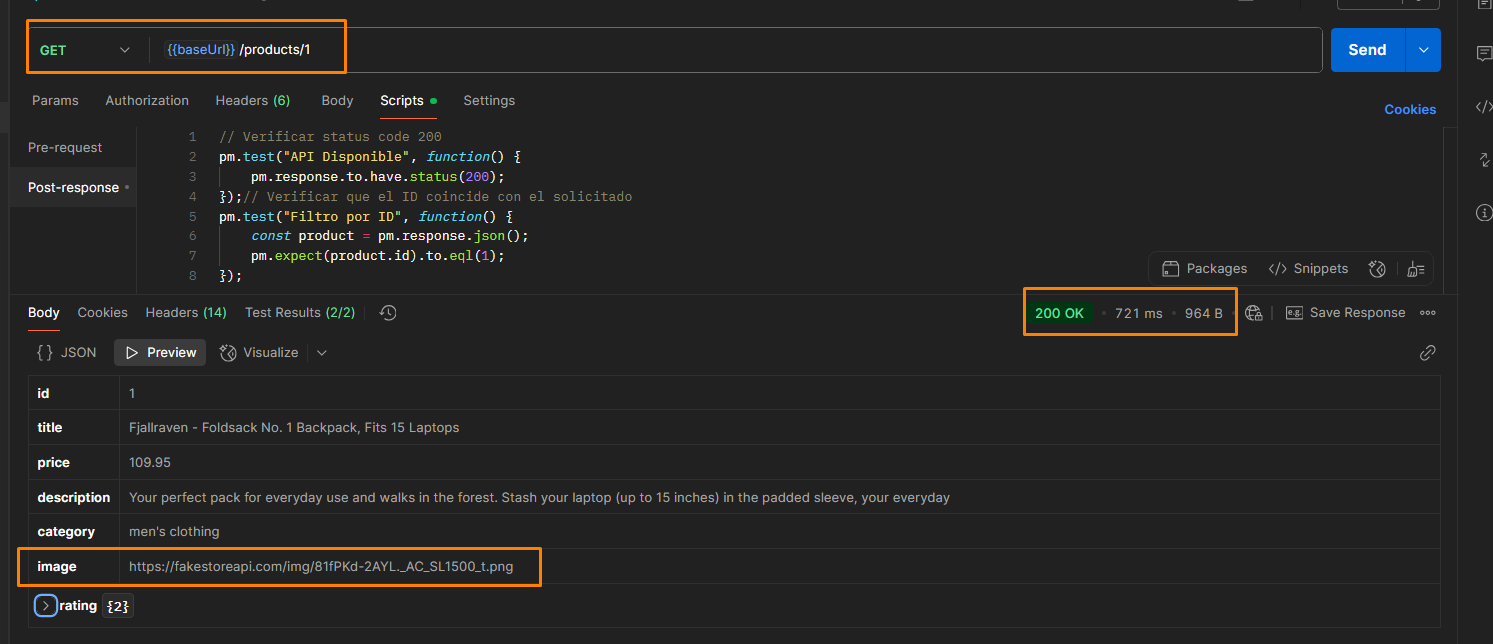
El contenido generado por IA puede ser incorrecto.

* Request y estado de los test ejecutados.

Captura de pantalla de computadora

El contenido generado por IA puede ser incorrecto.

* Validar persistencia de la imagen actualizada en el producto ID=1.



RESULTADOS OBTENIDOS:

• Status Code: 200 OK

• Response Time: 216 ms

• Tamaño Response: 731 B

• Tests Ejecutados: 2/2 passed

• Resultado Final: FALLIDO - Al enviar la petición para realizar la actualización de la imagen se recibe respuesta exitosa, pero al validar nuevamente si la URL se actualizo, la URL no persiste.

• Bug Asociado: 002