

Informe de Automatización de Pruebas en Postman

1. Introducción

Este documento detalla la automatización de pruebas realizada en Postman para validar el correcto funcionamiento de una API [DummyJson](#). Se han diseñado y ejecutado pruebas automatizadas utilizando Postman Runner, asegurando la integridad y fiabilidad de los endpoints probados.

2. Configuración del Entorno

Para la ejecución de las pruebas, se configuraron los siguientes elementos:

- **Colección de pruebas:** Incluye múltiples solicitudes para validar distintas funcionalidades de la API.
- **Entorno:** Configuración de variables de colección y de entorno para la correcta parametrización de los casos de prueba.

3. Ejecución de las Pruebas

Las pruebas se ejecutaron mediante Postman Runner siguiendo estos pasos:

1. Selección de la colección de pruebas.
2. Configuración del entorno adecuado.
3. Inicio de la ejecución mediante la opción "Run".
4. Observación y análisis de los resultados obtenidos.

4. Resultados

Tras la ejecución de las pruebas, se obtuvieron los siguientes resultados:

- **Número total de pruebas ejecutadas:** 92
- **Pruebas exitosas:** 91
- **Pruebas fallidas:** 1
- **Tiempo total de ejecución:** 7.5s.

Validaciones realizadas:

Los casos de prueba cubrieron diferentes escenarios, validando:

- ✓ Respuestas esperadas de la API
- ✓ Códigos de estado HTTP
- ✓ Estructura de los datos devueltos

Datos por Ítem

SUMMARY ITEM	TOTAL	FAILED
Requests	24	0
Prerequest Scripts	35	0
Test Scripts	46	0
Assertions	92	1
Skipped Tests	0	-

Imagen 01. Datos extraídos en newman.

Los casos de prueba cubrieron diferentes escenarios, validando respuestas esperadas, códigos de estado HTTP y la estructura de los datos devueltos por la API.

5. Observaciones y Mejoras

Durante la ejecución de las pruebas, se identificaron las siguientes áreas de mejora:

- Error en la validación de productos en **/products**.
 - Algunos productos no incluyen la propiedad **brand**, causando fallos en las pruebas automatizadas.
- Se recomienda:
 - Revisar la documentación de la API para confirmar si **brand** es opcional o si debe estar presente en todos los productos.
 - Modificar las pruebas para manejar productos sin **brand** y evitar fallos innecesarios.

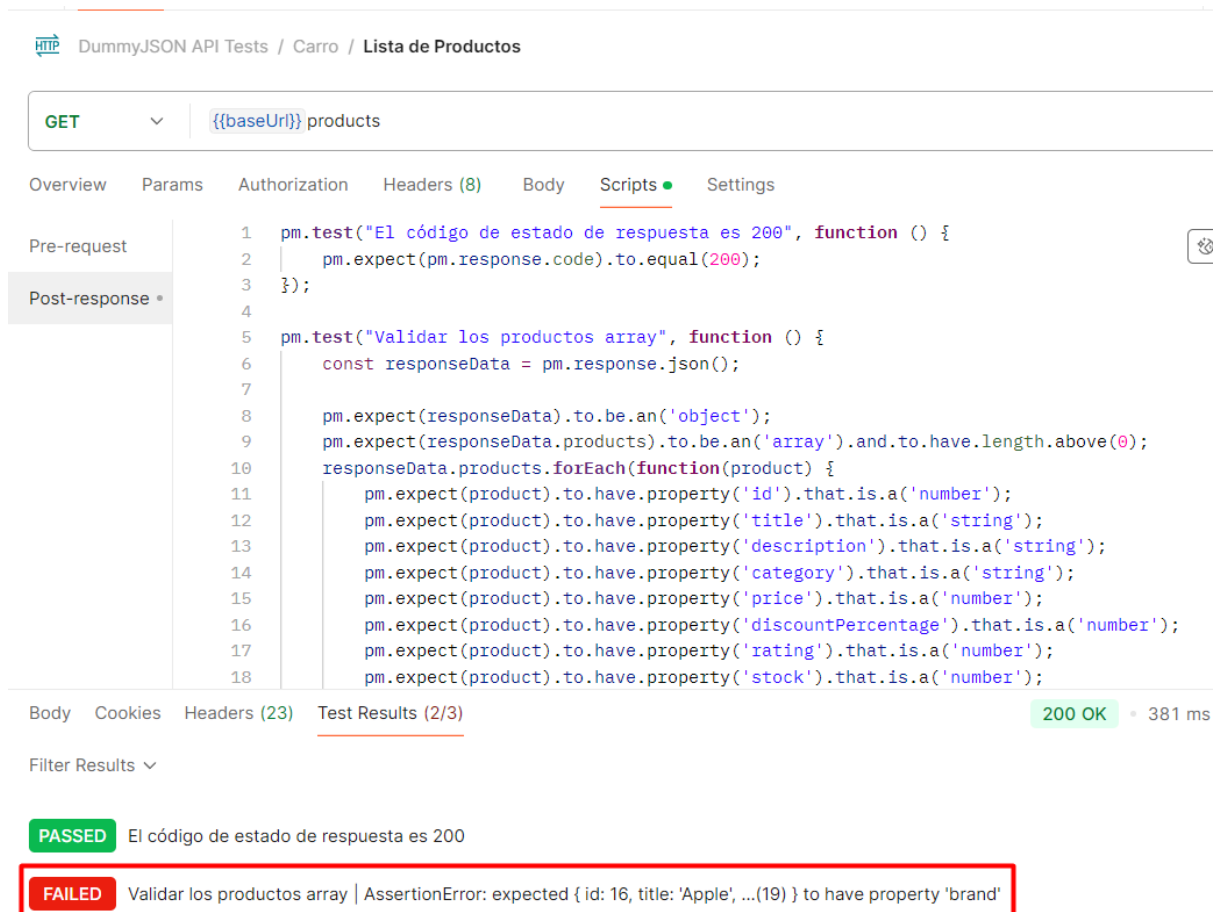


Imagen 02. Datos extraídos en Postman.

- **Limitaciones de DummyJSON**

- La API proporciona **datos estáticos**, lo que limita la validación de operaciones dinámicas como actualización o eliminación de registros.
- Se recomienda utilizar un entorno de prueba con datos que permitan modificaciones reales para evaluar completamente la API.

- **Mejoras en la documentación**

- Se sugiere mejorar la documentación de la API para incluir ejemplos más detallados y casos de uso específicos.

6. Conclusión

La ejecución automatizada de pruebas en **Postman** permitió validar la funcionalidad de la API de manera eficiente.

Se detectaron errores al obtener, actualizar o eliminar usuarios y carritos, ya que la API solo simula estos cambios sin realizarlos en el servidor. Se recomienda implementar las mejoras mencionadas para optimizar el funcionamiento de la API.

Este enfoque mejora la confianza en la API y facilita su mantenimiento.