
ACADEMY
Pruebas de API
con Postman

Pruebas de API con Postman

ELDAR ACADEMY

Versión: 1.0

Fecha: 28 de Diciembre de 2024

Objetivo

El objetivo de esta actividad es que el colaborador aprenda e investigue sobre pruebas básicas con API utilizando la herramienta de Postman, documentando el proceso y explique cómo se realizó.

Introducción

Para comenzar con esta actividad, vamos a desarrollar varias preguntas que serán indispensables saber antes de dar un ejemplo práctico.

¿Qué es una API?

API significa "**Interfaz de programación de aplicaciones**". Estas son un software que hacen el papel de intermediario entre dos aplicaciones, es decir la comunicación entre esas aplicaciones. Son un "puente" que facilita la interacción entre dos sistemas de forma que estos intercambien datos. Las API más populares y más utilizadas son las "REST" (Transferencia de estado representacional).

¿Con qué lenguaje se comunican las API?

El lenguaje con que se comunican las API se llama JSON (JavaScript Object Notation). Este es un formato de texto independiente del lenguaje, ideal para el intercambio de datos, siendo fácil de leer y escribir por seres humanos, simple de interpretar y generar para los equipos de cómputos.

¿Cómo interactúa con un servicio web?

La interacción con un servicio web o API web, es a través de los métodos o protocolos "HTTP", los cuales van a intercambiar información entre cliente y el servidor. Es decir, el servidor va a esperar una petición y si esta contiene el protocolo correcto, entonces devuelve al cliente la información solicitada.

La forma en como trabaja una API es mediante un "Contrato", es decir, va a tener una petición que contiene los headers (envía información), endpoints, body (parámetros). Cuando este servicio recibe la petición, envía un código de respuesta. Estos códigos por ejemplo pueden ser:

- Rango 100: Respuesta informativa.
- Rango 200: Respuesta correcta.
- Rango 300: Redirección.
- Rango 400: Error del cliente.
- Rango 500: Error del servidor.

Los "Verbos" más comunes para realizar peticiones son:

- GET: Permite la obtención de datos del servidor.
- POST: Crea nuevos elementos enviando información al servidor.
- PUT: Actualiza datos del servidor.
- DELETE: Elimina un dato o información del servidor.

¿Qué problema resuelven en QA?

En el ámbito QA las APIs son muy importantes, ya que garantizan que las aplicaciones puedan comunicarse entre sí sin errores, que los datos intercambiados sean correctos y que los servicios sean confiables. Algunos ejemplos de problemas que resuelven:

- *Errores de integración:* QA garantiza que las APIs puedan integrarse correctamente con otros sistemas.
- *Validación de datos:* Al intercambiar datos constantemente, las pruebas de QA van a ayudar a asegurar que los datos enviados y recibidos sean consistencia, en el formato esperado y cumplan con los requisitos del negocio/cliente.
- *Rendimiento:* QA asegura que las APIs sean escalables y respondan adecuadamente bajo diferentes niveles de carga.

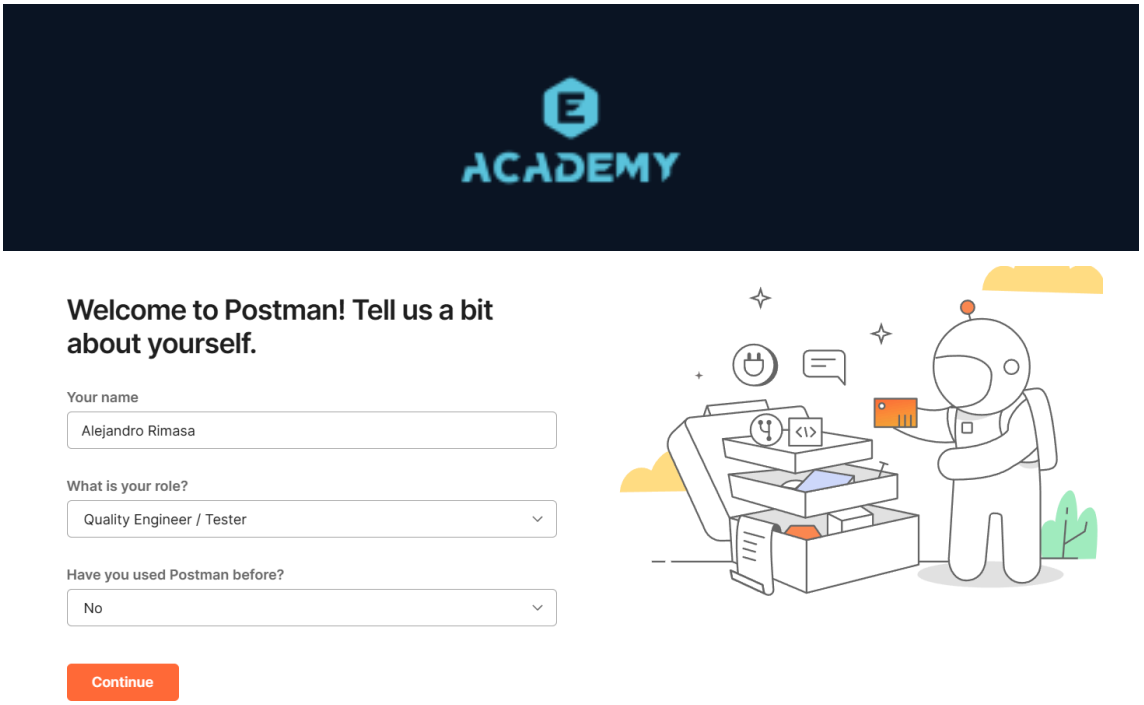
¿Qué es Postman?

Es una herramienta que sirve para construir y utilizar APIs, simplifican el ciclo de vida de la creación de APIs mejores y más rápido. Estas se consumen a través de las peticiones HTTP, (esta es la relación que existe entre postman y las peticiones.)

Instalación y configuraciones

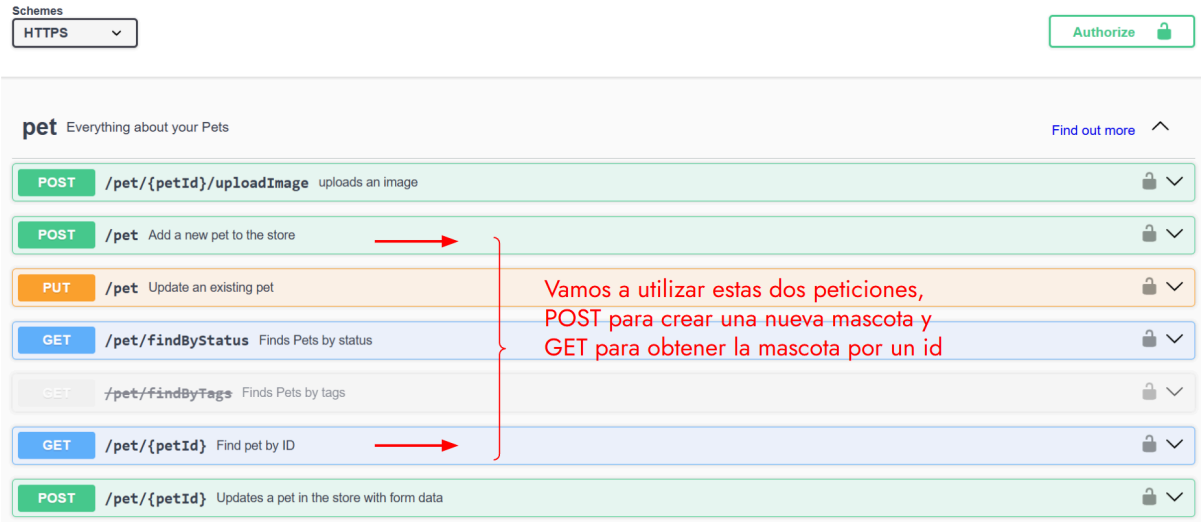
Descargar el instalador de Postman para el sistema que se desea. Una vez instalado, se debe crear una cuenta en postman. (Es opcional esto ya que se puede trabajar con Postman por la página web.)

Complete los campos requeridos:

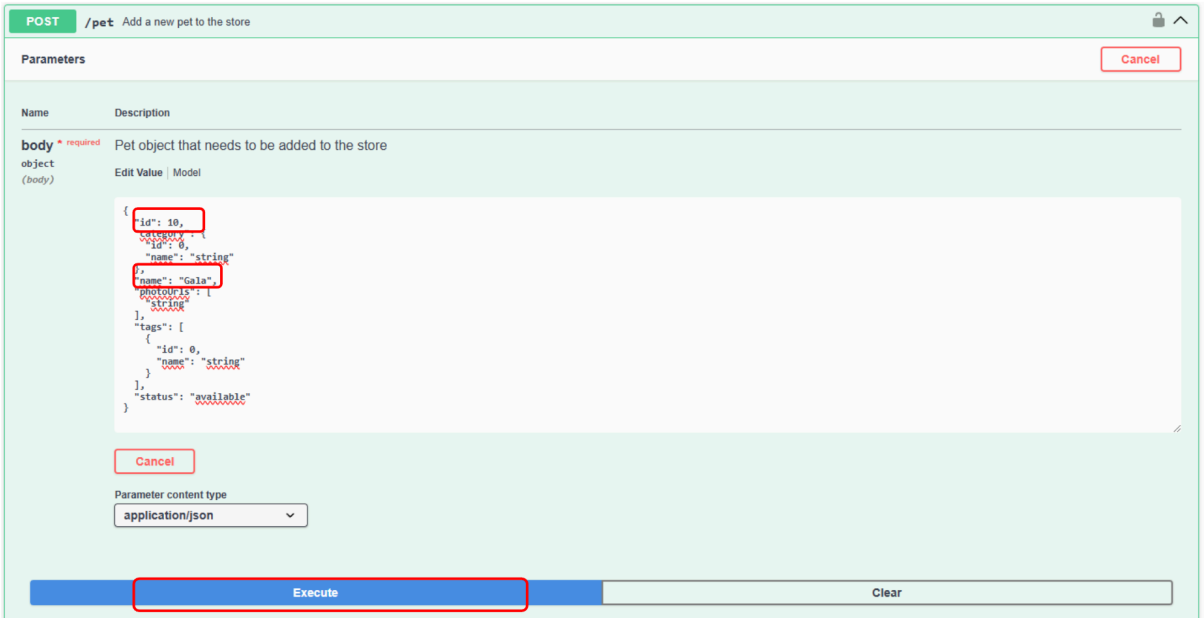


Demostración del uso de Postman: Capturas del proceso de solicitud y respuestas obtenidas. Ejemplo

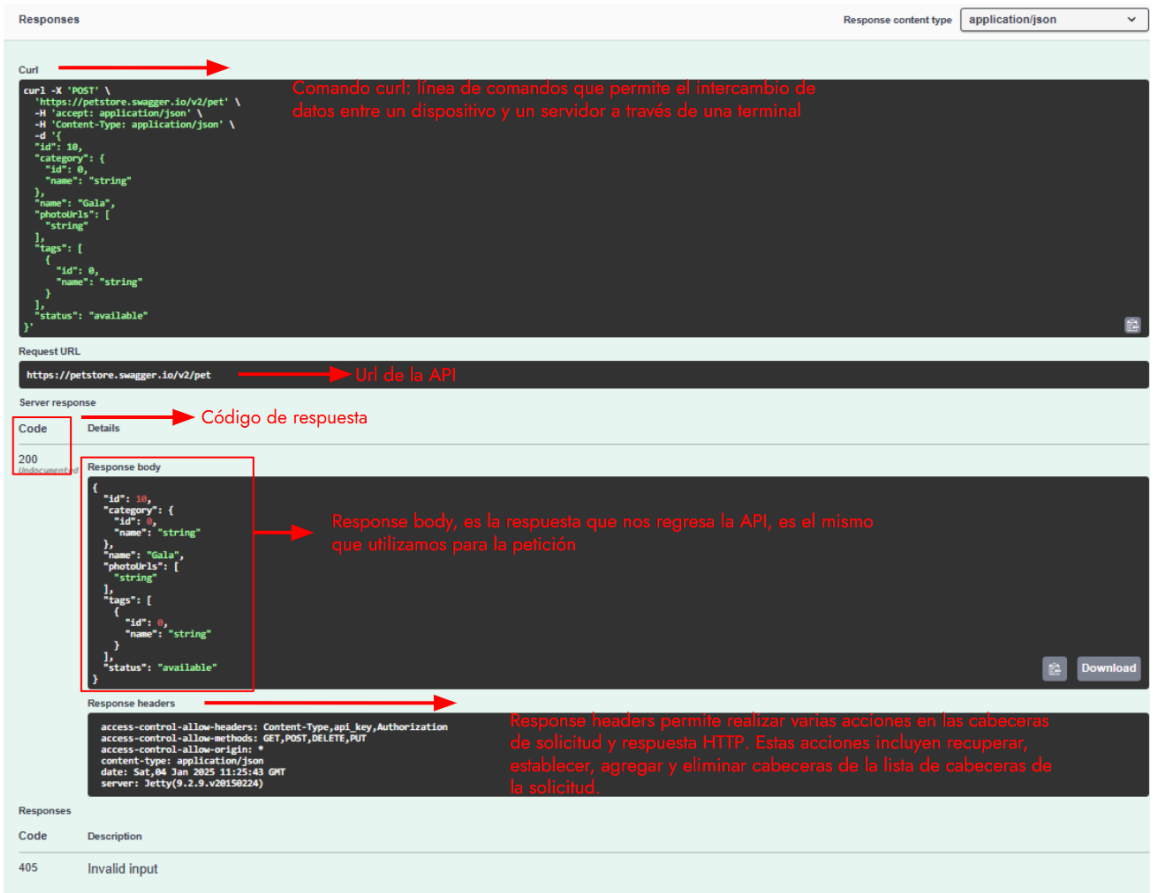
Para realizar el ejemplo voy a utilizar una página demo de una API para tiendas de mascotas https://petstore.swagger.io/#/



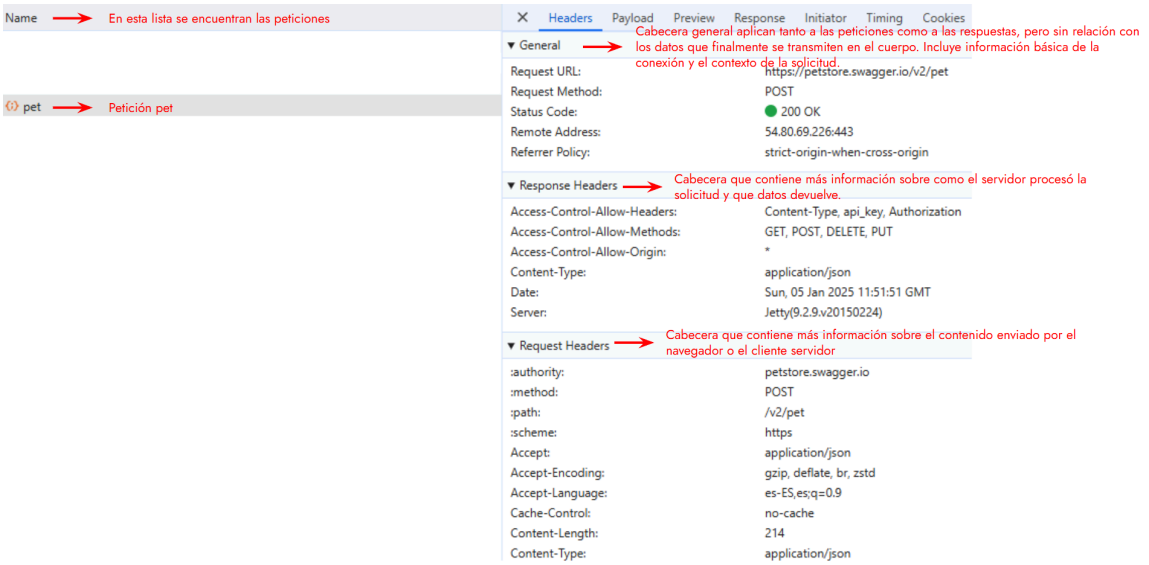
Primero vamos a trabajar con la petición de “POST” para ello debemos abrir el desplegable y nos van a aparecer unos parámetros por defecto, el cual edité el “ID” y el “Name”. Seguido de esto vamos a seleccionar el botón “Execute”.

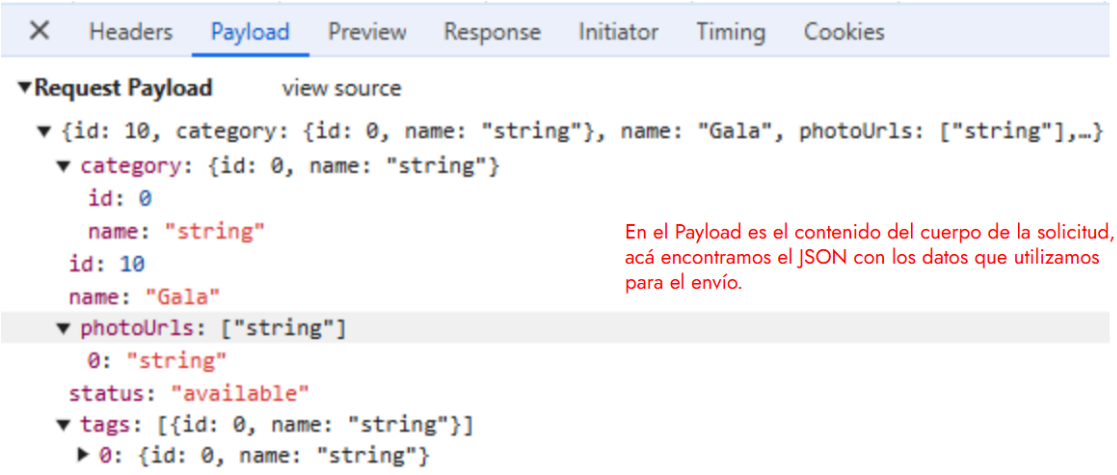


Al seleccionar el botón "Execute", se nos abrirá el comando "**Curl**", la "**url**" de la API, el "**Código de respuesta**" y el "**Response body**" que es la respuesta que nos trae la API.



Cuando una API se consume desde otra aplicación, visualmente no se pueden detectar lo que ocurre en el intercambio de datos, la forma en que se pueda saber si se están conectando o cuales son los elementos del request que vienen del response es abriendo la consola de las "Devtools" y dirigirse a la solapa de "Network", algunos de los siguientes elementos permiten entender y depurar cómo se está comunicando la app con la API. En este caso encontraremos en el "**Request Method": POST** y el "**Status Code": 200**, lo que significa que la solicitud fue procesada exitosamente por el servidor.

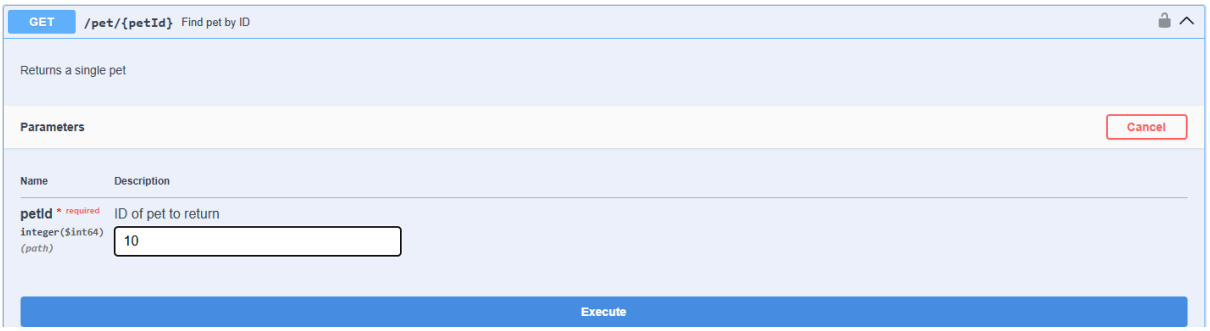




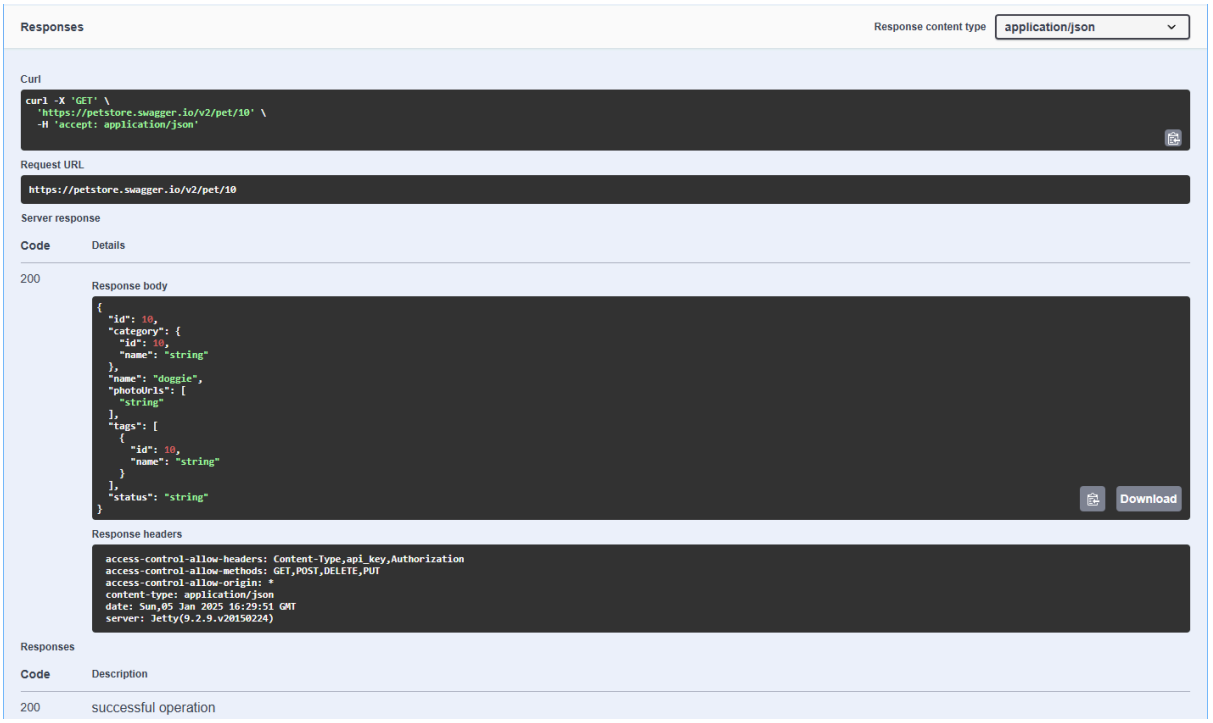
En el Payload es el contenido del cuerpo de la solicitud, acá encontramos el JSON con los datos que utilizamos para el envío.

Response muestra el contenido de la respuesta del servidor como fue recibido, puede ser en texto plano como JSON, HTML, XML, ect.

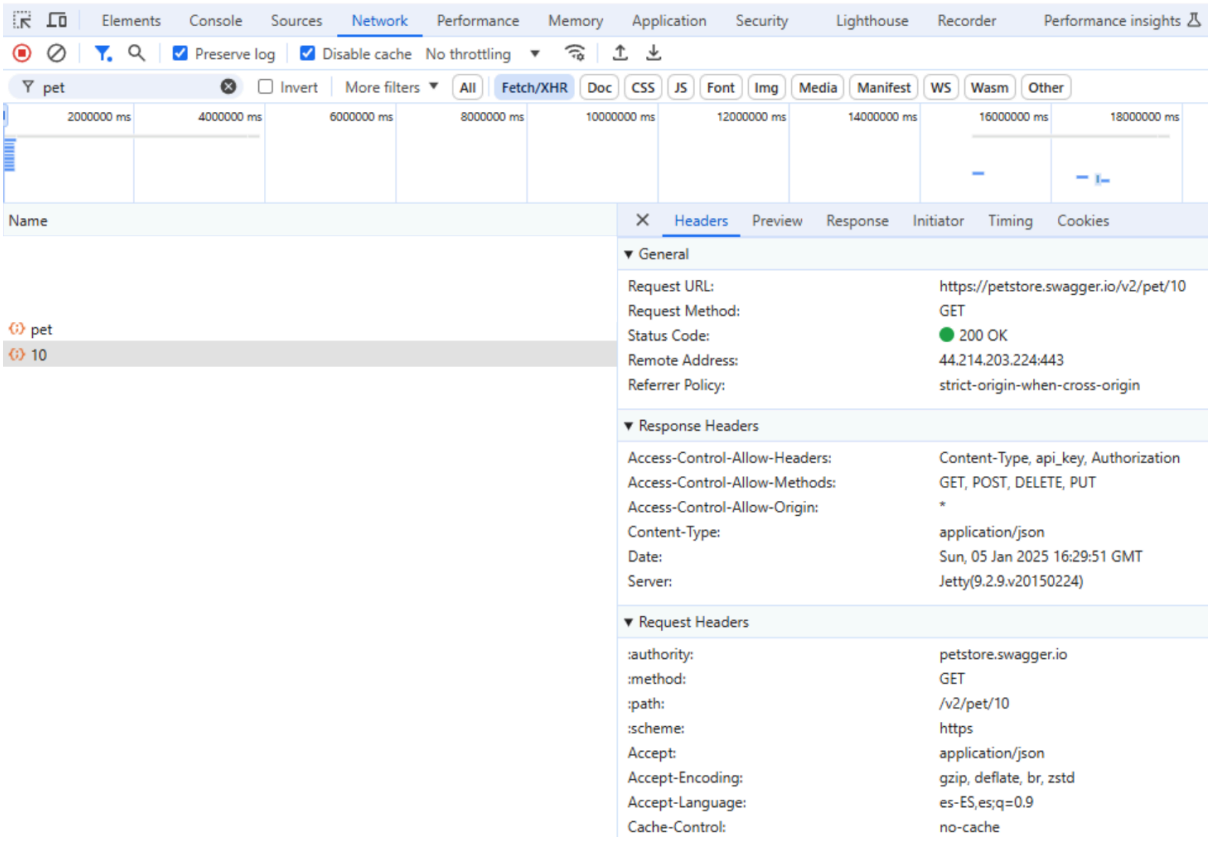
Seguido de la petición POST, vamos a trabajar con la petición GET esta nos va a servir para obtener información de un recurso, para ello debemos abrir el desplegable y en este caso nos va a pedir el "ID" de la mascota. Seguido de esto vamos a seleccionar el botón "Execute".



La respuesta va a ser el ID y el nombre que habíamos colocado anteriormente.

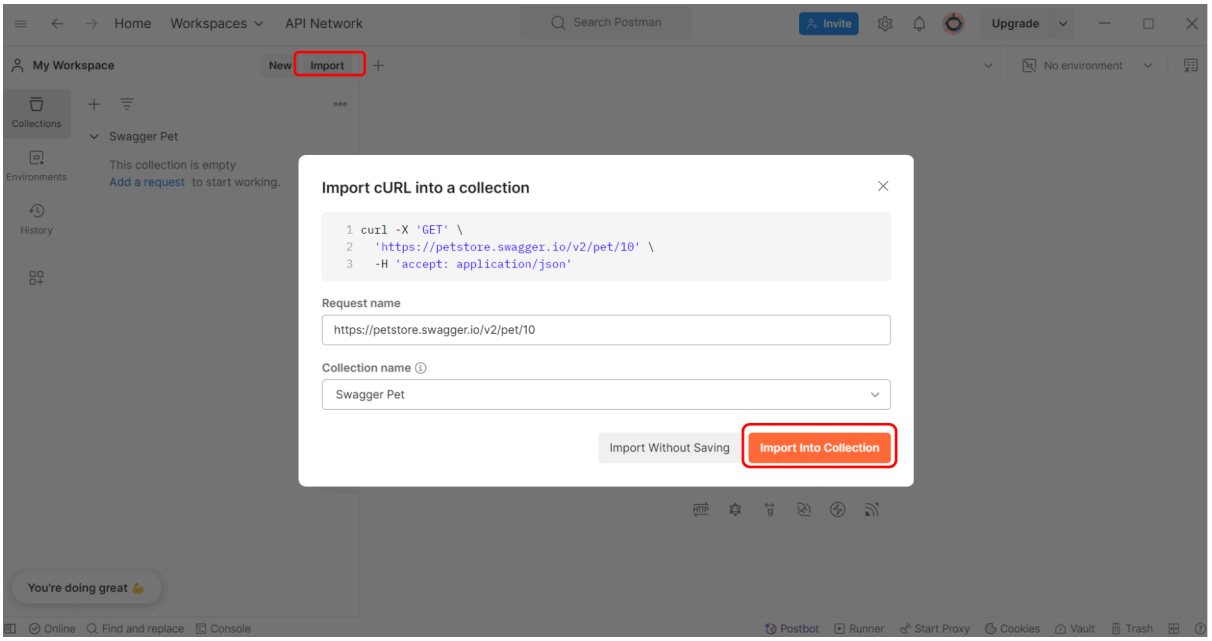


En nuestra consola del navegador nos va a mostrar la petición, en este caso encontraremos en "**Request Method**" : **GET** y el el "**Status Code**" : **200**, lo que significa que la solicitud fue procesada exitosamente por el servidor.

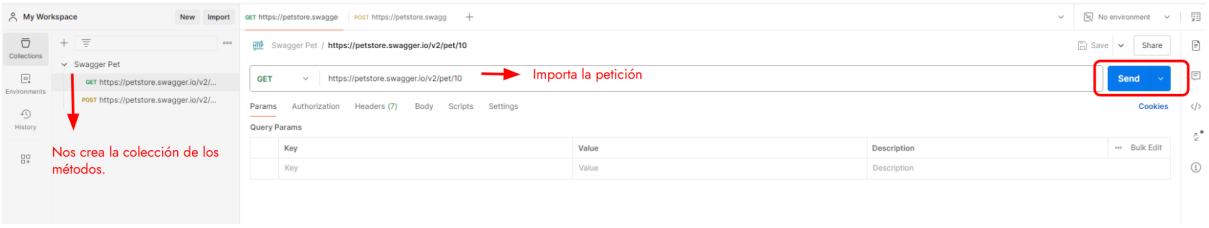


¿Cómo realizar una petición en POSTMAN?

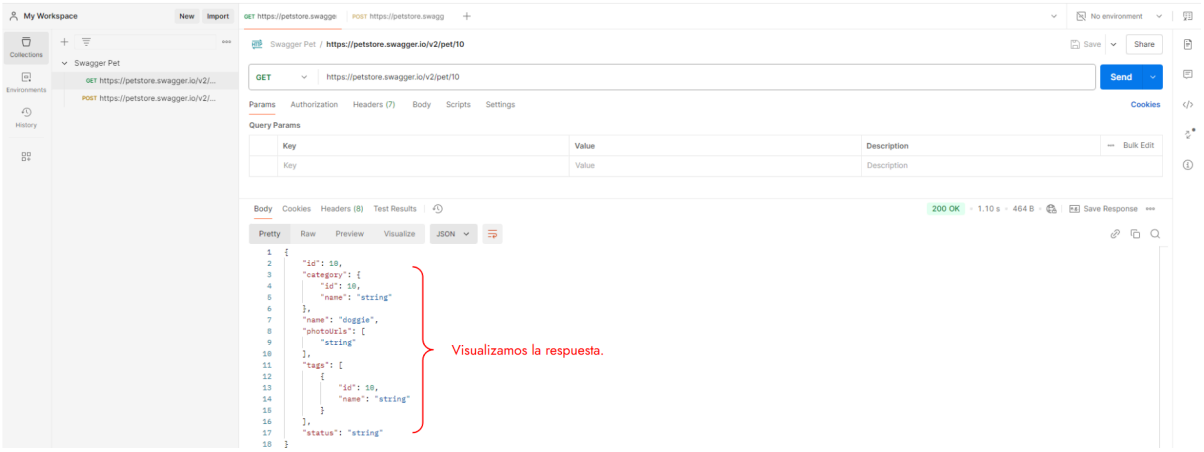
Vamos a copiar el comando "**CURL**", seguido de esto lo importamos en Postman y se nos abre un pop-up donde pegamos el comando, seleccionamos importar en una colección y automáticamente Postman nos importa esta petición **GET**. Este mismo paso lo vamos a realizar para la petición **POST**.



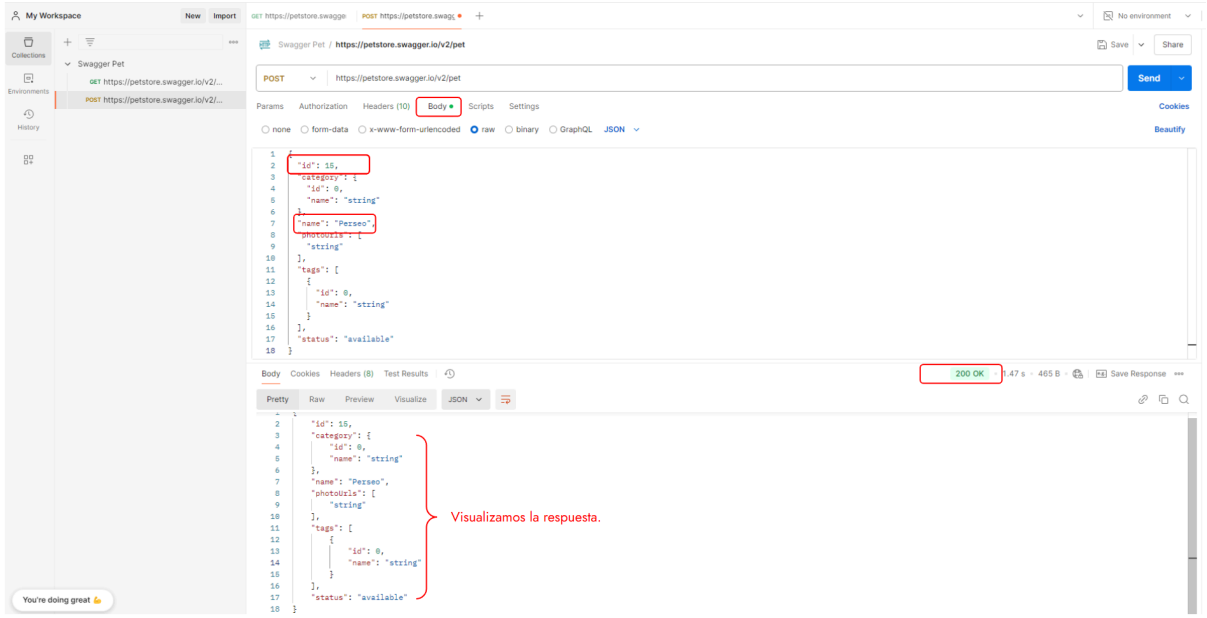
Postman nos va a crear la colección, seleccionamos "**SEND**"



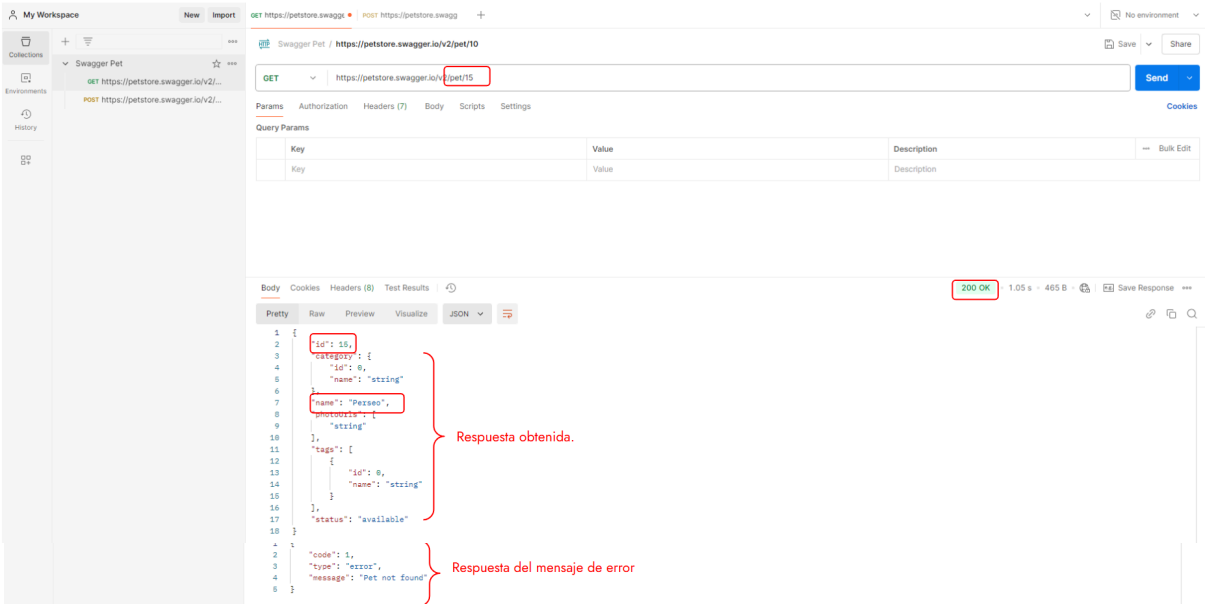
Al seleccionar "Send" nos va a mostrar la respuesta:



Para la petición "**POST**" vamos a realizar los mismos pasos, pero primero debemos dirigirnos a la solapa del "**Body**" para crear una nueva mascota, cambiando el número del "**ID**" y el "**Name**", seleccionamos "**SEND**" y nos debería de mostrar el código de respuesta 200, este código nos está diciendo que creó la mascota.



Si queremos realizar una prueba manual, para saber si esta mascota fue creada, nos vamos a la petición **"GET"** y cambiamos el número de **"ID"** de la importación para obtener ese dato y seguido de esto seleccionamos "**SEND**".



Conclusiones y recomendaciones

En conclusión, el uso de la herramienta Postman nos permite comprender como funcionan las APIs en la comunicación entre las aplicaciones, garantizando que las solicitudes se procesan correctamente y que los datos intercambiados sean consistentes y confiables, reflejando así una base para el uso de herramientas de pruebas en QA.

Como recomendación se podría seguir investigando las pruebas automatizadas ya que esto sería muy beneficioso aprovechar las capacidades de automatización que tiene Postman, permitiendo así realizar pruebas repetitivas de manera más eficiente y reducir tiempos en validaciones.

Bibliografía

Videos

https://www.youtube.com/watch?v=PvAsMgbRs5w&t=799s

https://www.youtube.com/watch?v=fTtWVCortxo&list=PLDbrnXa6SAzUsLG1gjECgFYLSZDov09fO&index=4 (03. JSON)

https://www.youtube.com/watch?v=inNR5enqRq0&list=PLDbrnXa6SAzUsLG1gjECgFYLSZDov09fO&index=4 (04. Protocolo HTTP)

https://www.youtube.com/watch?v=ooRGyFzH8Ww&list=PLDbrnXa6SAzUsLG1gjECgFYLSZDov09fO&index=5 (05. Postman)

Otros

https://www.redhat.com/es/topics/api/what-are-application-programming-interfaces

https://www.qatouch.com/blog/api-testing/

https://qalified.com/es/blog/api-testing/

https://lab.wallarm.com/what/pruebas-api/

https://es.abstracta.us/blog/api-testing-guia-practica/

https://blog.hubspot.es/website/comando-curl

https://developer.mozilla.org/es/docs/Web/API/Headers

https://developer.mozilla.org/es/docs/Web/HTTP/Headers

https://petstore.swagger.io/#/pet