



Casos prácticos con POSTMAN

- **Probar rutas dinámicas:** Si tu API tiene una ruta como `/item/:id`, puedes probarla ingresando una URL como `http://localhost:3000/item/12345` y verificando la respuesta.
- **Simular errores:** Prueba enviar datos incompletos o en formatos incorrectos para ver cómo responde tu API a errores comunes. Esto te ayudará a mejorar el manejo de errores y validar solicitudes.
- **Automatizar pruebas:** POSTMAN tiene una funcionalidad llamada Tests, donde puedes escribir pequeños scripts en JavaScript para verificar automáticamente las respuestas, cómo asegurarte de que el código de estado sea 200 o que ciertos campos existan en el JSON de respuesta.

Ejercicio Práctico

Configuración de Rutas y Manejo de Parámetros

Sabrina y Matías se acercan a revisar tu progreso. Matías sonríe con aprobación: "Has avanzado muchísimo, pero ahora es momento de subir un nivel. Necesitamos que configures la capa de rutas de manera profesional."

Sabrina se cruza de brazos y asiente. "Exacto. Queremos que aprendas a organizar tus rutas, además de implementar un par de configuraciones clave como **CORS** y un manejo de errores para rutas inexistentes. Esto te ayudará a preparar tu servidor para escenarios reales."



Matías continúa: "Pero eso no es todo. También queremos que practiques cómo manejar datos dinámicos que llegan a través de las solicitudes. Deberás configurar rutas que lean tanto **path params** como **query params** y luego devolver esa información en las respuestas. Y no te olvides de usar **POSTMAN** para validar todo."



Misión:

1. Establece 4 rutas en el archivo de entrada de la aplicación. Asegúrate de:
 - Permitir solicitudes entre dominios configurando **CORS**.
 - Agregar un middleware que capture las rutas no encontradas y devuelva una respuesta 404.
2. Define rutas que utilicen:
 - **Path params**: Por ejemplo, una ruta que devuelva información basada en el ID de un producto o usuario.
 - **Query params**: Una ruta que filtre datos con base en parámetros como categoría o precio.

Usa **POSTMAN** para probar las rutas que configuraste. Asegúrate de capturar correctamente los parámetros enviados y mostrar la información de manera clara en las respuestas.

Sabrina sonríe y agrega: "Recuerda, la organización es clave. Cuando logres que todo funcione, será el momento perfecto para revisar cómo cada pieza encaja en el proyecto. ¡Buena suerte!"

Tu desafío está claro. Ahora es momento de implementar lo aprendido y demostrar tus habilidades.

Materiales y Recursos Adicionales:

Documentación Oficial de Express: expressjs.com

Explora en detalle cómo configurar rutas, middlewares y manejar errores en Express.

Postman Learning Center: learning.postman.com

Guías interactivas para aprender a usar POSTMAN, desde conceptos básicos hasta funciones avanzadas como automatización y pruebas.



Artículo: “Understanding CORS”: [Mozilla Developer Network \(MDN\)](#)

Una guía detallada sobre qué es CORS, cómo funciona y cómo configurarlo correctamente.

Repositorio de Ejemplo: Explora ejemplos básicos de configuración de servidores Express en GitHub ([Node.js Express Examples](#)).

Preguntas para Reflexionar:

- ¿Por qué es importante organizar la estructura de un proyecto con capas como rutas, controladores y modelos? ¿Qué beneficios aporta a largo plazo?
- ¿Cómo puedes garantizar que tu API sea accesible para aplicaciones que no comparten el mismo dominio mediante el uso de CORS?
- ¿Qué diferencias encuentras entre **path params** y **query params**? ¿En qué escenarios usarías cada uno?
- ¿Cómo manejarías la validación de parámetros de ruta o consulta para asegurarte de que solo se procesen datos válidos en tu API?
- ¿Cómo contribuye el manejo de errores 404 y otras respuestas HTTP claras a mejorar la experiencia del usuario y el desarrollo de tu aplicación?

Próximos Pasos:

- **Capa lógica:** controlando la respuesta de nuestra aplicación.
- **Modelo de datos y trabajo con JSON:** consultamos datos de forma interna y los devolvemos al cliente desde nuestra API Rest.
- **Datos en la nube:** Configurando y accediendo a datos en un servidor externo.



Buenos Aires
aprende
Agencia de Promoción y Desarrollo

BA Buenos
Aires
Ciudad