

Casos de prueba creación de usuario

ID del caso de prueba	Descripción del caso	Datos de entrada (JSON)	Resultado esperado	Código de estado esperado	Comentarios	Estatus	Resultado actual
CT01	Crear usuario con datos válidos	{"username": "testUser", "email": "john.doe@example.com", "password": "password123", "phone": "123-456-7890", "userStatus": 1}	El usuario se crea correctamente, y la respuesta contiene el campo username igual a testUser.	200	Este caso verifica la creación correcta de un usuario con todos los datos válidos.	APROBADO	Se crea el usuario y se obtiene respuesta 200
CT02	username vacío	{"username": "", "email": "john.doe@example.com", "password": "password123", "phone": "123-456-7890", "userStatus": 1}	Error indicando que username es obligatorio.	400	Se espera un error porque el campo username está vacío.	NO APROBADO	Se crea el usuario y se obtiene respuesta 200
CT03	username con caracteres no permitidos	{"username": "@username\$", "email": "john.doe@example.com", "password": "password123", "phone": "123-456-7890", "userStatus": 1}	Error indicando que username contiene caracteres no permitidos.	400	Se espera un error porque el campo username tiene caracteres especiales no válidos.	NO APROBADO	Se crea el usuario y se obtiene respuesta 200
CT04	email con formato incorrecto	{"username": "testUser", "email": "Pruebasexample.com", "password": "password123", "phone": "123-456-7890", "userStatus": 1}	Error indicando que el formato de email es inválido.	400	Se espera un error debido al formato incorrecto del email.	NO APROBADO	Se crea el usuario y se obtiene respuesta 200
CT05	password demasiado corto	{"username": "testUser", "email": "john.doe@example.com", "password": "string", "phone": "123-456-7890", "userStatus": 1}	Error indicando que password debe tener al menos 8 caracteres.	400	Se espera un error debido a que el password es demasiado corto.	NO APROBADO	Se crea el usuario y se obtiene respuesta 200
CT06	password con solo números	{"username": "testUser", "email": "john.doe@example.com", "password": "12345678", "phone": "123-456-7890", "userStatus": 1}	El usuario se crea correctamente si cumple con los requisitos.	200	Aunque el password es solo numérico, es aceptado porque tiene al menos 8 caracteres.	APROBADO	Se crea el usuario y se obtiene respuesta 200
CT07	phone con formato incorrecto	{"username": "testUser", "email": "john.doe@example.com", "password": "password123", "phone": "123.456.7890", "userStatus": 1}	Error indicando que el formato de phone es incorrecto.	400	Se espera un error debido al formato incorrecto del teléfono (puntos en lugar de guiones).	NO APROBADO	Se crea el usuario y se obtiene respuesta 200
CT08	userStatus fuera del rango (valor 2)	{"username": "testUser", "email": "john.doe@example.com", "password": "password123", "phone": "123-456-7890", "userStatus": 2}	Error indicando que userStatus debe estar en el rango de 0 a 1.	400	Se espera un error porque el valor de userStatus está fuera del rango permitido.	NO APROBADO	Se crea el usuario y se obtiene respuesta 200
CT09	userStatus vacío	{"username": "testUser", "email": "john.doe@example.com", "password": "password123", "phone": "123-456-7890", "userStatus": ""}	Error indicando que userStatus no puede estar vacío.	400	Se espera un error porque el campo userStatus no puede ser vacío.	APROBADO	Se obtiene la respuesta 400

Acceso mediante usuario y contraseña								
ID	Descripción	Método HTTP	URL	Parámetros	Resultado esperado	Pruebas	Estado	Resultado actual
1	Login exitoso con credenciales válidas.	GET	https://petstore.swagger.io/v2/user/login?username=Usuario&password=Perro1234	username=Usuario, password=Perro1234	Código 200 OK, respuesta JSON con un mensaje de éxito, como "Login successful".	Verificar que el código de estado es 200.	APROBADO	La respuesta es 200
						Verificar que la respuesta es JSON.		
						Verificar que el mensaje en la respuesta es "Login successful".		
2	Login fallido con credenciales incorrectas. Caracteres invalidos	GET	https://petstore.swagger.io/v2/user/login?username=Üusuario&password=Perro1234	username=Üusuario, password=Perro1234	Código 401 Unauthorized, respuesta JSON con mensaje de error como "Invalid credentials".	Verificar que el código de estado es 401.	NO APROBADO	La respuesta es 200
						Verificar que la respuesta es JSON.		
						Verificar que el mensaje en la respuesta es "Invalid credentials".		
3	Login fallido sin nombre de usuario.	GET	https://petstore.swagger.io/v2/user/login?username=&password=l	username= (vacío), password=Perro1234	Código 400 Bad Request, mensaje de error que indica que el campo username es obligatorio.	Verificar que el código de estado es 400.	NO APROBADO	La respuesta es 200
						Verificar que la respuesta es JSON.		
						Verificar que el mensaje en la respuesta indica que el username es obligatorio.		
4	Login fallido sin contraseña.	GET	https://petstore.swagger.io/v2/user/login?username=l&password=	username=Usuario, password=	Código 400 Bad Request, mensaje de error que indica que el campo password es obligatorio.	Verificar que el código de estado es 400.	NO APROBADO	La respuesta es 200
						Verificar que la respuesta es JSON.		
						Verificar que el mensaje en la respuesta indica que la password es obligatoria.		
5	Login fallido con campos de entrada vacíos.	GET	https://petstore.swagger.io/v2/user/login?username=&password=	username= (vacío), password= (vacío)	Código 400 Bad Request, mensaje de error que indica que los campos username y password son obligatorios.	Verificar que el código de estado es 400.	NO APROBADO	La respuesta es 200
						Verificar que la respuesta es JSON.		
						Verificar que la respuesta indica que los campos son obligatorios.		
6	Login exitoso con credenciales de usuario diferente.	GET	https://petstore.swagger.io/v2/user/login?username=user123&password=secretpass	username=user123, password=secretpass	Código 200 OK, respuesta JSON con un mensaje de éxito, como "Login successful".	Verificar que el código de estado es 200.	APROBADO	La respuesta es 200
						Verificar que la respuesta es JSON.		
						Verificar que el mensaje en la respuesta es "Login successful".		

Listar Mascotas								
Casos de prueba positivos								
ID de	Descripción	Métod	URL	Parámetro status	Resultado esperado	Pasos para ejecutar	Resultado actual	Comentarios
1	Obtener todas las mascotas con estado "available".	GET	https://petstore.swagger.io/v2/pet/findByStatus?status=available	available	Respuesta con código 200 OK. La respuesta debe contener solo mascotas con estado available.	1. Abre Postman.	APROBADO	Se muestra como respuesta 200
						2. Crea una nueva solicitud GET.		
						3. Pega la URL https://petstore.swagger.io/v2/pet/findByStatus?status=available.		
						4. Enviar la solicitud.		
						5. Verifica que el estado de las mascotas sea available.		
2	Obtener todas las mascotas con estado "pending".	GET	https://petstore.swagger.io/v2/pet/findByStatus?status=pending	pending	Respuesta con código 200 OK. La respuesta debe contener solo mascotas con estado pending.	1. Abre Postman.	APROBADO	Se muestra como respuesta 200
						2. Crea una nueva solicitud GET.		
						3. Pega la URL https://petstore.swagger.io/v2/pet/findByStatus?status=pending.		
						4. Enviar la solicitud.		
						5. Verifica que el estado de las mascotas sea pending.		
3	Obtener todas las mascotas con estado "sold".	GET	https://petstore.swagger.io/v2/pet/findByStatus?status=sold	sold	Respuesta con código 200 OK. La respuesta debe contener solo mascotas con estado sold.	1. Abre Postman.	APROBADO	Se muestra como respuesta 200
						2. Crea una nueva solicitud GET.		
						3. Pega la URL https://petstore.swagger.io/v2/pet/findByStatus?status=sold.		
						4. Enviar la solicitud.		
						5. Verifica que el estado de las mascotas sea sold.		
4	Obtener mascotas con estados "available", "pending", "sold".	GET	https://petstore.swagger.io/v2/pet/findByStatus?status=available,pending,sold	available,pending,sold	Respuesta con código 200 OK. La respuesta debe contener mascotas con cualquiera de los tres estados.	1. Abre Postman.	APROBADO	Se muestra como respuesta 200
						2. Crea una nueva solicitud GET.		
						3. Pega la URL https://petstore.swagger.io/v2/pet/findByStatus?status=available,pending,sold.		
						4. Enviar la solicitud.		
						5. Verifica que el estado de las mascotas sea uno de los tres: available, pending, sold.		

Casos de prueba negativos								
ID de	Descripción	Métod	URL	Parámetro status	Resultado esperado	Pasos para ejecutar	Resultado actual	Comentarios
5	Intentar obtener mascotas con un estado inválido "invalid".	GET	https://petstore.swagger.io/v2/pet/findByStatus?status=invalid	invalid	Respuesta con código 400 o 500 (error), indicando que el valor del parámetro status es inválido.	1. Abre Postman.	NO APROBADO	Se muestra como resultado 200
						2. Crea una nueva solicitud GET.		
						3. Pega la URL https://petstore.swagger.io/v2/pet/findByStatus?status=invalid.		
						4. Enviar la solicitud.		
						5. Verifica que la respuesta contenga un mensaje de error indicando que el estado no es válido.		
6	Intentar obtener mascotas sin pasar el parámetro status.	GET	https://petstore.swagger.io/v2/pet/findByStatus	(vacío)	Respuesta con código 400 o 500 (error), indicando que el parámetro status es obligatorio.	1. Abre Postman.	NO APROBADO	Se muestra como resultado 200 y mostrando el resultado de available, sold y pending
						2. Crea una nueva solicitud GET.		
						3. Pega la URL https://petstore.swagger.io/v2/pet/findByStatus.		
						4. Enviar la solicitud.		
						5. Verifica que la respuesta contenga un mensaje de error indicando que el parámetro status es obligatorio.		