

Lenguaje
Rest

Intentos
0 de 30

Puntaje requerido
90%

Enunciado
RETO 4

En el reto 4, el equipo desarrollará los servicios de backend para realizar edición y borrado de las entidades, ahora, los servicios que serán implementados son los servicios PUT y DELETE. Se debe incluir en el frontend la posibilidad de editar las entidades y permitir el borrado de las mismas. Posteriormente se implementará el ejercicio de autenticación con oauth2.

El borrado de las entidades se hará a través de una petición DELETE y en este caso, el id del elemento borrado irá en la URL (así como en las peticiones GET).

Al finalizar este reto, tendremos funcionando nuestro propio backend con nuevas funcionalidades. Las funcionalidades que estarán resueltas al finalizar este reto son:

-Edición de categorías (endpoint: /api/Category/update).

-Edición de Palco (endpoint: /api/Box/update).

-Edición de Clientes (endpoint /api/Client/update).

-Edición de Mensajes (endpoint /api/Message/update).

-Edición de Reservas. (endpoint /api/Reservation/update).

-Edición de la calificación de las Reservas. (endpoint /api/Score/update).

-Edición de usuarios administrativos. (endpoint /api/Admin/update).

-Eliminación de Palco (endpoint: /api/Box/{id}).

-Eliminación de Categorías (endpoint: /api/Category/{id}).

- Eliminación de Mensajes(endpoint /api/Message/{id}).

-Eliminación de Reservas. (endpoint /api/Reservation/{id}).

-Eliminación de usuarios administrativos.(endpoint /api/Admin/{id}).

Historias de usuario:

Hisotoria de usuario Nro.	1		
Título:	Edición de palcos		
Descripción	COMO:	Usuario	
	QUIERO:	Ingresar los valores de ubicación, nombre, capacidad, descripción y categoría	
	PARA:	Poder modificar la información un palco en el sistema	
Criterios de aceptación	Los valores de ubicación y nombre deben ser un texto de no más de 45 caracteres.		

	Los valores de capacidad deben ser un número que representa el aforo máximo. Los valores de descripción deben ser un texto de máximo 250 caracteres. El valor de categoría no debe poderse modificar. El id de un palco no se debe poder modificar.
--	--

Hisotoria de usuario Nro.	2		
Título:	Edición de categorías		
Descripción	COMO:	Usuario	
	QUIERO:	Ingresar los valores de nombre y descripción	
	PARA:	Poder modificar una categoría existente en el sistema	
Criterios de aceptación	Los valores de nombre deben ser un texto de no más de 45 caracteres. Los valores de descripción deben ser un texto de máximo 250 caracteres. El id no se debe poder modificar		

Hisotoria de usuario Nro.3							
Título:	Edición de Clientes						
Descripción	<table><tr><td>COMO:</td><td>Usuario</td></tr><tr><td>QUIERO:</td><td>Ingresar los valores de nombre, edad y contraseña</td></tr><tr><td>PARA:</td><td>Poder modificar un cliente existente en el sistema</td></tr></table>	COMO:	Usuario	QUIERO:	Ingresar los valores de nombre, edad y contraseña	PARA:	Poder modificar un cliente existente en el sistema
	COMO:	Usuario					
	QUIERO:	Ingresar los valores de nombre, edad y contraseña					
PARA:	Poder modificar un cliente existente en el sistema						
Criterios de aceptación	Los valores de contraseña deben ser un texto de no más de 45 caracteres. Los valores de edad deben ser un número que represente los años. Los valores de nombre deben ser un texto de máximo 250 caracteres. El correo y el id no se deben poder modificar.						

Hisotoria de usuario Nro.4	
Título: Modificación de Mensajes	
Descripción	COMO: Usuario
	QUIERO: Ingresar el valor de texto
	PARA: Poder modificar el contenido en el sistema de un mensaje
Criterios de aceptación Los valores de texto deben ser una cadena de caracteres no superior a 250 caracteres El valor de palco corresponde no se debe poder modificar.	

Hisotoria de usuario Nro.	5		
Título:	Modificación de Reservas		
Descripción	COMO:	Usuario	
	QUIERO:	Ingresar los valores de fecha inicio, fecha entrega y status	
	PARA:	Poder modificar en el sistema una reserva	
Criterios de aceptación	Los valores de cliente y de palco no se deben poder modificar. Los valores de fecha de inicio y fecha entrega deben ser fechas en el formato YYYY-mm-dd la fecha inicio debe ser anterior a fecha fin. El estatus debe ser 'Programado', 'cancelado', 'Realizado'		

Hisotoria de usuario Nro.	6		
Título:	Modificación de Calificación de Reservas		
Descripción	COMO:	Usuario	

	QUIERO:	Ingresar los valores de calificación y mensaje
	PARA:	Poder modificar una reserva ya calificada
Criterios de aceptación	Los valores de cliente calificación debe ser un número entero entre 0 y 5. Los valores de mensaje deben ser un texto no superior a 250 caracteres. El valor de la reserva es un número entero y debe ser tomado de la reserva que se esté calificando. El usuario no ingresa el número de reserva.	

Hisotoria de usuario Nro.	7		
Título:	Modificación de Usuarios Administradores		
Descripción	COMO:	Usuario	
	QUIERO:	Ingresar los valores de nombre y contraseña	
	PARA:	Poder modificar un usuario administrador en el sistema	
Criterios de aceptación	Los valores de contraseña deben ser un texto de no más de 45 caracteres. Los valores de nombre deben ser un texto de máximo 250 caracteres. El correo no se debe poder modificar.		

Hisotoria de usuario Nro.		8	
Título:		Eliminación de palcos	
Descripción	COMO:	Usuario	
	QUIERO:	Seleccionar un palco	
	PARA:	Eliminarlo	
Criterios de aceptación		El palco seleccionado no debe tener mensajes relacionados. El palco seleccionado no debe tener relacionado ninguna reserva. El palco no se vuelve a ver cuando se visualizan los palcos.	

Hisotoria de usuario Nro.	9		
Título:	Eliminación de categorías		
Descripción	COMO:	Usuario	
	QUIERO:	Seleccionar una categoría	
	PARA:	eliminarla	
Criterios de aceptación	Ningún palco debe pertenecer a la categoría La categoría no se vuelve a ver cuando se visualizan las categorías		

Hisotoria de usuario Nro.	10		
Título:	Eliminación de Mensajes		
Descripción	COMO:	Usuario	
	QUIERO:	Seleccionar un mensaje	
	PARA:	Eliminarlo	
Criterios de aceptación	El mensaje no se vuelve a ver cuando se muestran los mensajes.		

Hisotoria de usuario Nro.	11		
Título:	Eliminación de Reservas		
Descripción	COMO:	Usuario	
	QUIERO:	Seleccionar una reserva	
	PARA:	eliminarla	

Criterios de aceptación	La reserva no se vuelve a ver cuando se muestran reservas.
-------------------------	--

Hisotoria de usuario Nro.	12		
Título:	Eliminación de Usuarios administradores		
Descripción	COMO:	Usuario	
	QUIERO:	Seleccionar un usuario administrador	
	PARA:	Eliminarlo	
Criterios de aceptación	El usuario administrador no se vuelve a ver cuando se visualizan usuarios administradores.		

Dificultad

Instrucciones

Instrucciones para la calificación automática

En este caso, no se ejecutará el código directamente sino que se evaluarán los resultados obtenidos al realizar peticiones HTTP al API construida y que actualmente debe estar desplegada en Oracle Cloud. Las peticiones que se realizarán serán de tipo GET, POST, PUT y DELETE Para realizar las pruebas, las tablas de la base datos deben estar vacías y el valor siguiente de id autogenerado debe ser 1.

Entrada	Cada grupo debe proveer URL (o la ip) donde se encuentra desplegada el API. Esta entrada se denominará URL Base (incluir también el puerto si no es 80)
Salida	Corresponde a los mensajes JSON que retorna cada petición. MasterTech hará la comparación de forma automática.

A continuación se presenta un ejemplo del proceso de evaluación. La prueba estará formada por 3 pasos y tendrá los siguientes parámetros.

- Tipo:** describe el tipo de petición HTTP
- Endpoint:** Base URL/enlace de acceso al servicio a evaluar
- Body:** JSON de entrada para la petición HTTP
- Salida:** JSON de salida de la petición HTTP

En el primer paso, MasterTech hará una petición de tipo tipo POST en la cual se agregará un registro a la tabla. Luego se hará una petición GET para validar que el registro se haya guardado de forma exitosa. Luego se realizará una petición de tipo PUT en la que se modificará el registro previamente ingresado. Finalmente, la prueba terminará con una petición GET en la cual se verificará que el registro haya sido modificado de forma exitosa. Todo este proceso se realiza de manera automática por MasterTech. Para la Eliminación de registros, se hará uso del registro previamente creado y modificado. Se consultará con una petición GET, luego se eliminará con una petición DELETE y luego una petición GET validará que se haya eliminado.

Recomendaciones generales

- Antes de enviar la solución del reto, por favor tenga en cuenta los siguientes aspectos:
- Verifique que la aplicación esté desplegada correctamente y sea accesible a través de la URL.
- Verifique que todos los endpoint funcionen. Recuerde que puede probarlos a través de PostMan.
- Las tablas de la base de datos deben estar vacías y recién creadas para que el identificador generado automáticamente en cada tabla esté en 0.
- Debe proveer a MasterTech para las pruebas es la URL de su API incluyendo el puerto en caso de ser diferente a 80, por ejemplo 150.250.45.12:8080.
- Debe copiar el contenido del archivo ReservationService.java en el editor de código para validar las buenas prácticas de programación.
- Debe subir un archivo .zip con la carpeta src de su proyecto y además incluir los archivos correspondientes al frontend en el mismo archivo .zip
- Se debe asegurar que el contenido sea inferior a 5mb. No debe incluir librerías (en caso que las tenga) para disminuir el tamaño del archivo.
- Se debe utilizar una URL distinta por estudiante. Se validará que las urls utilizadas no sean utilizadas por estudiantes diferentes para evitar acciones fraudulentas.
- Todos los integrantes del equipo, harán integración y despliegue para que sus pruebas sean exitosas.
- En el momento de ejecutar la evaluación automática, la aplicación debe estar en ejecución

Casos de prueba			
Paso 1.a	Tipo	POST	
	Endpoint	BASE_URL/api/Category/save	

	<div><div>Body</div><div>Salida</div></div> <div>{ "name": "categoria1", "description": "primera categoria" }</div>	
Paso 1.b	<div><div>Tipo</div><div>Endpoint</div><div>Body</div><div>Salida</div></div> <div>GET BASE_URL/api/Category/all [{"id":1,"name":"categoria1","description":"primera categoria","boxes":[]}]</div>	
Paso 2.a	<div><div>Tipo</div><div>Endpoint</div><div>Body</div><div>Salida</div></div> <div>PUT BASE_URL/api/Category/update { "id":1, "name": "modificada", "description": "se ha modificado" }</div>	
Paso 2.b	<div><div>Tipo</div><div>Endpoint</div><div>Body</div><div>Salida</div></div> <div>GET BASE_URL/api/Category/all { "id":1, "name": "modificada", "description": "se ha modificado", "boxes":[] }</div>	
Paso 3.a	<div><div>Tipo</div><div>Endpoint</div><div>Body</div><div>Salida</div></div> <div>POST BASE_URL/api/Box/save { "location": "VIP", "capacity": 15, "category": { "id": 1, "name": "Palco vip para 15 personas", "description": "Palco vip para 15 personas" } }</div>	
Paso 3.b	<div><div>Tipo</div><div>Endpoint</div><div>Body</div><div>Salida</div></div> <div>GET BASE_URL/api/Box/all [{"id":1,"name":"Palco vip para 15 personas","location":"VIP","capacity":15,"description":"Palco vip para 15 personas","category":{"id":1,"name":"modificada","description":"se ha modificado"},"messages": [], "reservations": []}]</div>	
Paso 4.a	<div><div>Tipo</div><div>Endpoint</div><div>Body</div><div>Salida</div></div> <div>PUT BASE_URL/api/Box/update { "id":1, "location": "NOTVIP", "name": "Palco Not vip para 18 personas", "description": "Palco Not vip para 18 personas", "capacity": 18 }</div>	
Paso 4.b	<div><div>Tipo</div><div>Endpoint</div><div>Body</div><div>Salida</div></div> <div>GET BASE_URL/api/Box/all [{"id":1,"name":"Palco Not vip para 18 personas","location":"NOTVIP","capacity":18,"description":"Palco Not vip para 18 personas","category":{"id":1,"name":"modificada","description":"se ha modificado"},"messages": [], "reservations": []}]</div>	
Paso 5.a	<div><div>Tipo</div><div>Endpoint</div><div>Body</div><div>Salida</div></div> <div>POST BASE_URL/api/Client/save { "name": "Agustin Parra", "email": "agustin@gmail.com", "password": "agustin123", "age": 18 }</div>	
Paso 5.b	<div><div>Tipo</div><div>Endpoint</div><div>Body</div><div>Salida</div></div> <div>GET BASE_URL/api/Client/all [{"idClient":1,"email":"agustin@gmail.com","password":"agustin123","name":"Agustin Parra","age":18,"messages": [], "reservations": []}]</div>	
Paso 6.a	<div><div>Tipo</div><div>Endpoint</div></div> <div>PUT BASE_URL/api/Client/update</div>	

	<table><tr><td>Body</td><td>{ "idClient":1, "name": "Adeodato Sanchez", "email": "agustin@gmail.com", "password": "adeodato123", "age": 15 }</td></tr><tr><td>Salida</td><td></td></tr></table>	Body	{ "idClient":1, "name": "Adeodato Sanchez", "email": "agustin@gmail.com", "password": "adeodato123", "age": 15 }	Salida					
Body	{ "idClient":1, "name": "Adeodato Sanchez", "email": "agustin@gmail.com", "password": "adeodato123", "age": 15 }								
Salida									
Paso 6.b	<table><tr><td>Tipo</td><td>GET</td></tr><tr><td>Endpoint</td><td>BASE_URL/api/Client/all</td></tr><tr><td>Body</td><td></td></tr><tr><td>Salida</td><td>[{"idClient":1, "email": "agustin@gmail.com", "password": "adeodato123", "name": "Adeodato Sanchez", "age": 15, "messages": [], "reservations": []}]</td></tr></table>	Tipo	GET	Endpoint	BASE_URL/api/Client/all	Body		Salida	[{"idClient":1, "email": "agustin@gmail.com", "password": "adeodato123", "name": "Adeodato Sanchez", "age": 15, "messages": [], "reservations": []}]
Tipo	GET								
Endpoint	BASE_URL/api/Client/all								
Body									
Salida	[{"idClient":1, "email": "agustin@gmail.com", "password": "adeodato123", "name": "Adeodato Sanchez", "age": 15, "messages": [], "reservations": []}]								
Paso 7.a	<table><tr><td>Tipo</td><td>DELETE</td></tr><tr><td>Endpoint</td><td>BASE_URL/api/Client/1</td></tr><tr><td>Body</td><td></td></tr><tr><td>Salida</td><td></td></tr></table>	Tipo	DELETE	Endpoint	BASE_URL/api/Client/1	Body		Salida	
Tipo	DELETE								
Endpoint	BASE_URL/api/Client/1								
Body									
Salida									
Paso 7.b	<table><tr><td>Tipo</td><td>GET</td></tr><tr><td>Endpoint</td><td>BASE_URL/api/Client/all</td></tr><tr><td>Body</td><td></td></tr><tr><td>Salida</td><td>[]</td></tr></table>	Tipo	GET	Endpoint	BASE_URL/api/Client/all	Body		Salida	[]
Tipo	GET								
Endpoint	BASE_URL/api/Client/all								
Body									
Salida	[]								
Paso 8.a	<table><tr><td>Tipo</td><td>DELETE</td></tr><tr><td>Endpoint</td><td>BASE_URL/api/Box/1</td></tr><tr><td>Body</td><td></td></tr><tr><td>Salida</td><td></td></tr></table>	Tipo	DELETE	Endpoint	BASE_URL/api/Box/1	Body		Salida	
Tipo	DELETE								
Endpoint	BASE_URL/api/Box/1								
Body									
Salida									
Paso 8.b	<table><tr><td>Tipo</td><td>GET</td></tr><tr><td>Endpoint</td><td>BASE_URL/api/Box/all</td></tr><tr><td>Body</td><td></td></tr><tr><td>Salida</td><td>[]</td></tr></table>	Tipo	GET	Endpoint	BASE_URL/api/Box/all	Body		Salida	[]
Tipo	GET								
Endpoint	BASE_URL/api/Box/all								
Body									
Salida	[]								
Paso 9.a	<table><tr><td>Tipo</td><td>DELETE</td></tr><tr><td>Endpoint</td><td>BASE_URL/api/Category/1</td></tr><tr><td>Body</td><td></td></tr><tr><td>Salida</td><td></td></tr></table>	Tipo	DELETE	Endpoint	BASE_URL/api/Category/1	Body		Salida	
Tipo	DELETE								
Endpoint	BASE_URL/api/Category/1								
Body									
Salida									
Paso 9.b	<table><tr><td>Tipo</td><td>GET</td></tr><tr><td>Endpoint</td><td>BASE_URL/api/Category/all</td></tr><tr><td>Body</td><td></td></tr><tr><td>Salida</td><td>[]</td></tr></table>	Tipo	GET	Endpoint	BASE_URL/api/Category/all	Body		Salida	[]
Tipo	GET								
Endpoint	BASE_URL/api/Category/all								
Body									
Salida	[]								