

NODE.JS

Luis Miguel Avendaño Lozano
ACADEMLO

TAREA 3

Temas:

- Relaciones
- Error Handling
- Autorización

“Hola de nuevo, vimos que has hecho un buen trabajo, ahora hay que establecer una relación entre estos 2 modelos, recuerda que un usuario puede tener muchos servicios (repairs) y un servicio le pertenece a un usuario.

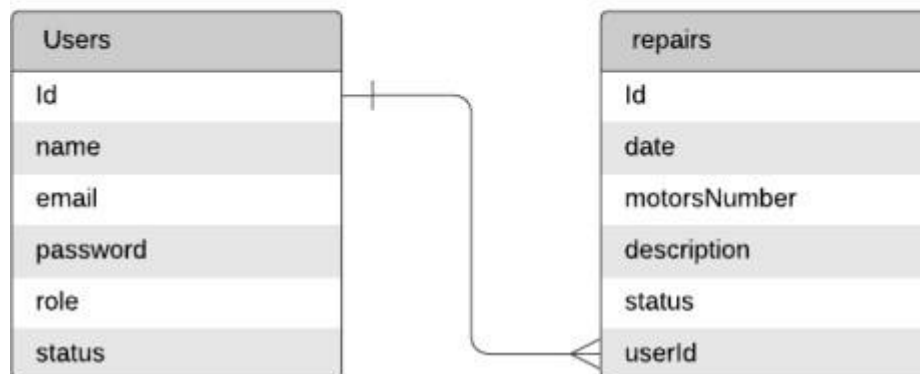
También nuestros empleados nos pidieron lo siguiente, que al buscar los servicios pendientes y los completados, podemos adjuntar la información del usuario que solicito este servicio.

Adicionalmente escuche que hay una técnica de manejo de errores, seria importante que lo implementases.

Por último, se nos olvidó la entrega pasada proteger ciertas rutas para que solo el dueño de la cuenta pueda realizar esas acciones mas adelante te dejo cuáles son esas rutas.

Muy buen trabajo, saludos.”

ACADEMLO MOTORS
Luis Miguel Avendaño Lozano | December 10, 2022



Estos endpoints se deben proteger para que el dueño de la cuenta que se intenta actualizar o deshabilitar pueda realizar dichas acciones:

<i>/api/v1/users</i>		
<i>HTTP VERB</i>	<i>URL</i>	<i>Descripción</i>
PATCH	/:id	Actualizar los datos de un usuario dado un id, solo puede actualizar su <i>name</i> y <i>email</i>
DELETE	/:id	Deshabilitar la cuenta de un usuario

Utilizar lo visto para aplicar *error handling* en nuestro proyecto (`catchAsync`, `globalErrorHandler` y `AppError`) y optimizar nuestro código.