

**Instituto Tecnológico y de Estudios Superiores de Monterrey
Campus Querétaro**



**Tecnológico
de Monterrey**

Construcción de software

Laboratorio 25 - Transacciones

Daniel Contreras Chávez

11 abr 2024

Link del commit:

<https://github.com/DanielContrerasCh/Team-404/commit/ddb8eeadaf9aa1ea491bafb7364a1fc97aaff84>

Cree la transacción eliminaMarca que se encarga de que, para no tener problemas con las constraints de las llaves, realiza los delete en orden:

BEGIN

```
DELETE FROM opciones_pregunta WHERE IDPreguntas IN (SELECT IDPreguntas
FROM preguntas WHERE NombreMarca = uNombreMarca COLLATE utf8mb4_unicode_ci);
```

```
DELETE FROM preguntas WHERE NombreMarca = uNombreMarca COLLATE
utf8mb4_unicode_ci;
```

```
DELETE FROM categorias WHERE nombre_marca = uNombreMarca COLLATE
utf8mb4_unicode_ci;
```

```
DELETE FROM imagenmarca WHERE nombre = uNombreMarca COLLATE
utf8mb4_unicode_ci;
```

END

Explicación de los commits:

```

@@ -218,9 +218,11 @@ module.exports = class Marca {
218
219     await connection.beginTransaction(); // Inicia una transacción
220
221 -     await connection.execute('call eliminarMarca(?)',
222         [nombreMarca]);
222     await Marca.constructImagePath(nombreMarca);
223
224     await connection.commit(); // Si todo va bien, confirma los
        cambios
225
226     } catch (error) {
227         await connection.rollback(); // Si hay un error, revierte
        los cambios
228         console.log(error);
229
230     }
231 }
232
233 +     return 'Operación realizada con éxito';
234 + } catch (error) {
235     await connection.rollback(); // Si hay un error, revierte
        los cambios
236     console.log(error);
237 }

```

Primero me encargué de cambiar el orden del llamado de las funciones, de manera que, se elimina primero la imagen de la marca, ya que si primero elimino la marca, entonces la imagen asociada no la va a encontrar.

```

176 +
177 +     static constructImagePath(nombre) {
178 +         console.log("Nombre de la marca recibido:", nombre);
179 +         db.execute(`SELECT imagen FROM imagenmarca WHERE nombre = ?`,
180 + [nombre])
181 +         .then((rows, fields) => {
182 +             console.log("Resultado de la consulta SQL:", rows);
183 +             if (rows.length === 0) {
184 +                 console.log('La marca no tiene una imagen asociada.');
```

```

185 +                 return; // No hay necesidad de continuar si no hay
186 +                 imagen asociada
187 +             }
188 +             const rutaImagen = rows[0].imagen;
189 +             console.log(rutaImagen);
190 +
191 +             // Construir la ruta completa del archivo
192 +             const rutaCompleta = path.join('public/img/', rutaImagen);
193 +             console.log(rutaCompleta)
194 +             // Verificar si el archivo existe antes de intentar
195 +             eliminarlo
196 +             if (fs.existsSync(rutaCompleta)) {
197 +                 // Intentar eliminar el archivo
198 +                 try {
199 +                     // Eliminar el archivo
200 +                     fs.unlinkSync(rutaCompleta);
201 +                     console.log('El archivo fue eliminado
correctamente.');
```

```

61
62 -         // Luego de eliminar el archivo, eliminar la fila de
63 -         la base de datos
64 -         return db.execute(`DELETE FROM imagenmarca WHERE
65 -         nombre = ?`, [marca]);
66 -         });
67 -         .catch(error => {
68 -             console.error(error);
69 -             throw new Error('Error al eliminar la marca o el
archivo');
70 -         });

```

Creé la función constructImagePath, que se encarga de realizar una consulta para obtener la imagen asociada a la marca, y luego crea el path del archivo de la imagen que se debería eliminar, finalmente elimina la imagen asociada.