

## Funcionamiento del botón "Restablecer estilos"

Este documento explica los cambios realizados en los archivos del visor 3D para que el botón 'Restablecer estilos' devuelva el modelo a su aspecto original.

### ¿Qué hace esta funcionalidad?

Permite que el usuario aplique cambios de color, roughness y metalness a un modelo 3D (.glb), pero también pueda restaurar el material original (textura o color inicial del modelo) con un botón.

## 1. Cambios en scene.js

### 1.1 → Guardar el material original del modelo

Cuando se carga un modelo, lo recorremos (traverse) y guardamos el **material original** en una propiedad especial de cada malla.

```
child.userData.originalMaterial = child.material.clone();
```

Esto lo añadimos **dentro de la función loadModel(file)**, justo antes de añadir el modelo a la escena:

```
currentModel = gltf.scene;

// Guarda el material original por si luego se quiere restaurar
currentModel.traverse((child) => {
  if (child.isMesh && child.material) {
    child.userData.originalMaterial = child.material.clone();
  }
});

scene.add(currentModel);
```

## 1.2 → Crear la función restaurarMaterialesOriginales()

Creamos una función que recorra el modelo y devuelva los materiales originales guardados en `userData.originalMaterial`.

```
export function restaurarMaterialesOriginales() {  
  if (!currentModel) {  
    alert("No hay modelo cargado actualmente.");  
    return;  
  }  
  
  currentModel.traverse((child) => {  
    if (child.isMesh && child.userData.originalMaterial) {  
      child.material = child.userData.originalMaterial.clone();  
      child.material.needsUpdate = true;  
    }  
  });  
}
```

Esta función se exporta para poder usarla desde fuera.

## 2. Cambios en viewerStyle.js

### 2.1 → Importar la nueva función

Al principio del archivo, además de `actualizarModelo`, ahora importamos también `restaurarMaterialesOriginales`:

```
import { actualizarModelo, restaurarMaterialesOriginales } from './js/scene.js';
```

### 2.2 → Código del botón “Restablecer”

Se modificó el comportamiento del botón creado con ID `resetEstilos` para que al hacer clic:

- Se llame a `restaurarMaterialesOriginales()`.
- Se borren los estilos aplicados (`sessionStorage`).
- Se reinicie el formulario visualmente (color blanco, sliders a 500).

```

btnReset.addEventListener('click', () => {
  restaurarMaterialesOriginales();          // Restaura los materiales origina
  sessionStorage.removeItem('estilos');     // Borra los cambios guardados

  // (Opcional) Reinicia el formulario visualmente
  const colorInput = document.getElementById('chooseColor');
  const roughnessInput = form.elements['roughness'];
  const metalnessInput = form.elements['metalness'];

  if (colorInput) colorInput.value = "#ffffff"; // o el color por defecto
  if (roughnessInput) roughnessInput.value = 500;
  if (metalnessInput) metalnessInput.value = 500;
});

```

## Resumen Final

Al cargar un modelo, se guarda su material original. Si el usuario modifica los estilos, puede luego recuperar el aspecto original con el botón de 'Restablecer', que accede a esos materiales guardados.

## ¿Y el sessionStorage?

Se sigue usando sessionStorage como antes para guardar los estilos cuando el usuario mueve sliders o cambia el color. Pero **al restablecer**, limpiamos la entrada llamada 'estilos' con:

```

sessionStorage.removeItem('estilos');

```