

Laboratorio 24 - AJAX

En este laboratorio, las funcionalidades se implementaron para el avance 4 con una funcionalidad para eliminar usuarios: Este código utiliza la API Fetch, una forma moderna de realizar operaciones AJAX (Asynchronous JavaScript and XML), aunque en este caso se usa para hacer solicitudes HTTP asíncronas utilizando JSON como formato de intercambio de datos, no XML. La API Fetch permite hacer solicitudes HTTP desde JavaScript para realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) sin necesidad de recargar la página, lo cual es una característica central de AJAX.

1. Importancia de AJAX en RIAs

AJAX ha revolucionado el desarrollo web permitiendo que las páginas web se actualicen de manera asíncrona, lo que mejora significativamente la interactividad y la experiencia del usuario. Las principales ventajas incluyen:

- Interactividad Mejorada: Actualizaciones parciales de la página sin recargar toda la página web.
- Eficiencia en el Uso del Ancho de Banda y la Carga del Servidor: Reducción del volumen de datos transferidos entre el cliente y el servidor.
- Facilitación del Desarrollo de Aplicaciones Web Dinámicas: Permite la creación de aplicaciones que requieren interacción en tiempo real.

2. Implicaciones de Seguridad de AJAX

- El uso de AJAX introduce varias preocupaciones de seguridad que deben ser cuidadosamente manejadas, incluyendo:
- Cross-Site Scripting (XSS): Riesgo de inyección de código malicioso en la página web.
- Cross-Site Request Forgery (CSRF): Riesgo de engañar a un usuario para que realice acciones no deseadas en una aplicación web.
- Exposición de Datos Sensibles: Riesgo de exponer datos sensibles a través de comunicaciones asíncronas no seguras.

3. Validaciones de Seguridad: Cliente vs. Servidor

Las validaciones de seguridad deben realizarse tanto en el cliente como en el servidor para garantizar una aplicación robusta y segura. Mientras que las validaciones del lado del cliente mejoran la experiencia del usuario con retroalimentación inmediata, las del lado del servidor son cruciales para la seguridad, ya que el lado del cliente es susceptible de ser manipulado.

4. JSON: El Formato de Intercambio de Datos Preferido

JSON (JavaScript Object Notation) es un formato de intercambio de datos eficiente y el más utilizado en el contexto de AJAX. Ofrece una sintaxis sencilla para la serialización y transmisión de estructuras de datos complejas, facilitando la integración y manipulación de datos en aplicaciones web.