

Universidad de Guadalajara

CENTRO UNIVERSITARIO DE CIENCIAS EXACTAS E INGENIERÍAS

Ingeniería Informática.



Actividad 3.2 :Manipulación de DOM y eventos.

Desarrollo de Front End.

Yosef Sánchez Gutiérrez

Marco Alejandro González Mireles

Mirella Stephania Palomera Gómez

Roberto Carlos Martínez Aviña

04/07/25

Mtra. Rosalia Iñiguez .

Introducción:

En esta actividad desarrollamos una tabla interactiva que toma su información desde un archivo XML. La idea principal fue mostrar todos los registros de forma clara y funcional, permitiendo al usuario hacer búsquedas, ordenar los datos por columna (ya sea ascendente o descendente), elegir cuántos registros ver por página y navegar entre ellos con paginación.

El objetivo fue lograr que, sin importar cuántos registros tuviera el XML, se pudieran visualizar y manipular fácilmente desde una página web, brindando una experiencia más cómoda y dinámica al usuario.

A lo largo del proceso aprendí a cargar datos XML con jQuery, recorrerlos para armar una tabla HTML y agregarle funciones útiles como el filtrado en tiempo real, botones de navegación entre páginas, y control sobre la cantidad de registros visibles. Todo esto se logró combinando JavaScript, HTML, CSS y algunas herramientas de jQuery UI para mejorar la presentación.

Cargando el archivo XML

Utilicé \$.ajax() de jQuery para cargar el archivo autos.xml, que contiene una lista de automóviles con información como marca, modelo, año, color, precio, tipo, transmisión y combustible.

```
function loadXMLData() {  
  $.ajax({  
    type: "GET",  
    url: "autos.xml",  
    dataType: "xml",  
    success: (xml) => {  
      parseXMLData(xml) any  
      $("#loading-message").hide()  
      displayTable()  
    },  
    error: () => {  
      $("#loading-message").hide()  
      $("#error-message").show()  
      console.error("Error al cargar el archivo XML")  
    },  
  })  
}
```

Archivo XML:

```
autos.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <autos>
3   <auto>
4     <marca>Toyota</marca>
5     <modelo>Corolla</modelo>
6     <año>2023</año>
7     <color>Blanco</color>
8     <precio>385000</precio>
9     <tipo>Sedán</tipo>
10    <transmision>Automática</transmision>
11    <combustible>Gasolina</combustible>
12  </auto>
13  <auto>
14    <marca>Honda</marca>
15    <modelo>Civic</modelo>
16    <año>2022</año>
17    <color>Negro</color>
18    <precio>420000</precio>
19    <tipo>Sedán</tipo>
20    <transmision>Manual</transmision>
21    <combustible>Gasolina</combustible>
22  </auto>
```

Agregando búsqueda en tiempo real

Se añadió un campo de búsqueda para filtrar los resultados mientras el usuario escribe. La búsqueda funciona sobre cualquier columna (marca, modelo, color, etc.):

```
// Búsqueda en tiempo real
$("#search-input").on("input", function () {
  const searchTerm = $(this).val().toLowerCase()
  filterData(searchTerm)
})
```

Tabla Dinámica de Autos

Buscar:

Registros por página:

Total de registros: 2

Mostrando: 1 - 2

Marca	Modelo	Año	Color	Precio	Tipo	Transmisión	Combustible
Toyota	Corolla	2023	Blanco	\$385,000	Sedán	Automática	Gasolina
Toyota	RAV4	2024	Blanco	\$545,000	SUV	Automática	Híbrido

Ordenamiento por columnas

Cada columna de la tabla es ordenable. Al hacer clic sobre un encabezado, se alterna entre orden ascendente y descendente:

```
// Ordenamiento por columnas
$(".sortable").on("click", function () {
  const column = $(this).data("column")
  handleSort(column)
})
```

Ascendente:

Marca	Modelo	Año	Color
Acura	TLX	2023	Negro

Descendente:

Marca	Modelo	Año	Color
Volvo	XC60	2024	Azul

Paginación y control de registros

Se agregó un control para elegir cuántos registros mostrar por página (5, 10, 15, etc.) y botones para moverse entre páginas:

Buscar:

Registros por página:

5

51020

Total de registros: 50Mostrando: 1 - 5

Marca	Modelo	Año	Color	Precio	Transmisión	Combustible
Volvo	XC60	2024	Azul	\$785,000	SUV	Automática
Volkswagen	Jetta	2022	Plata	\$395,000	Sedán	Gasolina
Volkswagen	Tiguan	2023	Blanco	\$525,000	SUV	Gasolina
Toyota	Corolla	2023	Blanco	\$385,000	Sedán	Gasolina
Toyota	RAV4	2024	Blanco	\$545,000	SUV	Híbrido

« Anterior

123...10

Siguiente »

Página 1 de 10

¿Cómo funciona la tabla?

Carga los datos desde el XML con \$.ajax.

Los convierte en objetos JavaScript y los guarda en un arreglo.

Ese arreglo se usa para mostrar la tabla.

Al buscar, ordenar o cambiar de página, se actualiza la tabla sin necesidad de recargar la página.

Tabla Dinámica de Autos

Buscar:

Registros por página:

10

Total de registros: 50

Mostrando: 1 - 10

Marca	Modelo	Año	Color	Precio	Tipo	Transmisión	Combustible
Volvo	XC60	2024	Azul	\$785,000	SUV	Automática	Híbrido
Volkswagen	Jetta	2022	Plata	\$395,000	Sedán	Automática	Gasolina
Volkswagen	Tiguan	2023	Blanco	\$525,000	SUV	Automática	Gasolina
Toyota	Corolla	2023	Blanco	\$385,000	Sedán	Automática	Gasolina
Toyota	RAV4	2024	Blanco	\$545,000	SUV	Automática	Híbrido
Tesla	Model 3	2023	Blanco	\$850,000	Sedán	Automática	Eléctrico
Tesla	Model Y	2024	Rojo	\$925,000	SUV	Automática	Eléctrico
Suzuki	Swift	2023	Azul	\$265,000	Hatchback	Manual	Gasolina
Subaru	Outback	2023	Verde	\$525,000	SUV	Automática	Gasolina
Smart	ForTwo	2022	Blanco	\$325,000	Hatchback	Automática	Eléctrico

« Anterior

1

2

3

...

5

Siguiente »

Página 1 de 5

Herramientas utilizadas:

HTML: para la estructura de la página.

CSS y jQuery UI: para mejorar el estilo y presentación de los controles.

JavaScript y jQuery: para cargar el archivo XML, organizar los datos, y agregar funciones como búsqueda, ordenamiento y paginación.

XML: como fuente de datos estática (archivo autos.xml).

Conclusión:

Esta actividad nos ayudó a entender cómo conectar datos estáticos (como un archivo XML) con una interfaz web interactiva. Aprendí a:

- Cargar y procesar archivos XML con JavaScript.
- Generar tablas HTML dinámicamente.
- Aplicar filtros, ordenamientos y paginación de forma manual.
- Mejorar la experiencia del usuario con controles intuitivos.

Aunque parezca una tarea sencilla, este tipo de implementación se usa bastante en aplicaciones reales donde se necesitan tablas con muchos datos y navegación fluida. También nos sirvió para repasar jQuery, trabajar con datos externos y estructurar mejor el código.