



ALUMNO:

María Fernanda Fuentes Abascal

DOCENTE:

José Miguel Carrera Pacheco

ACTIVIDAD:

DOM y componentes dinámicos accesibles

MATERIA:

Desarrollo Web Profesional

FECHA:

08/02/2026

## 1. INTRODUCCIÓN

Las aplicaciones web modernas se caracterizan por interfaces altamente dinámicas que responden en tiempo real a las acciones del usuario. Estos cambios suelen lograrse mediante la manipulación del DOM utilizando JavaScript o frameworks modernos. Sin embargo, cuando estas manipulaciones no se realizan correctamente, pueden generar problemas graves de accesibilidad, usabilidad y experiencia de usuario.

Esta investigación analiza los fundamentos teóricos y técnicos del DOM, el Virtual DOM, la gestión del foco y el uso correcto de ARIA, con el objetivo de comprender cómo crear componentes dinámicos sin afectar la accesibilidad ni confundir al usuario.

## 2. DOM (DOCUMENT OBJECT MODEL)

El DOM es una interfaz de programación que representa un documento HTML como una estructura jerárquica de nodos. Cada elemento, atributo y texto del documento se convierte en un nodo que puede ser manipulado mediante JavaScript.

### 2.1 Funcionamiento del DOM

Cuando el navegador carga una página web:

1. Interpreta el HTML
2. Construye el árbol del DOM
3. Aplica CSS (CSSOM)
4. Renderiza la interfaz visual

Cada vez que el DOM es modificado:

- El navegador puede recalcular estilos
- Reorganizar el layout (reflow)
- Volver a pintar la interfaz (repaint)

Estos procesos impactan directamente en:

- Rendimiento
- Consumo de recursos
- Experiencia del usuario
- Accesibilidad

## 3. VIRTUAL DOM

El Virtual DOM es una representación en memoria del DOM real, utilizada por frameworks como React. En lugar de modificar directamente el DOM real ante cada cambio, se genera una copia virtual que se compara con el estado anterior.

### 3.1 Diferencias clave entre DOM y Virtual DOM

DOM	Virtual DOM
Cambios directos al navegador	Cambios primero en memoria
Operaciones costosas	Optimización de actualizaciones
Riesgo de pérdida de foco	Mayor control del render
Difícil de testear	Más predecible

### 3.2 Accesibilidad y Virtual DOM

Aunque el Virtual DOM mejora el rendimiento, no garantiza accesibilidad. Si los componentes se re-renderizan sin control:

- El foco puede perderse
- Lectores de pantalla pueden volver a leer contenido
- Se puede romper la navegación por teclado

Por ello, el desarrollador debe implementar prácticas adicionales para mantener la accesibilidad.

## 4. PROBLEMAS COMUNES AL MANIPULAR EL DOM

La manipulación incorrecta del DOM puede generar múltiples problemas de experiencia y accesibilidad.

### 4.1 Problemas de foco

- El foco desaparece al actualizar contenido
- El foco regresa al inicio de la página
- El usuario pierde referencia de su posición

### 4.2 Problemas con lectores de pantalla

- Cambios no anunciados
- Información duplicada
- Mensajes invisibles para tecnologías asistivas

### 4.3 Problemas semánticos

- Uso excesivo de `<div>` y `<span>`
- Falta de encabezados jerárquicos
- Elementos interactivos sin rol correcto

## 5. GESTIÓN DEL FOCO

El foco representa el elemento activo con el que el usuario interactúa, especialmente relevante para usuarios que navegan con teclado o lectores de pantalla.

### 5.1 Importancia del foco

Una mala gestión del foco puede:

- Confundir al usuario
- Romper la navegación
- Hacer la interfaz inutilizable

### 5.2 Principios básicos de gestión del foco

- El foco debe ser visible
- El foco debe moverse de forma lógica
- No debe perderse en cambios dinámicos
- Debe colocarse en el contenido más relevante

## 6. USO CORRECTO DE ARIA

ARIA (Accessible Rich Internet Applications) es un conjunto de atributos diseñados para mejorar la accesibilidad de aplicaciones dinámicas.

### 6.1 aria-live

Permite anunciar cambios dinámicos a lectores de pantalla sin mover el foco.

Tipos:

- `aria-live="polite"`: anuncios no urgentes
- `aria-live="assertive"`: anuncios urgentes

### 6.2 Roles ARIA

- `role="alert"`
- `role="dialog"`
- `role="navigation"`
- `role="status"`

### 6.3 Riesgos del mal uso de ARIA

- Información redundante

- Confusión en lectores de pantalla
- Accesibilidad peor que sin ARIA

## **7. ACCESIBILIDAD EN COMPONENTES DINÁMICOS**

Los componentes dinámicos deben:

- Respetar HTML semántico
- Mantener foco estable
- Anunciar cambios relevantes
- Evitar cambios inesperados

La accesibilidad no debe ser un añadido posterior, sino parte del diseño inicial.

## **8. RELACIÓN CON ESTÁNDARES WCAG 2.1**

Esta investigación se alinea con los principios:

- Perceptible: Cambios anunciados
- Operable: Navegación por teclado
- Comprensible: Comportamiento predecible
- Robusto: Compatible con tecnologías asistivas

## **9. CONCLUSIÓN DE LA INVESTIGACIÓN**

La manipulación del DOM y la creación de componentes dinámicos representan uno de los mayores retos de accesibilidad en aplicaciones web modernas. Comprender las diferencias entre DOM y Virtual DOM, así como los problemas comunes, la gestión del foco y el uso adecuado de ARIA, es fundamental para desarrollar interfaces dinámicas profesionales, accesibles y centradas en el usuario.

Esta investigación establece la base teórica necesaria para justificar las decisiones técnicas que se tomarán en las siguientes fases del proyecto.