



# Traffic Light

(Project Final)

## Presentación sobre DOM y WebAssembly: Manejo con JavaScript

Cómo JavaScript interactúa con el DOM y WebAssembly para la creación de páginas web dinámicas y eficientes y porque WebAssembly no puede interactuar directamente con el DOM.

### ¿Qué es el DOM?

El DOM, o Modelo de Objetos del Documento (Document Object Model), es una interfaz de programación para documentos HTML, XHTML y XML que proporciona una representación jerárquica de la estructura de una página web. Permite a los programas, como JavaScript, acceder y manipular los elementos de la página, así como responder a eventos del usuario de forma dinámica.

Si usas etiquetas como: `<html>`, `<head>`, `<body>`, `<div>`, etc. es representado como un nodo en el árbol

- **Acceso a elementos**
- **Manipulación del contenido, estilo, estructura**
- **Manejo de eventos**

## **¿Porque WebAssembly no tiene acceso directo al DOM en los navegadores WEB?**

WebAssembly (Wasm) es un formato binario de bajo nivel diseñado para ejecutar código de manera eficiente en navegadores web, es muy poderoso en términos de rendimiento y capacidad de ejecución.

Pero WebAssembly opera en un nivel inferior al del DOM y no tiene acceso directo a las API del navegador, incluido el DOM.

**Seguridad:** Permitir a Wasm interactuar directamente con el DOM podría abrir la puerta a posibles vulnerabilidades y ataques.

**Estandarización:** Proporcionar acceso directo al DOM desde WebAssembly requeriría una interfaz estandarizada y uniforme que funcione en todos los navegadores, lo cual es complejo de implementar y mantener.

## **¿Hay alguna solución?**

Para interactuar con el DOM, WebAssembly debe comunicarse con JavaScript, que a su vez tiene acceso al DOM y puede realizar operaciones en él en nombre de WebAssembly.

## **¿Hay alguna solución?**

JavaScript facilita la interacción entre WebAssembly y el DOM a través de una serie de mecanismos que permiten la comunicación entre los dos entornos:

- Carga de módulos WebAssembly**

JavaScript puede cargar módulos WebAssembly utilizando la API

- Transferencia de datos entre WebAssembly y JavaScript**

JavaScript y WebAssembly pueden intercambiar datos a través de parámetros de función y valores de retorno. JavaScript puede pasar datos a funciones WebAssembly como argumentos, y WebAssembly puede devolver datos a JavaScript como valores de retorno de función.

## **¿Qué hacemos en el proyecto?**

En el proyecto JavaScript carga los módulos WebAssembly que son los datos que requerimos para realizar los procedimientos donde toda la logica se hace en WebAssembly.