

# **Fundamentos de Tecnología Web**

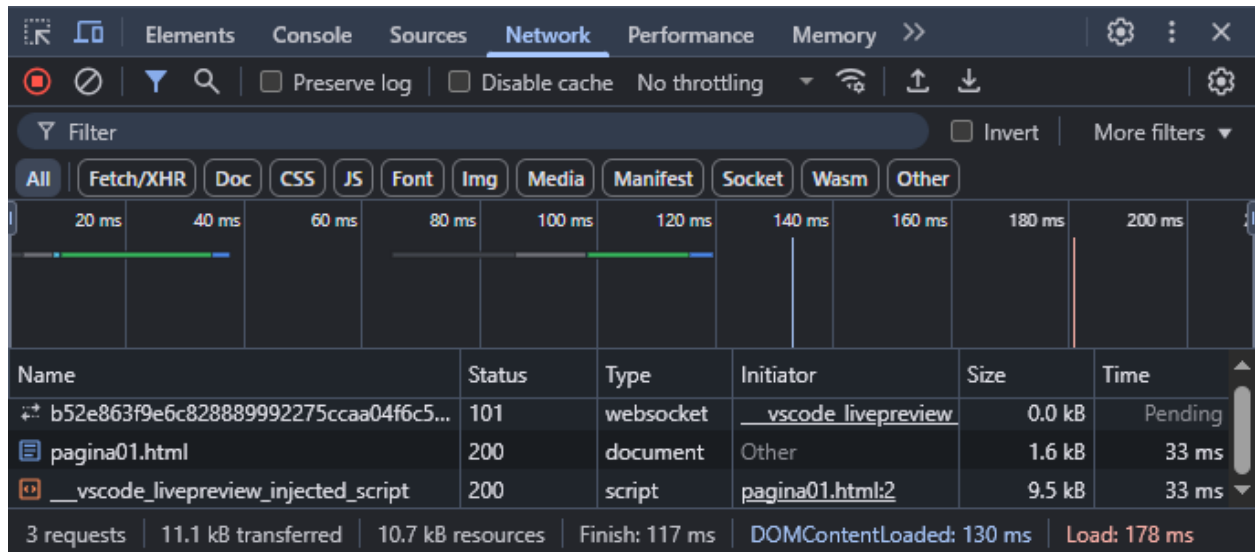
## **Refactorización**

**Pale Arenas Iván**

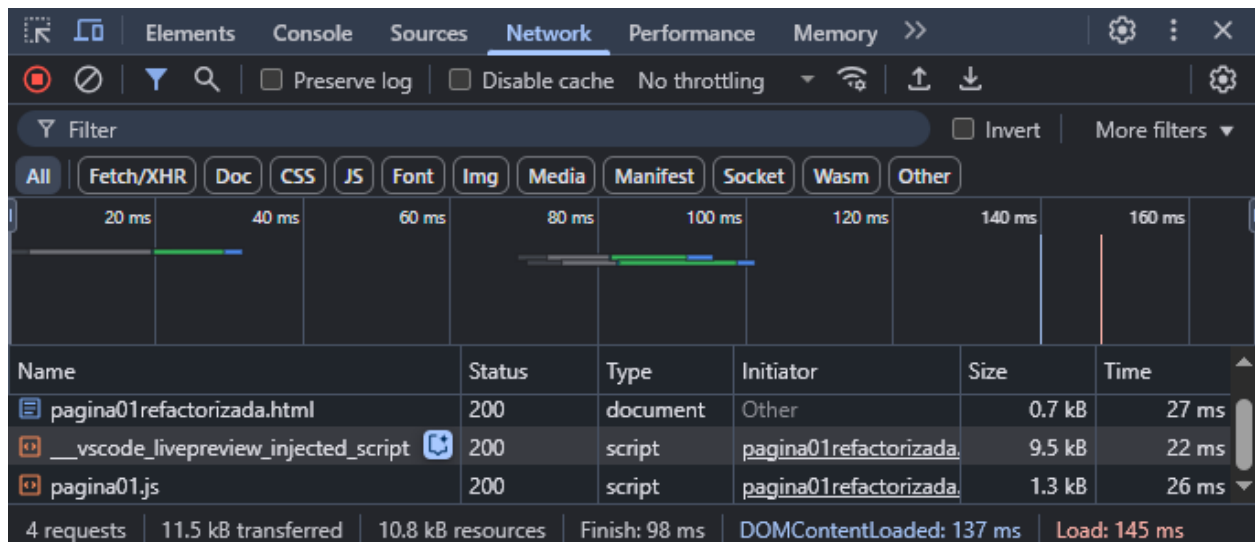


## Página 01

### Sin Factorizar

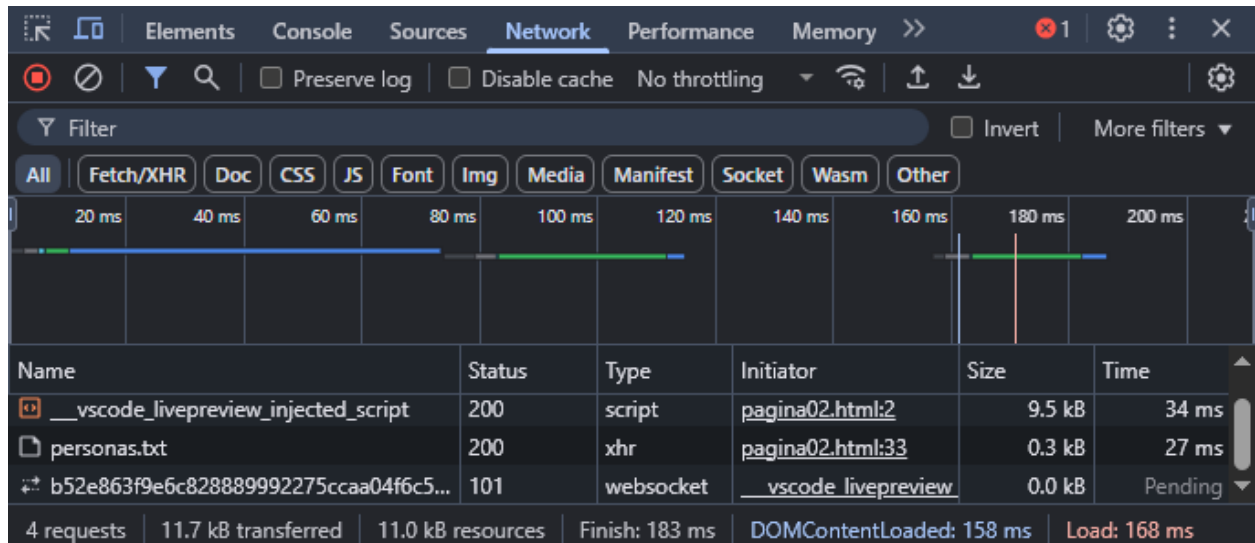


### Refactorizado

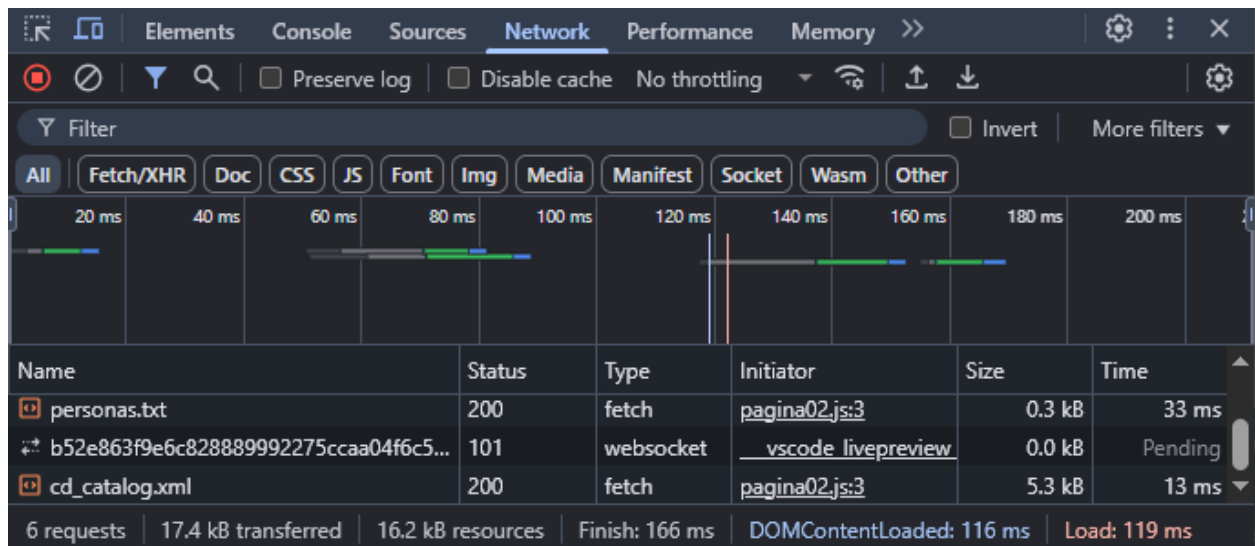


## Página 02

### Sin Factorizar

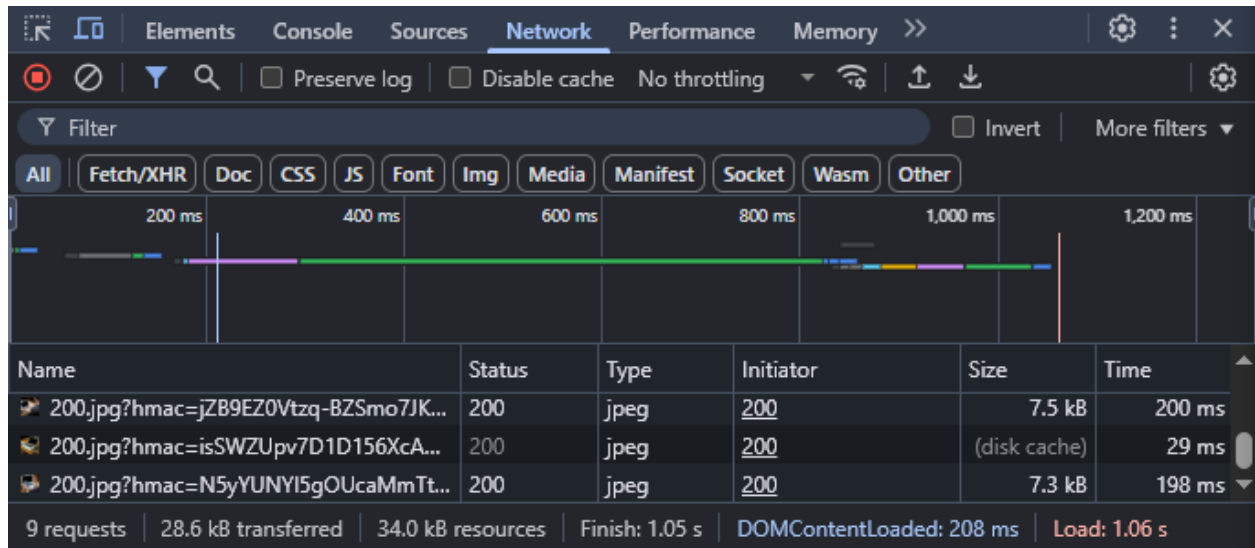


### Factorizada

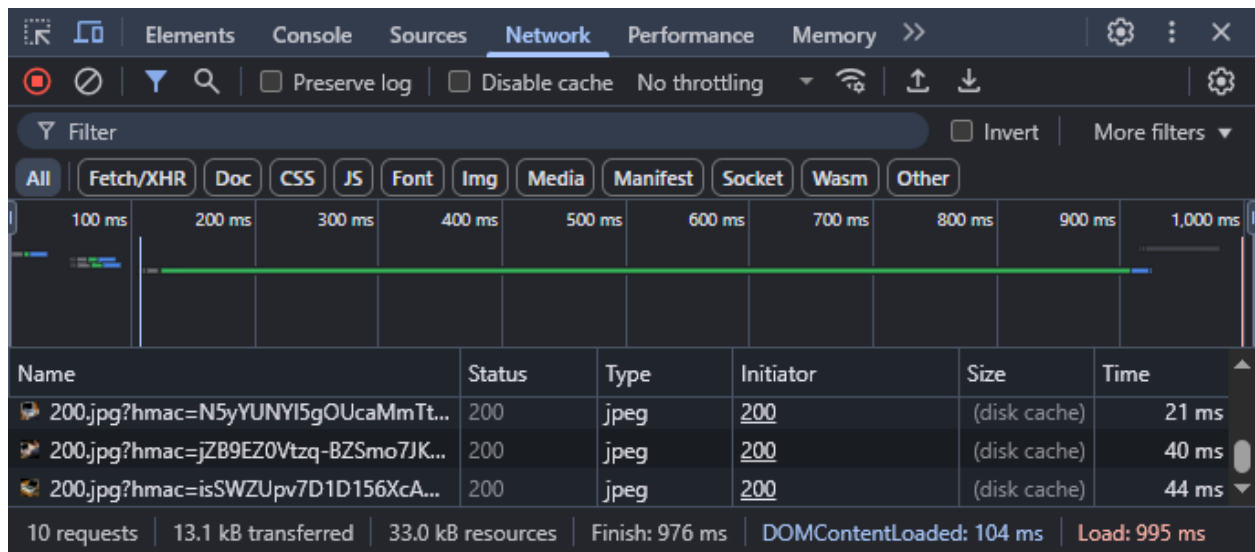


## Página 05

### Sin Factorizar

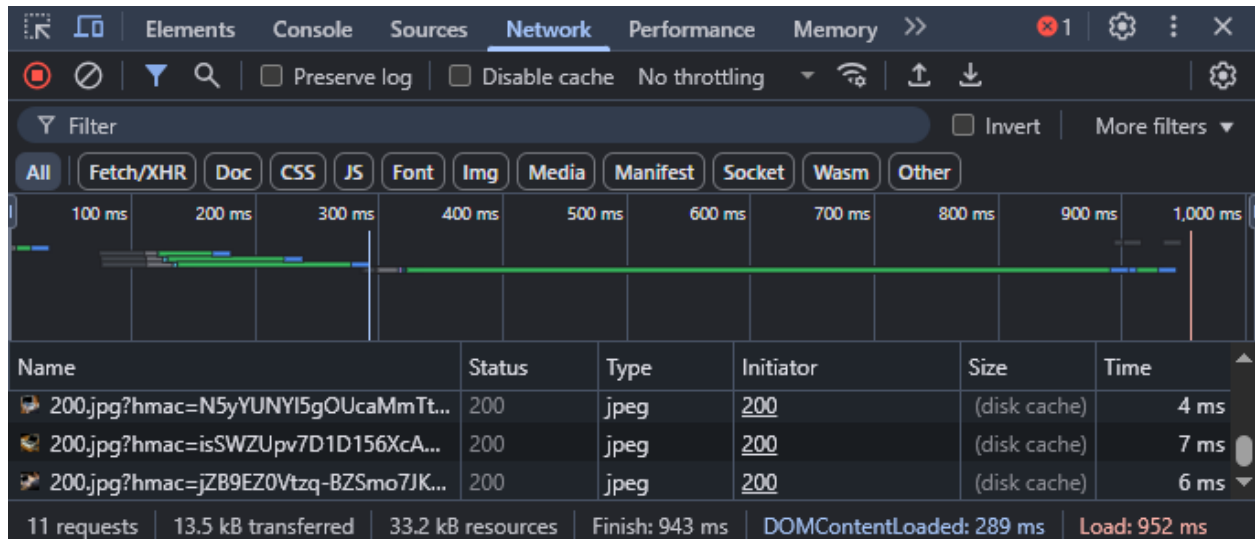


### Factorizada

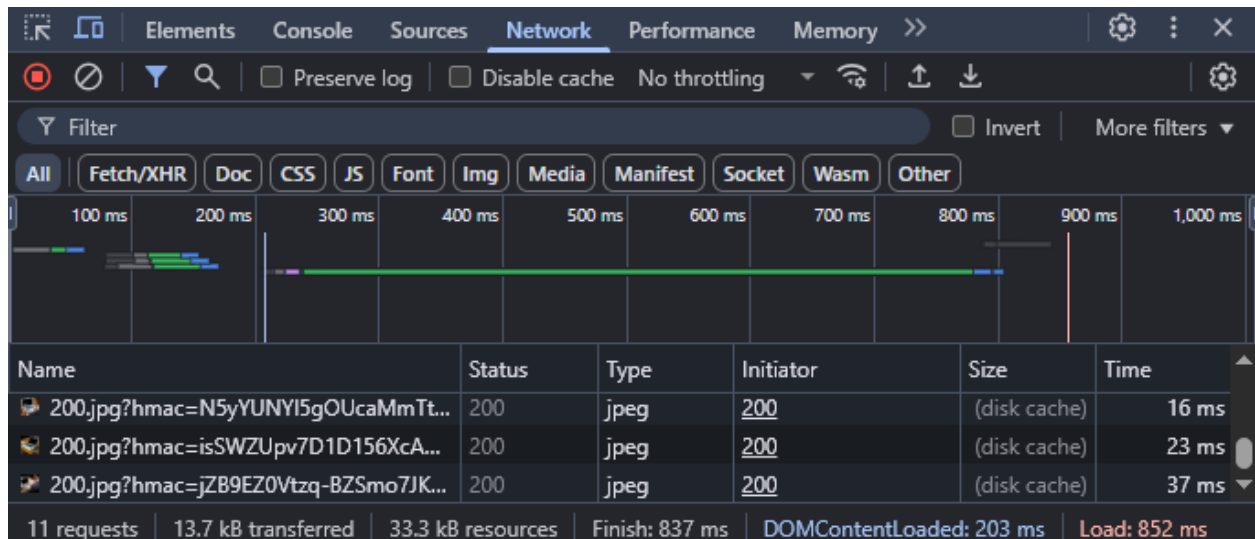


## Página 06

### Sin Factorizar

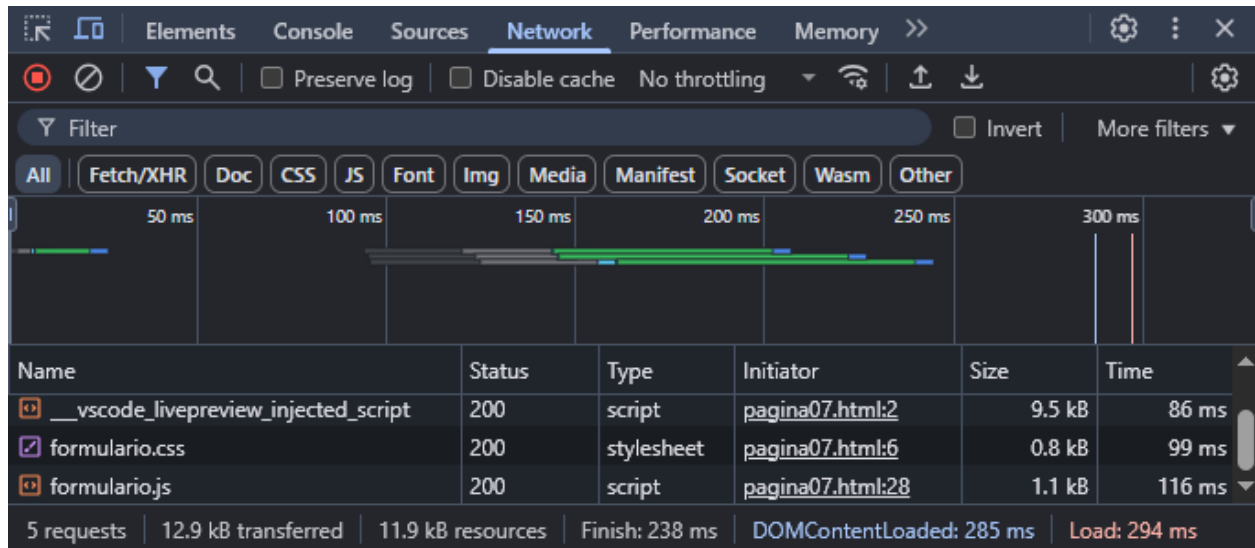


### Factorizada

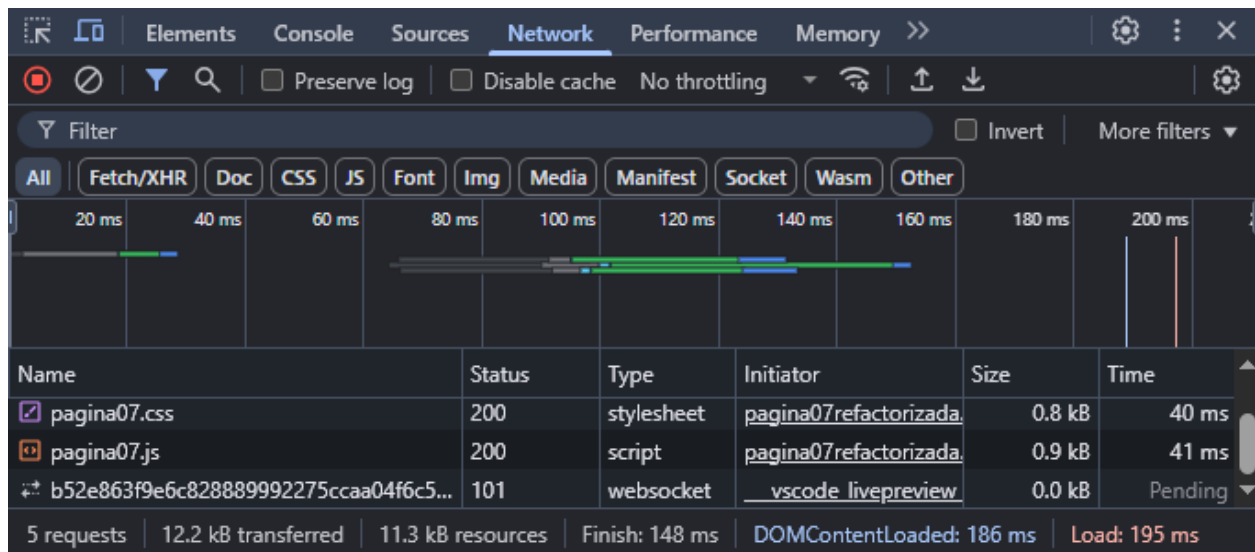


## Página 07

### Sin Factorizar



### Factorizada



## Código Sin Refactorizar

A veces tiene funciones mezcladas, código repetitivo y sin una estructura clara.

Definir variables en cada función para solo utilizarlas una vez puede hacer el código más pesado.

Todo el código está en un solo archivo grande

## Código Refactorizado

Separar la lógica en funciones reutilizables y módulos específicos hace que el código sea más fácil de leer.

Se evita definir muchas veces las mismas cosas.

En vez de mezclar toda la lógica en un solo archivo, separamos en archivos **JS**, **CSS** y **HTML**.

Podemos modificar cualquier parte sin afectar el resto debido a que separamos los archivos.

Es mucho más fácil encontrar errores sin afectar otras partes.

## En Conclusión

Aprendí que separar el HTML, JavaScript y CSS en distintos archivos nos permite que el proyecto sea más ordenado y fácil de entender.

Cada uno cumple su función:

**HTML** estructura el contenido, **JavaScript** maneja la lógica y las interacciones y **CSS** controla la presentación visual.

También mejora los tiempos de carga ya que al dividir los archivos la carga de la página se disminuye ya que no carga todo en una sola.

También ayuda a poder hacer crecer el proyecto en un futuro de una forma más sencilla ya que se encuentra todo mejor organizado y dividido.