Omar Teixeira González,

UO281847

*Informe Ejercicio-2-Tarea-1*

*XML*

***Software y estándares para la Web***

# Índice.

[1 Forma de uso del programa. 3](#_Toc118498386)

[2 Versión del compilador o intérprete utilizado. 3](#_Toc118498387)

[3 Informe sobre los documentos HTML y CSS. 4](#_Toc118498388)

[3.1 Capturas de pantalla con el resultado de la herramienta de validación de HTML para cada documento del sitio web. 4](#_Toc118498389)

[3.2 Capturas de pantalla con el resultado de la herramienta de validación de CSS para cada hoja de estilo del sitio web. 5](#_Toc118498390)

[3.2.1 estilo.css: 5](#_Toc118498391)

[3.2.2 layout.css: 6](#_Toc118498392)

[3.3 Comprobación de la adaptabilidad. 6](#_Toc118498393)

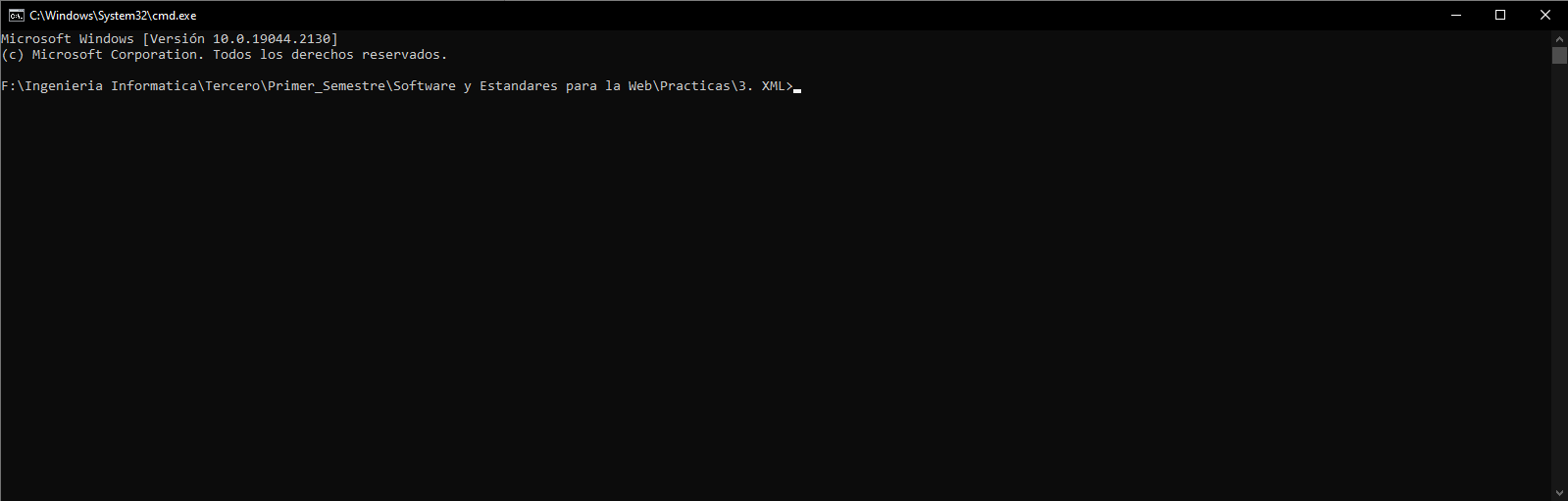
[3.4 Capturas de pantalla con la herramienta Screenfly de la visualización de cada documento del sitio web en al menos un móvil y una Tablet. 7](#_Toc118498394)

[3.5 Comprobación de la accesibilidad. 8](#_Toc118498395)

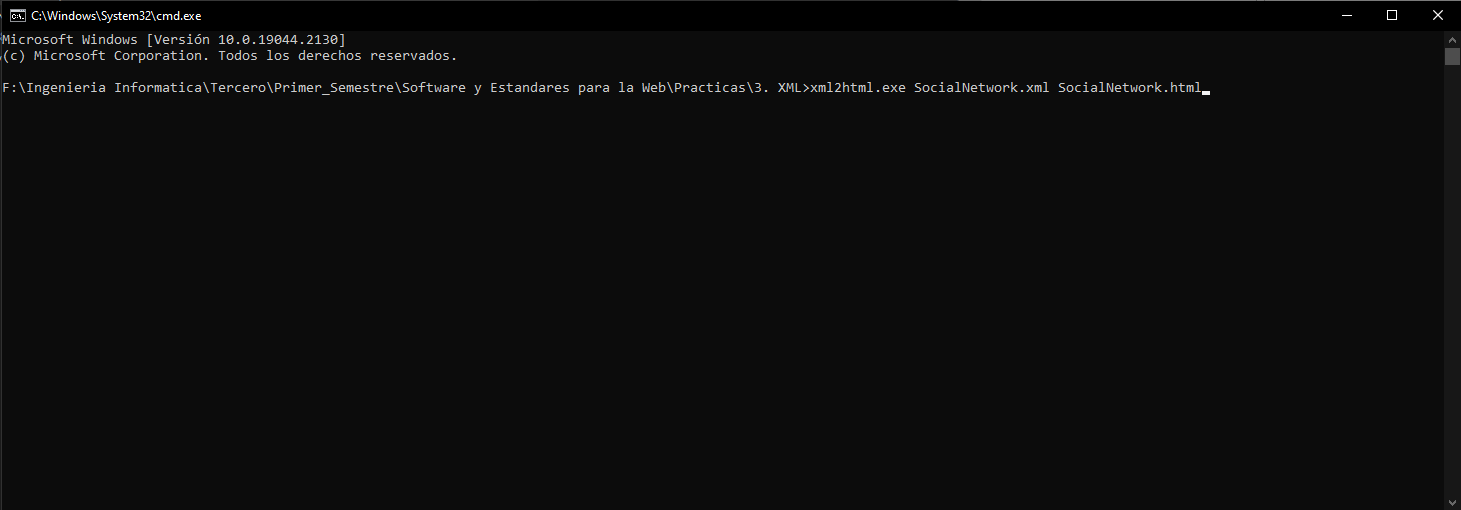
[3.5.1 Capturas de pantalla con el resultado de las 2 herramientas de verificación de la accesibilidad nivel AAA para cada documento HTML5. 8](#_Toc118498396)

# Forma de uso del programa.

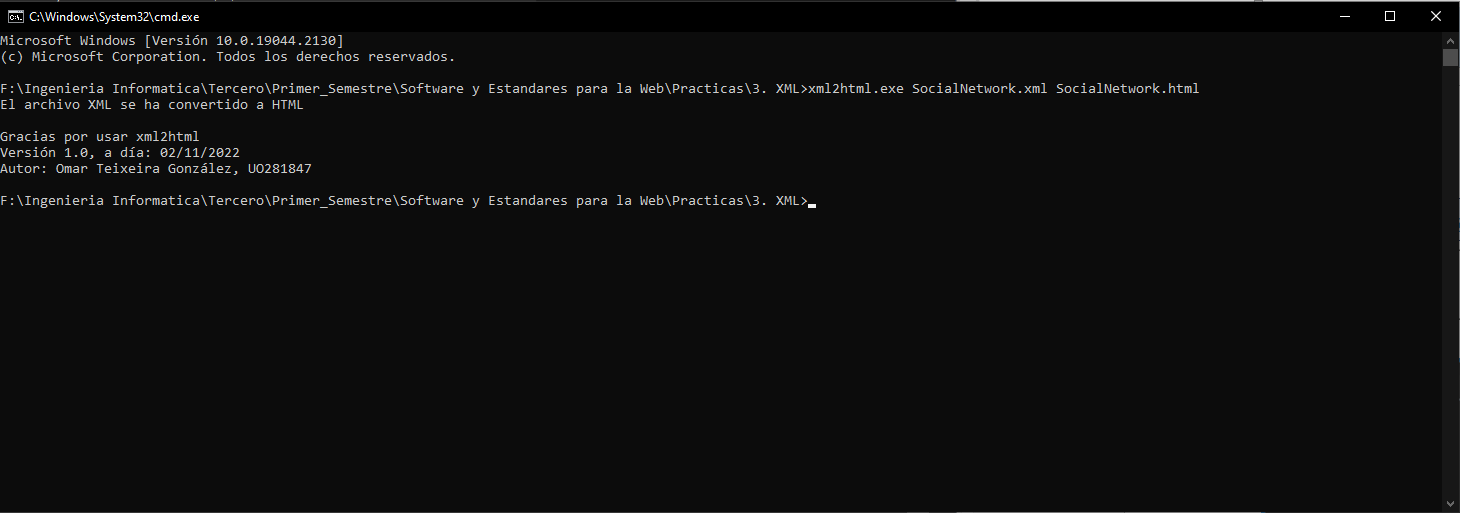
El uso del programa “xml2html.exe” es idéntico al programa dado en prácticas para transformar xml a svg, “xml2svg.exe”. De esta forma, el ejecutable debe de estar en el mismo directorio en el que se encuentra el fichero xml.



Posteriormente, se debe abrir mediante consola dicho directorio y ejecutar el programa de la siguiente manera:



“*xml2html.exe <archivo\_xml\_entrada> <archivo\_html\_salida>*”.

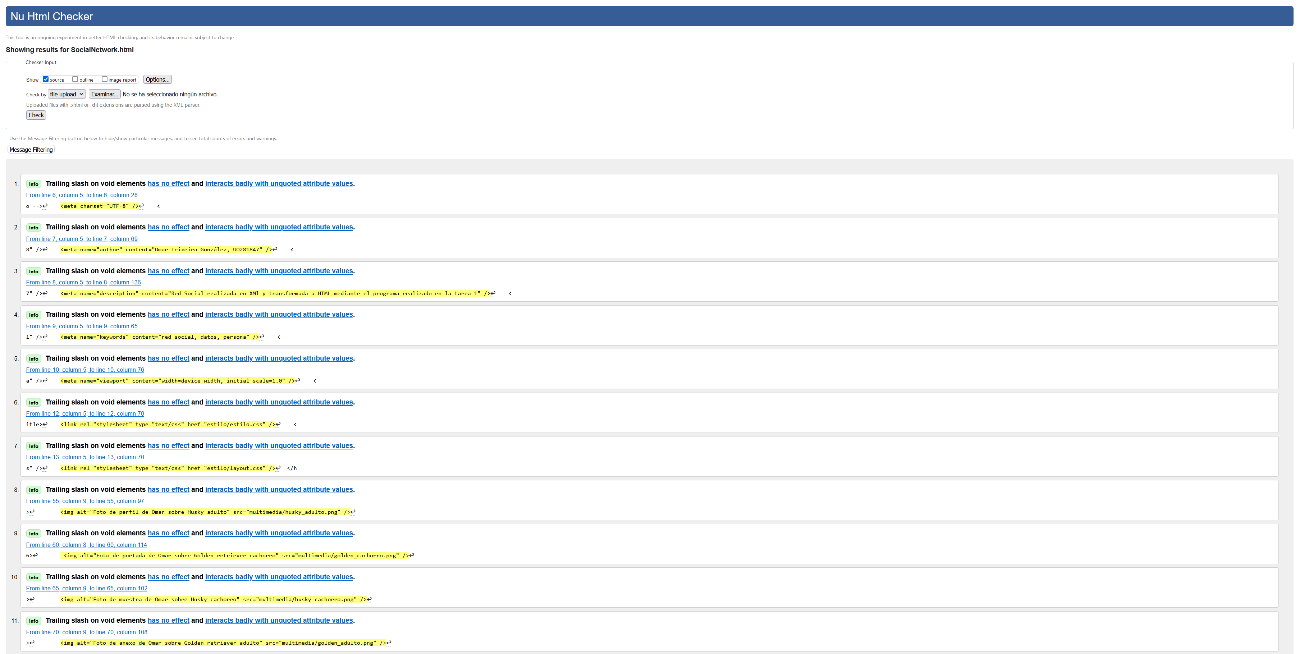


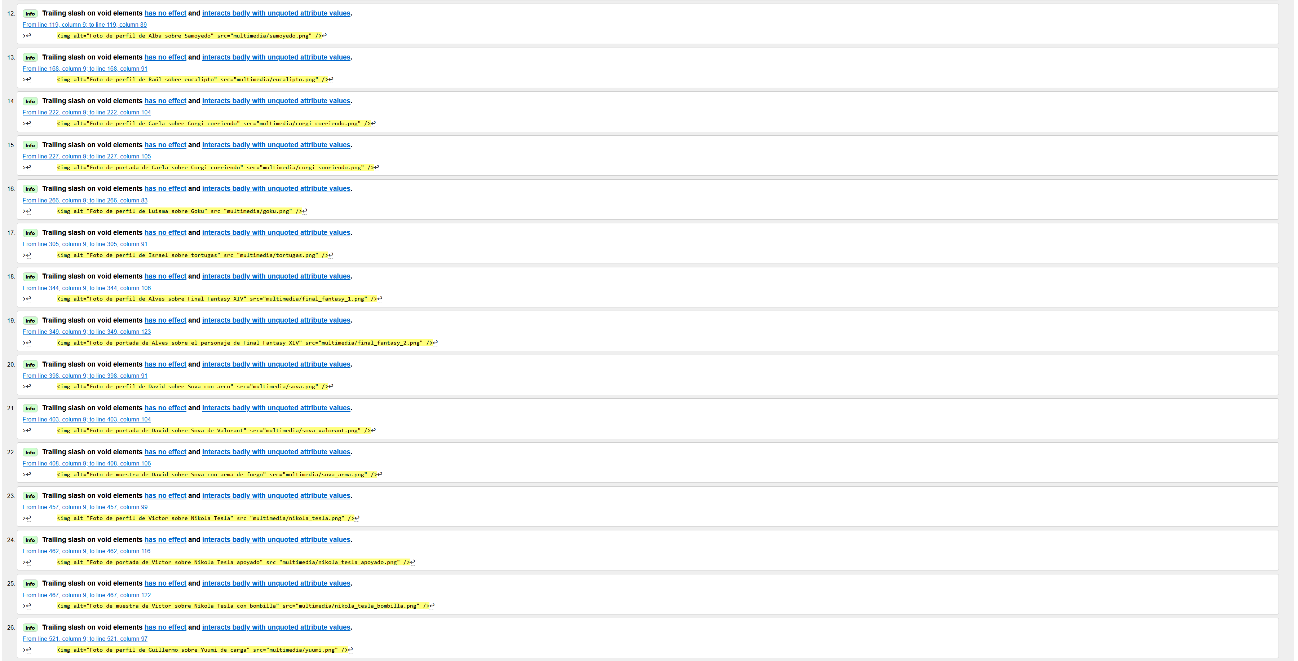
# Versión del compilador o intérprete utilizado.

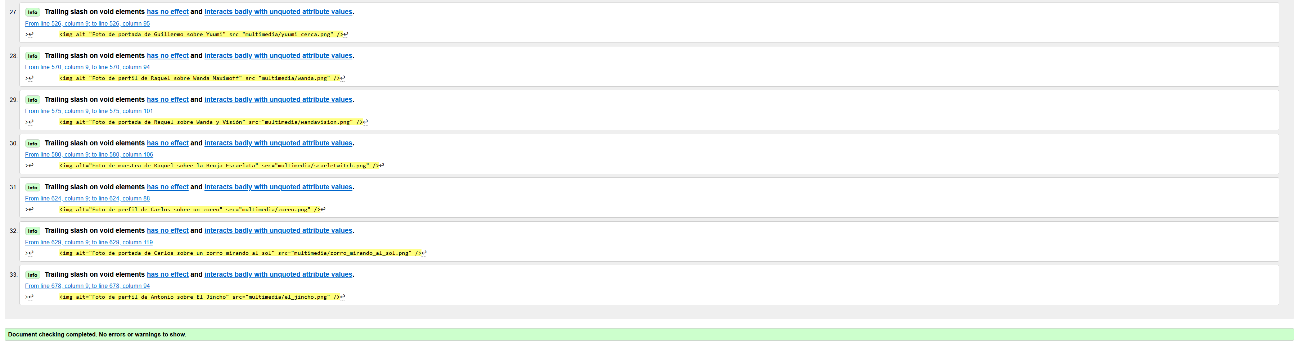
La versión utilizada es .NET3.1, ha sido realizado mediante lenguaje C# usando el entorno de Visual Studio 2022.

# Informe sobre los documentos HTML y CSS.

## Capturas de pantalla con el resultado de la herramienta de validación de HTML para cada documento del sitio web.

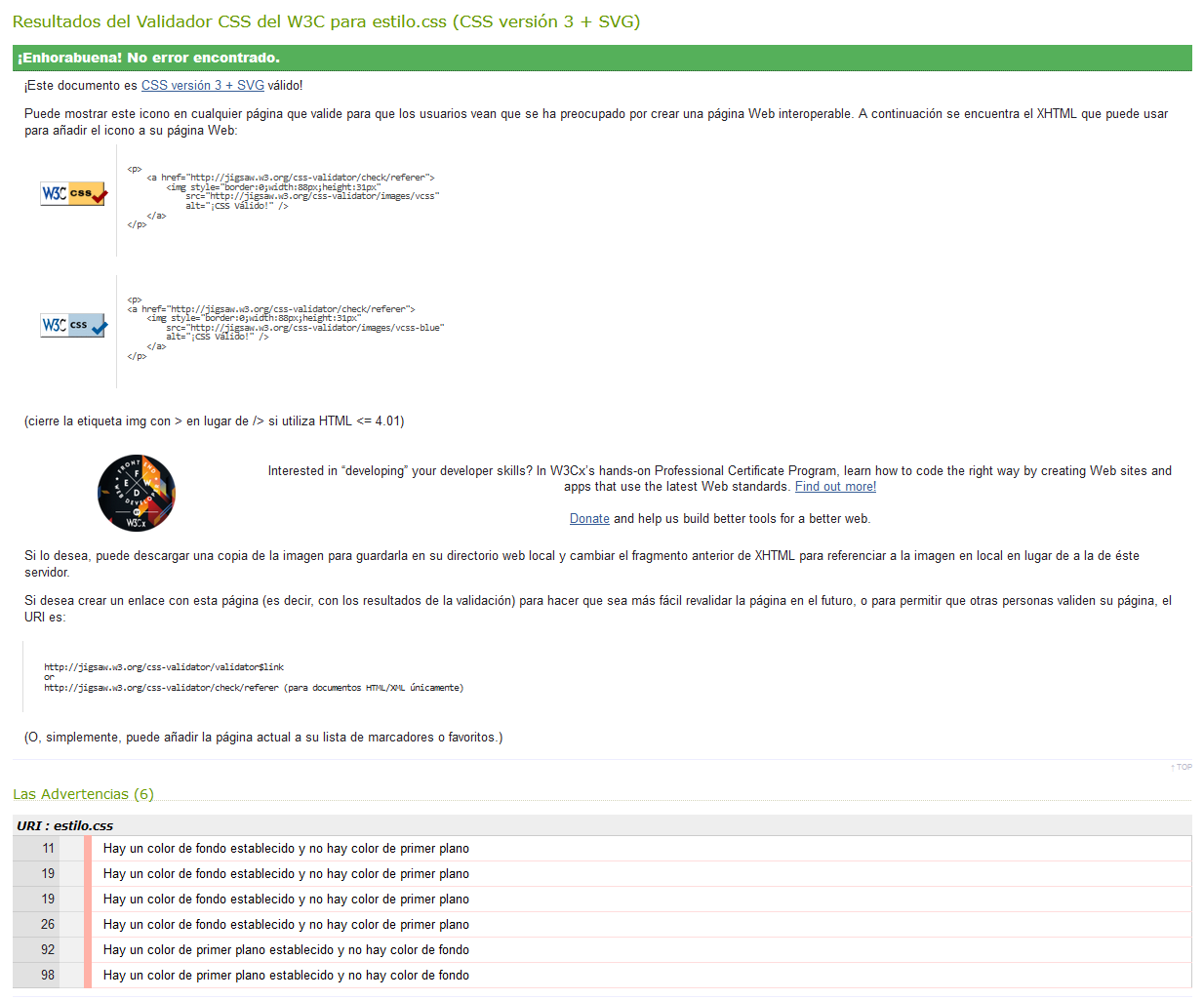






## Capturas de pantalla con el resultado de la herramienta de validación de CSS para cada hoja de estilo del sitio web.

### estilo.css:



Justificación de las advertencias:

La línea 11 corresponde a herencia de color de letra por defecto en *body*.

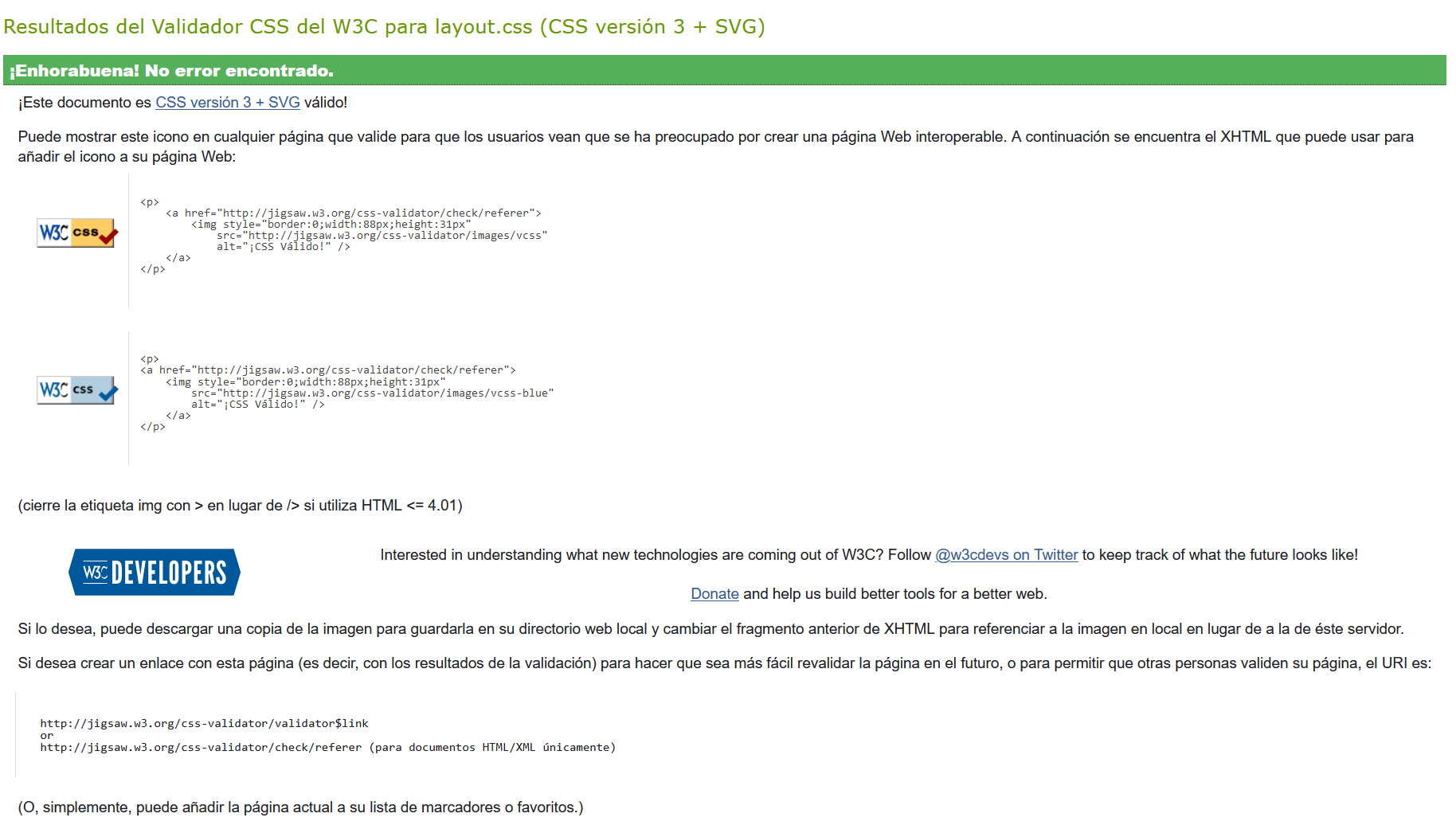
La línea 19 corresponde a herencia de color de letra de *body* en *header* y *main*.

La línea 26 corresponde a herencia de color de letra de *main* en *section*.

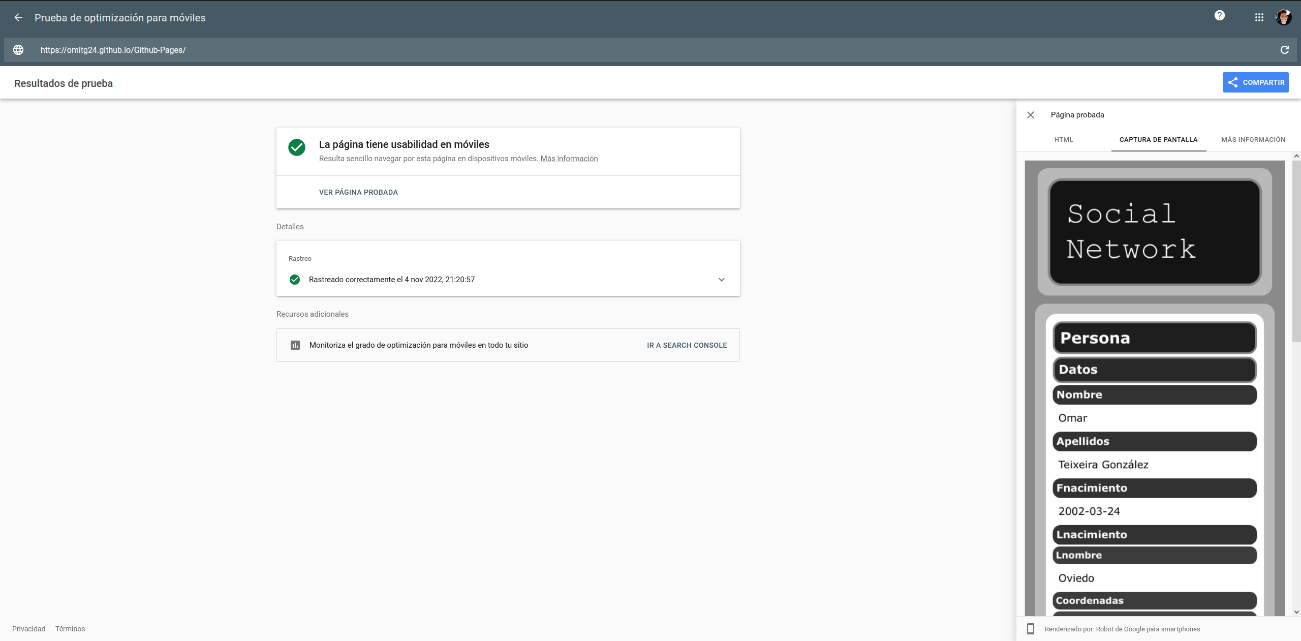
La línea 92 corresponde a herencia de color de fondo de *main* en *a*.

La línea 98 corresponde a herencia de color de fondo de *main* en *a:hover*.

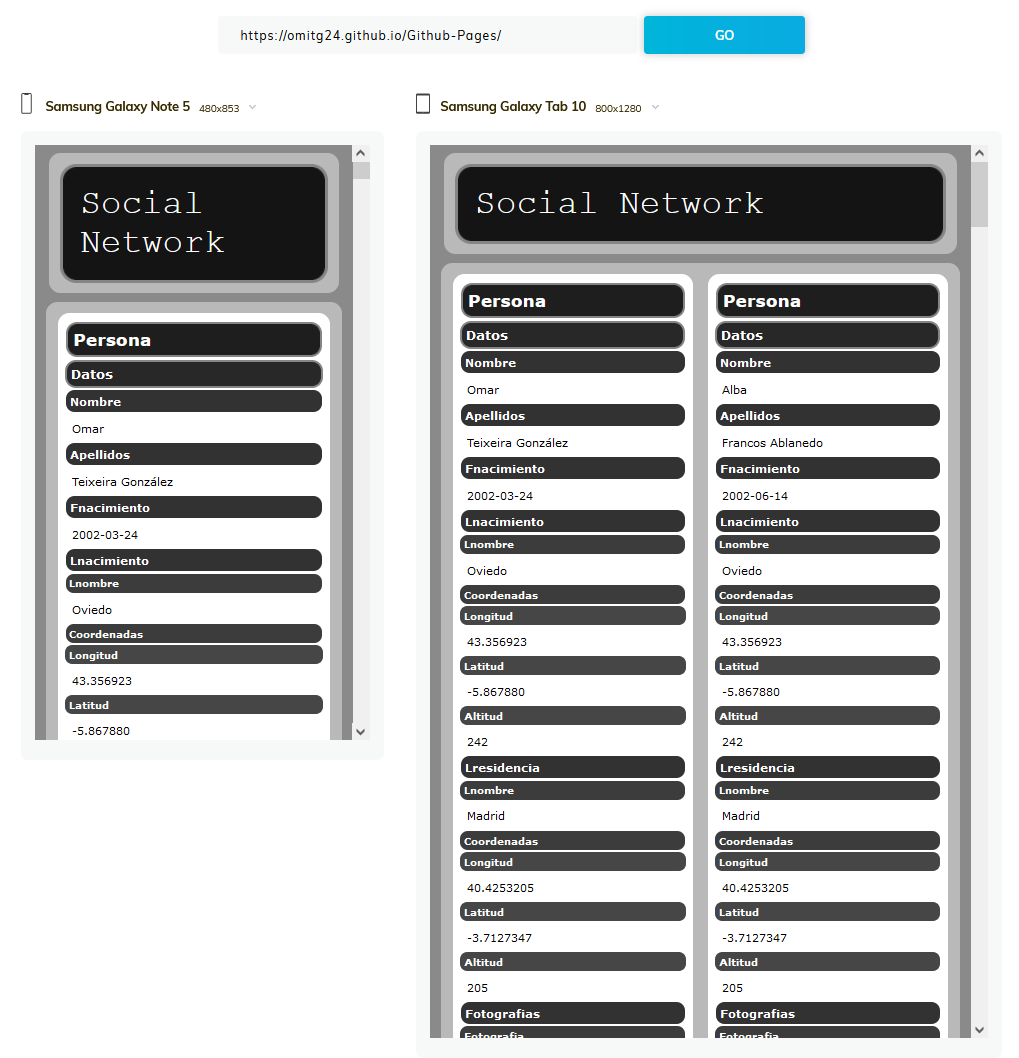
### layout.css:



## Comprobación de la adaptabilidad.

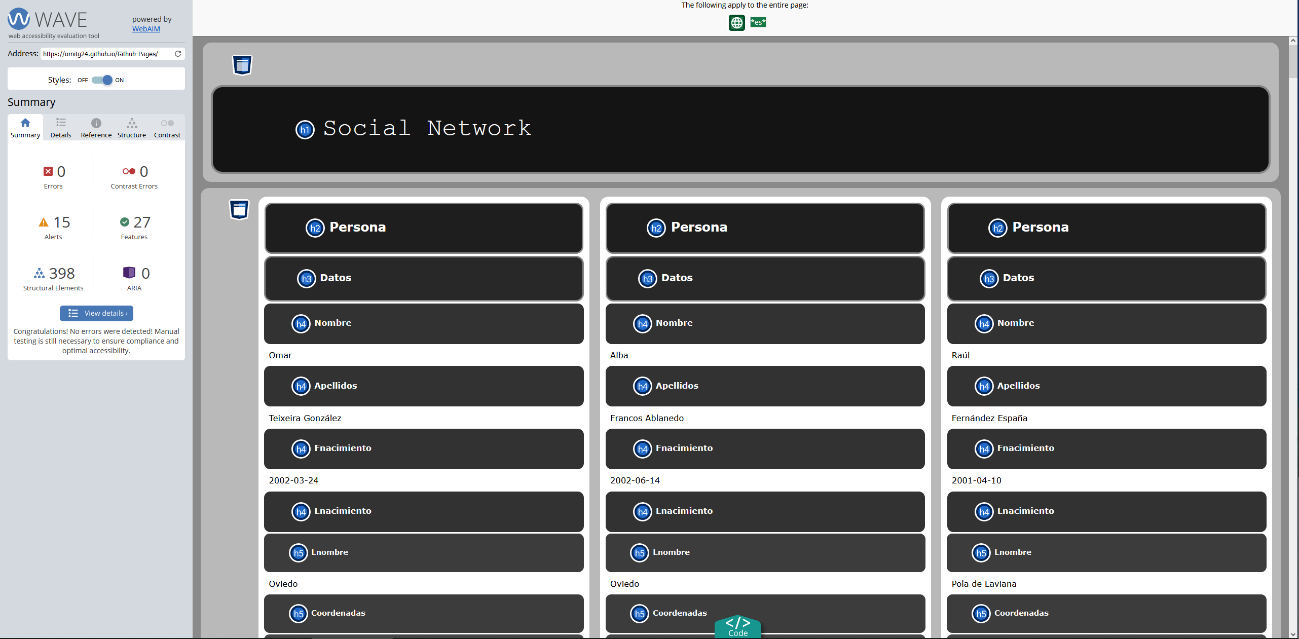


## Capturas de pantalla con la herramienta Screenfly de la visualización de cada documento del sitio web en al menos un móvil y una Tablet.



## Comprobación de la accesibilidad.

### Capturas de pantalla con el resultado de las 2 herramientas de verificación de la accesibilidad nivel AAA para cada documento HTML5.



En las capturas que se muestran a partir de la siguiente página, se ven los distintos errores que AChecker da, sin embargo, estos errores son en un color de letra negro (o azul oscuro para los enlaces y rojo para cuando se sitúa el ratón sobre ellos, mediante *a:hover*) sobre un fondo blanco puro, por lo que, en una primera instancia no debería de dar error, sin embargo, AChecker los detecta como errores por alguna razón.

Los colores utilizados en el texto general son:

* background-color: #FFFFFF
* color: #000000

Los colores utilizados en los enlaces son:

* background-color: #FFFFFF
* color: #002952

Los colores utilizados en los enlaces al situarse sobre ellos son:

* background-color: #FFFFFF
* color: #8C0000

Como se puede observar, pasándolo por el validador de contraste de *Wave* y el *ContrastRatioCalculator*, estos colores pasan las pruebas necesarias de contraste.

