**Xavier Ledesma Pons**

**Enlace al repositorio:** <https://github.com/xalepo4/HtmlAndCssI>

**Enlace web pública:** <https://agitated-poincare-1e3e8f.netlify.app>

**Documentación**

Para iniciar esta PEC, como en la PEC anterior se uso erróneamente UOC boilerplate, primeramente se ha actualizado el fichero **package.json** creando un boilerplate propio llamado **html-css-boilerplate**.

Este boilerplate implementa los siguientes scripts principales:

* **dev**: ejecuta el script **clean** que borra la **carpeta dist, .cache y .cache\_loader** y posteriormente ejecuta el script **parcel:dev** que ejecuta **parcel** sobre el fichero inicial del proyecto.
* **build**: ejecuta el script **clean** que borra la carpeta **dist, .cache y .cache\_loader** y posteriormente ejecuta el **parcel:build** que genera el contenido de la web en la carpeta **dist**.

Como dependencias de desarrollo, se ha instalado **autoprefixer, npm-run-all, parcel-bundler, parcel-plugin-imagemin, rimraf y sass.**

* **autoprefixer**: se ha creado el fichero **.postcssrc** con la siguiente configuración**: { “plugins”: { “autoprefixer”: true }}**. De esta manera, se añaden automáticamente los prefijos del navegador al código CSS que generamos (-**ms para Explorer, -webkit para Chrome, -moz para Mozilla y -o para Opera**).
* **npm-run-all**: permite ejecutar diversos scripts secuencialmente.
* **parcel-bundler:** permite generar y empaquetar todos los contenidos de la web.
* **parcel-plugin-imagemin**: permite reducir el tamaño de las imágenes. Con el fichero de configuración **imagemin.config.js** se pueden seleccionar los plugins que queremos usar para reducir nuestras imágenes. En este caso, he cogido el fichero por defecto y he añadido en el apartado **svgo** el plugin **{“minifyStyles”: false }**. De esta manera, evito que los svg pierdan las animaciones **CSS**.
* **rimraf**: paquete que permite borrar ficheros del sistema.

Como dependencias normales he añadido **@fortawesome/fontawesome-free** para tener una librería de iconos para la web.

Para cumplir con las reglas básicas **WCAG 2.1 A y AA** de accesibilidad, he restructurado las paginas que ya tenia creadas de la página anterior siguiendo las siguientes reglas:

* Una sección **header** donde se encuentra el titulo y el logo del sitio web. Como es un logotipo he optado por usar ficheros **PNG** que puede tener transparencia, aunque este no haya sido el caso.

Interfaz de usuario gráfica

Descripción generada automáticamente

Ilustración : Header con el titulo y una imagen de fondo

* Una sección **nav** donde se encuentra la sección superior de navegación.



Ilustración : Barra superior de navegación

* Una sección **main** donde se encuentra el contenido principal de cada página.
* Una sección **footer** donde se encuentra el pie de pagina de cada pagina. Este contiene el logo del sitio web y una barra de navegación.



Ilustración : Footer con logo de la web y navegación

En cuanto al **footer**, he seleccionado una combinación de colores que permita una correcta visualización en personas con dificultades visuales, ya que las reglas mencionadas anteriormente requieren un contraste alto entre fondo y texto.

**Página de portada**

En la portada, he editado la imagen central con clip-path. Mediante el comando CSS **clip-path: polygon(0% 0%, 100% 0%, 100% 75%, 75% 75%, 75% 100%, 50% 75%, 0% 75%)** se define la siguiente forma de bocadillo:

Una bandeja con varios tipos de comida

Descripción generada automáticamente

Ilustración : Imagen recortada con clip-path

**Página de detalle**

En las paginas de detalle se ha añadido una imagen destacada que se adaptara tanto para dispositivos móviles como para versión de escritorio. Esta imagen destacada incluye la técnica de gestión del arte. Para ello, se han cogido las imágenes y se han recortado con **Inkscape** seleccionado la región más representativa de esta. Se ha elegido el formato **JPG** para todas las imágenes de la página de detalle porque no se van a requerir transparencias.

Un ejemplo donde podemos apreciar este comportamiento en versión escritorio y móviles es el siguiente:

Un plato de comida diferente

Descripción generada automáticamente

Ilustración : Imagen destacada en versión escritorio

Un plato con comida

Descripción generada automáticamente

Ilustración : Imagen destacada en versión móvil

Tal y como se puede apreciar, en dispositivos móviles se eliminan zonas de las imágenes que no son relevantes para el usuario.

Además, como hay que añadir dos imágenes más, he añadido una imagen donde aparecen unos cocineros. Para poner a prueba los nuevos conocimientos adquiridos, he usado la técnica **resolution switching different sizes** con el código siguiente:

<img srcset="assets/images/elaboracion-480w.jpg 480w, assets/images/elaboracion-800w.jpg 800w"

sizes="(max-width: 600px) 480px, 800px"

src="assets/images/elaboracion-800w.jpg"

alt="Elaboración en una cocina">

Con las líneas anteriores, he indicado que tenemos una imagen que se llama **elaboracion-800w.jpg** y que para resoluciones inferiores a **600px** se cargara la imagen **elaboracion-480w.jpg**. Si el tamaño de pantalla es mayor se cargará la imagen **elaboracion-800w.jpg**.

Además, he aprovechado el diseño de la página de categorías para añadir las recetas populares con su correspondiente imagen destacada y así cumplir con el requerimiento de añadir 2 imágenes.

**Página de categoría**

En esta página me he decidido por un diseño usando un **flex layout**. En cada celda del layout he incluido la imagen destacada de la receta y un enlace a la página de detalle. Para dar un toque especial a las imágenes, he decidido redondearlas mediante **CSS**.

El resultado final en dispositivos móviles y escritorio es el siguiente:

Un plato de comida

Descripción generada automáticamente con confianza media

Ilustración : Categorías en versión escritorio

Imagen de la pantalla de un celular con la foto de comida

Descripción generada automáticamente con confianza media

Ilustración : Categorías en versión móvil

**Página de presentación**

En esta página se pide que se añada una imagen **SVG** animada con **CSS**. Para ello, me he descargado un SVG de internet de una librería de **SVG’s** de código abierto y he editado manualmente el código para generar la animación.

Como se observa en la siguiente captura la imagen seleccionada es la siguiente:

Icono

Descripción generada automáticamente

Ilustración : Imagen SVG editada

En el código, he agrupado en una clase llamada **exclamation** el circulo y el path del signo de admiración.

Mediante el código CSS siguiente, he aplicado a la clase **exclamation** la propiedad **opacity: 1;** para que se inicie en color lila el signo de admiración. Como animación he creado un **keyframe** que mantiene la **opacidad a 1 en el 0%, el 10% y el 15%** y la **opacidad a 0 en el 100%.** Esta animación se ejecutará de manera infinita.

<style type="text/css">

@keyframes showhide {

0% { opacity: 1; }

10% { opacity: 1; }

15% { opacity: 1; }

100% { opacity: 0; }

}

.exclamation {

opacity: 1;

animation: showhide 2s ease infinite;

}

</style>

Como se puede observar, en esta página tanto la animación **SVG** como el mapa de Cataluña (en **PNG** para trabajar en un formato distinto) se adapta perfectamente a versión de escritorio y versión móvil.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Ilustración : Imagen SVG en versión escritorio

Imagen que contiene Texto

Descripción generada automáticamente

Ilustración : Mapa en versión escritorio

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Ilustración : Imagen SVG en versión móvil

Mapa

Descripción generada automáticamente

Ilustración : Mapa en versión móvil

Además, para practicar y añadir un caso de **resolution switching, same size, different resolutions** con la imagen del mapa, he añadido el siguiente código:

<img srcset="assets/images/cataluña-mapa-320w.png,

assets/images/cataluña-mapa-480w.png 1.5x,

assets/images/cataluña-mapa-640w.png 2x"

src="assets/images/cataluña-mapa-640w.png"

alt="Mapa de Cataluña">

Con las líneas anteriores, he indicado que tenemos una imagen llamada **cataluña-mapa-640w.png**. Si nuestro dispositivo tiene una resolución estándar, se cargará la imagen **cataluña-mapa-320w.png**. Si el dispositivo tiene **1.5 pixeles por píxel CSS** se cargará la imagen **cataluña-mapa-480w.png**. Si el dispositivo tiene **2 pixeles por píxel CSS** se cargará la imagen **cataluña-mapa-640w.png.**

**Página de enlaces**

Por último, en la página de enlaces, he añadido todos los recursos de donde provienen y añadido una imagen en formato **GIF**, ya que en las paginas anteriores había **usado JPG, PNG y SVG** y me faltaba usar un **GIF**.

En ninguna página he usado ninguna **media query** tal como en el foro se especifica ya que esta técnica es más ineficiente que gestionar correctamente el tamaño de los recursos.

**Validación de reglas**

Mediante la herramienta **Wave**, he validado la accesibilidad de mi sitio web para evitar errores con las reglas que se deben aplicar al desarrollar. Inicialmente he tenido algún problema con el contraste del pie de pagina, pero finalmente he obtenido una combinación óptima para evitar problemas a personas con dificultades visuales. También he resuelto algunas advertencias sobre el correcto uso de los niveles de heading, ya que al usar <h3> sin un previo <h2> puede generar problemas con herramientas de assistive technology.

Tal y como se adjunta a continuación, podemos ver el resultado de la herramienta Wave para cada una de nuestras páginas:

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Ilustración : Resultado validación de la portada

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Ilustración : Resultado validación categorías

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Ilustración : Resultado validación página de detalle

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Ilustración : Resultado validación de la presentación

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Ilustración : Resultado validación de enlaces

**Comparativa del tamaño de las imágenes al usar imagemin**

|  |  |  |  |
| --- | --- | --- | --- |
| **Recurso** | **Tamaño original** | **Tamaño reducido** | **Optimización** |
| cataluña-mapa-320w.png | 108KB | 25KB | 76,85% |
| cataluña-mapa-480w.png | 220KB | 44KB | 80% |
| cataluña-mapa-640w.png | 360KB | 69KB | 80,83% |
| collage-platos-catalanes.jpeg | 429KB | 258KB | 39,86% |
| comida-bg.jpeg | 263KB | 237KB | 9,89% |
| elaboracion-320w.jpg | 16KB | N/A | N/A |
| elaboracion-480w.jpg | 30KB | 19KB | 36,67% |
| elaboracion-640w.jpg | 46KB | N/A | N/A |
| elaboracion-800w.jpg | 66KB | 41KB | 37,88% |
| escalivada-sm.jpeg | 154KB | 127KB | 17,53% |
| escalivada.jpeg | 149KB | 94KB | 36,91% |
| esqueixada-sm.jpeg | 177KB | 108KB | 38,98% |
| esqueixada.jpeg | 225KB | 132KB | 41,33% |
| exclamacion.svg | 5KB | 4KB | 20% |
| fricando-sm.jpeg | 130KB | 78KB | 40% |
| fricando.jpeg | 125KB | 74KB | 40,8% |
| internet.gif | 158KB | 158KB | 0% |
| logo.png | 11KB | 2KB | 81,81% |
| trinxat-sm.jpeg | 205KB | 121KB | 40,98% |
| trinxat.jpeg | 194KB | 114KB | 41,24% |

Como podemos observar hay dos recursos que no existen debido a que no se usan y por tanto parcel ya no los compila en la carpeta dist. En el resto, he calculado el porcentaje de optimización de cada imagen y claramente se observa una bajada de peso en todas las imágenes, que se traslada en un menor tiempo de carga de los recursos cuando se cargan las paginas.

Tal y como podemos observar, puedo sacar como conclusión que las imágenes **PNG** están mejor optimizadas que las que están el **JPG**, **SVG** y **GIF**. Utilizando este plugin ahorramos en tamaño de imágenes un total de **43,34%** respecto los ficheros originales.