# **Prueba Técnica Castor Data – Daniel García**

## **Qué hice**

* Realicé una réplica basada en el PDF **LANDING EPSON** dentro de los recursos disponibles.
* Se utilizaron las tecnologías **HTML5, CSS3** y **JavaScript vainilla** para el carrusel de imágenes.
* La idea fue darle vida al diseño del documento, pero ya en formato web.

## **Cómo construí la página**

### 1. Basado en el PDF

* Revisé la estructura sugerida en el documento (logo, menú, productos) y la adapté al código.
* Usé **HTML** para la estructura, **CSS** para el estilo y un poco de **JavaScript** para la interactividad.

### 2. Elementos añadidos

* **Footer de contactos**: agregué un pie de página con correo, teléfonos, dirección y horarios.
  + No solo le da un cierre visual sino que también mejora la usabilidad.
  + Este cuenta con un diseño totalmente **responsive**.
  + Incluye una clase que **oculta información en responsive**.
* **Carrusel con JavaScript**: cada categoría de productos (impresoras, plotters, tintas y proyectores) tiene un carrusel que se mueve solo y también puede controlarse manualmente.

### 3. Diseño y estilo

* Mantengo los colores corporativos de **Epson** (azul y tonos claros).
* La página es **responsiva**, adaptándose a distintos tamaños de pantalla.

## **Ventajas técnicas y de SEO**

* **Estructura clara y semántica**: usé etiquetas como <header>, <nav>, <section> y <footer> para que los buscadores entiendan mejor el contenido.
* **Imágenes con alt**: ayudan al SEO y también a la accesibilidad.
* **Contenido organizado por títulos (<h2>)**: cada sección de producto tiene su encabezado, lo que facilita la indexación en Google.
* **Página ligera y rápida**: solo usa HTML, CSS y un poco de JS → buena experiencia de usuario y mejor posicionamiento.

## **Detalles técnicos**

### HTML

* Cuenta con sintaxis **semántica**, manteniendo SEO para la página.
* Incluye **comentarios explicando el código** y está ordenado.

### CSS

* Cuenta con **variables globales** definidas para \*, body y html.
* Se manejan clases para el **header** y para cada sección (impresoras, plotter, proyectores y tintas).
* Incluye clases para el **carrusel**.
* El archivo mantiene un **orden descendente** y se encuentra **comentado por secciones**, finalizando con el diseño responsivo.

### JavaScript

* Todo el código está **comentado línea por línea**, explicando las funcionalidades implementadas en el carrusel.

### Fuentes

* Se implementaron los tipos de letra **Roboto** y **Roboto Italic**, disponibles en assets/fonts.

### Footer

* Se agregó un **footer totalmente responsivo** para mostrar los datos de contacto.

### Botón Ver más

* Se modificó el botón de Ver más con un efecto **hover en azul**, manteniendo la fidelidad del color representativo de la marca.

### Carrusel

* Se agregaron **puntos bajo el carrusel** de 4 imágenes (modificables).

### Control de versiones

* Se implementó un sistema de **control de versiones con Git**.
* El proyecto fue subido a **GitHub**, lo que permite llevar historial de cambios, colaborar fácilmente y mantener un flujo de trabajo ordenado

### Mejoras propuestas

* **Modularización del JS**: separar el código del carrusel en un archivo .js independiente para mantener el proyecto más ordenado.
* **Escalabilidad**: si en el futuro se quiere mostrar más productos, se podría conectar la página a un **JSON** o a una **API** (también puedo hacer backend) para que los carruseles se generen dinámicamente.