

MEMORIA EXPLICATIVA

Proyecto final: Diseño Web con HTML5, CSS3 Y Javascript.

Gabriel Martín Rodríguez.

CEI Sevilla.

23/01/2023

1. Introducción.

Este proyecto consiste en el diseño y desarrollo de una página web utilizando los lenguajes de programación HTML, CSS y Javascript. He elegido crear el *homepage* de un negocio ficticio, que fácilmente puede ser ajustado para satisfacer las necesidades de cualquier empresa pequeña. El negocio se llama *Plantree* y consiste en la venta al por menor de plantas de interior. Este tipo de empresas requieren de páginas webs con un enfoque visual minimalista, con alta presencia de imágenes y textos sencillos.

A lo largo de esta memoria explicaré el proceso de creación de la página web, desde la conceptualización y el diseño UI pasando por el planteamiento y la estructuración del contenido, hasta llegar a la escritura de código en lenguajes de programación front-end.

URL DE LA WEB: <https://proyectowebplantree.netlify.app/>

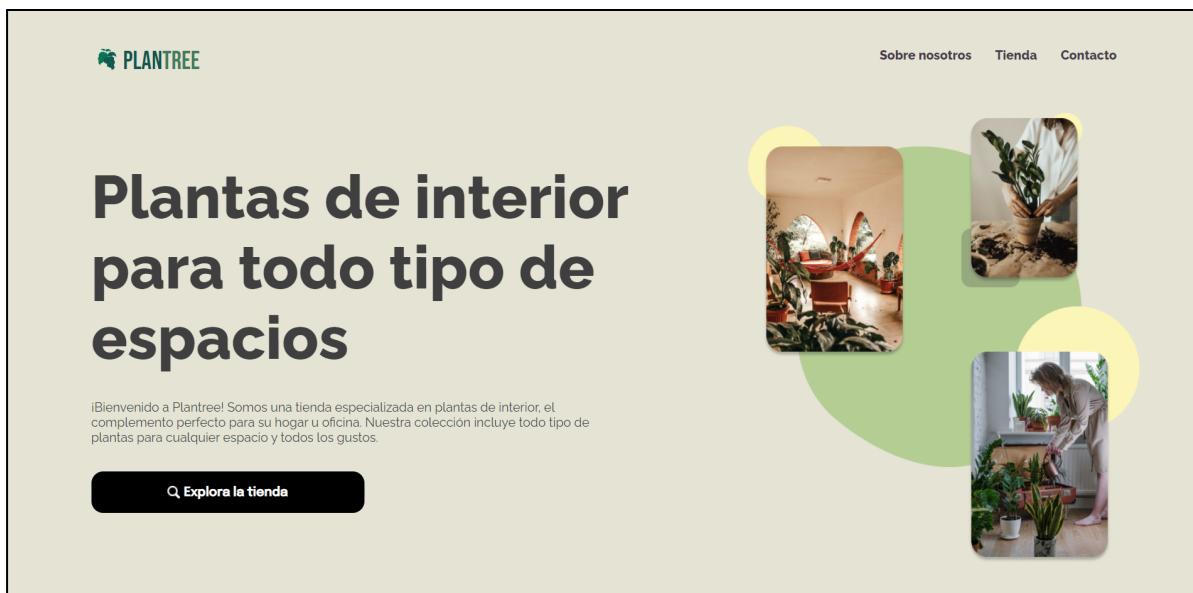
2. Conceptualización y diseño.

Para el prototipo he utilizado *Figma*, una herramienta de diseño de interfaz de usuario y creación de prototipos basada en la nube. La temática y el diseño de la página están basados en una práctica que había realizado previamente, reestructurando el contenido y la propuesta visual. La primera imagen corresponde con el diseño inicial y la segunda con el diseño final de esta página web. Cómo se puede observar, elementos como el logo y el título de la *homepage* han sido estilizados para encajar con la nueva propuesta de marca.



El prototipo inicial de la página web Plantree muestra la siguiente estructura:

- Barra superior:** Contiene el logo "PLANTREE" con un icono de planta verde, y tres enlaces: "CARACTERÍSTICAS", "VIDEO" y "PRECIOS".
- Contenido principal:** Un lema "Plantas de interior para todo tipo de espacios." y un formulario para introducir tu email, con un botón "SUBMIT".
- Sección "NUMEROSOS BENEFICIOS":** Ilustración de una planta en un recipiente, con el subtitulo "NUMEROSOS BENEFICIOS" y el texto: "Las plantas de interior reducen los niveles de dióxido de carbono, incrementan la humedad del aire, relajan y hacen más agradable el ambiente del hogar. También se ha comprobado que aumentan la eficiencia laboral en las oficinas."
- Sección "ASESORAMIENTO GRATUITO":** Ilustración de una planta en un recipiente, con el subtitulo "ASESORAMIENTO GRATUITO" y el texto: "En Plantree, amamos las plantas y queremos compartir nuestra pasión contigo ayudándote a elegir tus plantas en función de tus gustos y necesidades. Si tienes dudas, nuestros expertos te asesorarán para escoger las plantas que más se ajusten a tus necesidades."
- Sección "GRAN VARIEDAD DE PLANTAS":** Ilustración de una planta en un recipiente, con el subtitulo "GRAN VARIEDAD DE PLANTAS" y el texto: "Existe una gran variedad de colores, formas y tamaños de las hojas, por lo que siempre es posible combinar con cualquier tipo de decoración y colores de la estancia. El estilo de la maceta también es muy importante, para no desentonar con el resto de elementos decorativos."



El prototipo final de la página web Plantree muestra la siguiente estructura:

- Barra superior:** Contiene el logo "PLANTREE" y enlaces "Sobre nosotros", "Tienda" y "Contacto".
- Contenido principal:** Un lema "Plantas de interior para todo tipo de espacios" y un botón "Explora la tienda".
- Imagen central:** Una fotografía que muestra una sala de estar llena de plantas y un cuadro de imágenes que ilustra la actividad de cuidado y decoración de las plantas.
- Textos adicionales:** Un párrafo informativo sobre la bienvenida a Plantree y su colección de plantas.

Teniendo en cuenta el objetivo de esta página web, he optado por un diseño sencillo caracterizado por colores que aluden a la naturaleza, tonos tierra, textos grandes y un uso intencionado del espacio negativo. Con estas ideas como base surge una propuesta de diseño que posteriormente se modifica conforme llevo a cabo la programación de la página web. Mi primera intención fue utilizar el color beige como fondo, presente en toda la web.

PLANTREE

HOME CARACTERÍSTICAS VIDEO PRECIOS

NUMEROSOS BENEFICIOS

Las plantas de interior reducen los niveles de dióxido de carbono, incrementan la humedad del aire, relajan y hacen más agradable el ambiente del hogar. También se ha comprobado que aumentan la eficiencia laboral en las oficinas.

ASESORAMIENTO GRATUITO

En Plantree, amamos las plantas y queremos compartir nuestra pasión contigo ayudándote a elegir tus plantas en función de tus gustos y necesidades. Si tienes dudas, nuestros expertos te asesorarán para escoger las plantas que más se ajusten a tus necesidades.

GRAN VARIADAD DE PLANTAS

Existe una gran variedad de colores, formas y tamaños de las hojas, por lo que siempre es posible combinar con cualquier tipo de decoración y colores de la estancia. El estilo de la maceta también es muy importante, para no desentonar con el resto de elementos decorativos.

Pothos Golden

Improve your mood with some green splashed with touches of yellow. This climbing indoor plant can actually climb up walls and even space into a golden-green cascade. The Pothos Golden also known as Money Plant, is a plant that can grace your home or office because it is beautiful AND easy to keep healthy.

\$27.50 [COMPRAR](#)

Calathea Compact

The Prayer Plant brings elegance to any party. The unique green and red foliage of the Calathea Compact will fall at night and rise again to greet the new day. This Praying Plant brings elegance to the prayer plant party. The unique green and red foliage of the Calathea Compact Star will fall at night.

\$39.00 [COMPRAR](#)

Monstera Adansonii

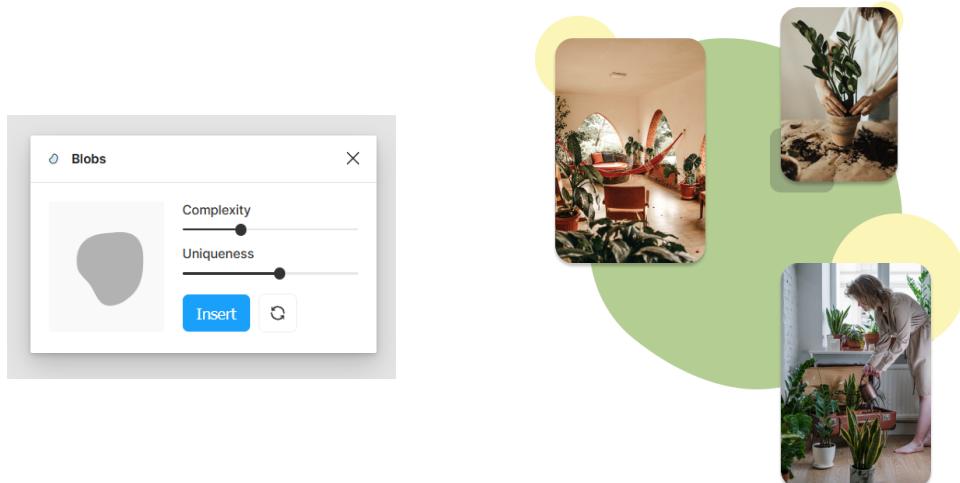
This unique, easy-care vine plant can be hard to find, but you come to the right spot! Named after the Swiss Cheese Vine, for its pockmarked leaves, it is fast-growing and loves to climb. This unique and elegant plant will add a touch of style while demanding minimal attention to thrive.

\$65.00 [COMPRAR](#)

Sin embargo, tras hacer varios bocetos en *Figma* me di cuenta de que las secciones orientadas al texto pierden legibilidad al usar tonos pesados como fondo. Solo mantuve el color de fondo en la página principal, que funciona como primera impresión de la marca y está más enfocada en lo visual que en el contenido.

Con esta intención, añado a la derecha un collage de figuras orgánicas sólidas e imágenes que ilustran la actividad de la empresa. Llevo a cabo este diseño en *Figma* con uno de sus plugins más populares: *Blobs*. Se trata de un autogenerador de

figuras curvilíneas que se han popularizado en los últimos años en el diseño gráfico y web. Normalmente se usa para anclar visualmente las *landing page*, actuando meramente como telón de fondo.



PALETA DE COLORES

El resto de la web usa el blanco como color de fondo y una paleta de colores limitada. Para los textos elijo un negro menos saturado. El logo combina dos tonos de verde oscuro, que también aparecen en los efectos de hover de algunos botones. El verde menta se usa para elementos de fondo, como alguno de los *blobs* o el fondo de los cajas en la sección de tienda. El amarillo también aparece en los *blobs* y en otros elementos decorativos de la web.



Para los gradientes de fondo he usado la página web uigradients.com, que permite elegir entre una colección de gradientes llamativos.

TIPOGRAFÍA

El texto de la web usa principalmente dos familias tipográficas sans serif: *Raleway* y *Epilogue*. He usado la primera en casi todo el cuerpo de la página con diferentes tipos de pesos y la segunda para botones como elemento diferenciador. Son tipografías modernas, minimalistas y legibles, adecuadas para textos descriptivos y que siguen la línea visual de la marca. Para el logo de la empresa he elegido *Bebas Neue*, una tipografía con más fuerza y llamativa, apropiada para logos y títulos.

Raleway

BEBAS NEUE

Epilogue

Estas fuentes no aparecen por defecto en la mayoría de dispositivos, por lo que hay que enlazarlas en el elemento HTML `<head>` a través de recursos externos. He usado Google Fonts para seleccionar los pesos que iba a necesitar en el diseño.

```
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>PLANTREE: Prototype</title>
  <link rel="preconnect" href="https://fonts.googleapis.com">
  <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>
  <link href="https://fonts.googleapis.com/css2?family=Bebas+Neue&family=Epilogue:wght@500&family=Raleway:wght@400;500;800&display=swap" rel="stylesheet">
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
  <link rel="stylesheet" href="css/estilos.css">
</head>
```

RESPONSIVE DESIGN

Tras diseñar la interfaz de usuario para pantallas grandes, llevo a cabo una serie de bocetos que ilustran cómo se vería la web en tablets y smartphones. En la vista de ordenador podemos hacer uso del espacio negativo para obtener un diseño más atractivo y simple. Sin embargo, en pantallas pequeñas abusar de espacios en blanco hace que los textos y las imágenes sean demasiado pequeñas, dificultando la legibilidad del contenido. Por tanto, planteo un nuevo diseño con márgenes más pequeños y textos que ocupan más espacio.



Mi primera idea era realizar diferentes diseños para la página principal en tablets y smartphones. Usando elementos flex y unidades de medida relativa he podido adaptar el mismo diseño para ambas pantallas, consiguiendo así una estética uniforme a través de los dispositivos.

Uno de los grandes retos de este proyecto ha sido conseguir una página totalmente responsive. El primer paso en mi proceso de desarrollo ha sido escribir en Visual Studio las reglas básicas, con unidades estimadas. A través de Chrome DevTools, realizando cambios en la hoja de estilo a tiempo real, he obtenido las medidas necesarias para el diseño

adaptativo. Tras experimentar en Chrome, regreso a Visual Studio con el planteamiento final.

3 . HTML .

Este código se compone principalmente por los elementos HEAD y BODY. Ambos agrupados en la etiqueta HTML. La HEAD, invisible para el usuario, contiene información acerca del documento: tipo de caracteres, tamaño de la pantalla, título, links a recursos externos, etc. Sólo el título puede verse en la pestaña del navegador. Aquí he añadido los links a las fuentes de Google Fonts, así como los íconos que uso en el footer para conectar a redes sociales de la empresa. Cómo se trata de una empresa ficticia, hay algunos hipervínculos que no conectan a ninguna página externa, pero he decidido añadirlos para posteriormente poder reutilizar partes de la web.

Dentro del body se encuentran seis bloques de contenido: la página principal que vemos al abrir la web, tres secciones que albergan información e imágenes relacionadas con la marca y un footer que sirve también como índice.

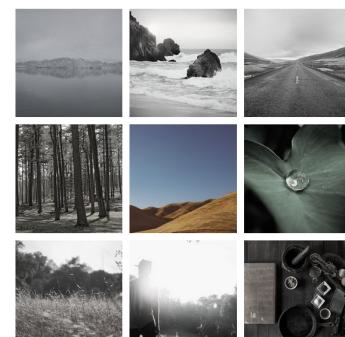
`<div id="openingPage">` Aunque normalmente situamos el header antes que cualquier bloque de contenido, para poder estructurar la vista inicial y que ocupe todo el alto de la pantalla, he incluido el header dentro de este div.

`<button class="abrir"> <button class="cerrar">` Estos botones sólo son visibles en pantallas pequeñas. El primer botón muestra tres líneas horizontales que representan un menú normalmente conocido como *hamburguesa*. Al hacer click en él, se despliega una *sidebar* que contiene el índice. El segundo botón aparece en esta *sidebar* y permite cerrarla.

`<div id="bannerTableView">` Para introducir una nueva imagen de fondo, qué sólo sea visible en tablets y smartphones, debo incluir este bloque en el cuerpo de mi documento. Aunque no alberga contenido, a través de CSS y los media queries podré estilizar un diseño diferente.

La primera sección tras la vista inicial (#aboutUs) contiene tres contenedores con tarjetas *fullscreen*. Dentro de cada una hay una imagen de stock que he sacado de *pexels.com*. Junto a la imagen, un texto descriptivo de la actividad de la empresa.

La segunda sección (#shop) es una galería de productos que podría expandirse posteriormente para obtener una tienda con todas sus funcionalidades. He usado como base una práctica realizada en clase. En este ejercicio desarrollamos una pequeña galería con nueve imágenes sobre las que se podía hacer click y



obtener una versión a tamaño completo de cada una de ellas. Esta vista se corresponde con el siguiente contenedor (*.modal*), qué solo es visible al hacer click en uno de los productos.

Lo más vendido

Pothos Golden



27.50€

[COMPRAR](#)

Calathea Compact



39.00€

[COMPRAR](#)

Monstera Adansonii



65.50€

[COMPRAR](#)

Chinese Evergreen



47.50€

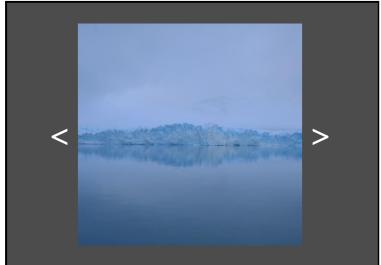
[COMPRAR](#)

Fittonia albivenis

Hoya Sweetheart

Umbrella Plant

Pencil Cactus



Utilizando el mismo concepto, una galería con botones que permiten desplazarse entre productos, creé una tarjeta con contenido adicional: título, texto descriptivo, precio y botón de compra. Conseguir que todo el texto de la nueva vista se correspondiera con cada producto ha sido uno de mis mayores problemas de este proyecto. Para esto he utilizado JavaScript.

Lo más vendido

Pothos Golden



27.50€

< >



65.50€

[COMPRAR](#)

Monstera Adansonii

Es una planta nativa con un hábito vertical único. Puede crecer hasta 25 pies de altura, pero a menudo se poda para mantenerla más pequeña. Las hojas tienen forma de corazón, con bordes festoneados y coloración verde brillante. Esta planta produce flores muy grandes que brotan de las axilas de las hojas en verano.

Fittonia albivenis

Hoya Sweetheart

Umbrella Plant

Pencil Cactus

La última sección es un formulario de contacto que usa la etiqueta *form* junto a *inputs* de texto. He agrupado cada *input* con una *label* explicativa a través de contenedores llamados "*inputPair*". Finalmente añado el *footer*, elemento que suele posicionarse al final de las páginas web con información sobre la empresa y enlaces de interés. Se divide en tres

contenedores, uno que alberga iconos de redes sociales, otro que contiene un índice y un pequeño texto sobre el copyright.

4. CSS .

Los estilos de la página vienen definidos en este archivo. Junto a algunas de estas reglas añado pequeños comentarios explicativos sobre el funcionamiento del código. Quizás lo más característico de esta parte del proyecto es el uso de la tecnología *flex*. Esta forma de estructurar el contenido me ha permitido llevar a cabo un diseño flexible y adaptativo. En pantallas grandes he distribuido el contenido en filas. Al elaborar los media queries en CSS he podido cambiar la distribución a columnas, acercándome a una vista apropiada para pantallas pequeñas de forma rápida. Afinando, entre otras, las medidas de *gap*, *margin* y *padding* obtengo un diseño totalmente responsive. También ha sido necesario cambiar el tamaño de muchas de las imágenes.

4. JavaScript .

A través de pequeños scripts añado dos funcionalidades interactivas a la web. La primera de ellas tiene que ver con la barra de navegación. En la vista de escritorio vemos el índice al completo en la *opening page*, pero al reducir las dimensiones de la pantalla incluir toda esta información es difícil. Para solucionar este problema incluyo un menú desplegable que se activa al hacer click en el ícono de “hamburguesa”.

Primero selecciono los botones que me permiten abrir y cerrar el menú, así como el contenido de la barra de navegación que quiero mostrar. Esto es posible almacenando en constantes dichos elementos, gracias a *querySelector* y los selectores de css.

```
const controls = document.querySelectorAll(".abrir, .cerrar");
const navegacion = document.querySelector("nav");
```

Una vez hago visible para JavaScript los botones, puedo pasárselos una función que me permite añadir o eliminar la clase *desplegado* al elemento nav. Combinando esta función con la propiedad *top* de css, consigo mostrar la *sidebar* cuando sea necesario.

```
controls.forEach(function(control){
    control.addEventListener("click", function(){
        navegacion.classList.toggle("desplegado");
    });
})
```

Utilizando una función muy parecida, puedo hacer que la barra de navegación desaparezca al hacer click en uno de los elementos del índice, para así centrar mi atención en el contenido y no tener que cerrar la sidebar manualmente con el botón de cerrar.

```
const links = document.querySelectorAll("nav a");

links.forEach(function(link) {
    link.addEventListener("click", function() {
        navegacion.classList.toggle("desplegado");
    });
})
```

Por otro lado tenemos el script que hace posible la vista completa de los productos en la sección de la tienda. Primero albergamos en constantes toda la información relativa al producto:

```
const miniaturas = document.querySelectorAll(".galeria a");
const modal = document.querySelector(".modal");
const modalCard = document.querySelector(".modalCard")
const imgModal = document.querySelector(".modal img");

const plantNames = document.querySelectorAll(".galeria h3")
const plantNameModal = document.querySelector(".modal h3");

const plantInfo = document.querySelectorAll(".galeria .info");
const plantInfoModal = document.querySelector(".modal .info");

const plantPrices = document.querySelectorAll(".galeria .price");
const plantPriceModal = document.querySelector(".modal .price");
```

Y luego guardo el contenido en *arrays* que me permiten moverme entre productos al hacer click en los botones de izquierda y derecha. Utilizando el índice de estos *arrays*, un nuevo *addEventListener* y condicionales que me permiten saber en qué punto de la galería me encuentro, consigo implementar esta funcionalidad.

`evento.stopPropagation();` Me permite anular la propagación adicional de los elementos para no cerrar la vista fullscreen al seleccionar el texto de esta.