



Diseño, desarrollo y optimización de una página web interactiva para una empresa de servicios tecnológicos

Memoria de Proyecto Final de Máster

Máster en Aplicaciones Multimedia

Itinerario profesional

Autor: Manuel Sánchez Moreno

Consultor: Laura Porta Simó

Profesor: Mikel Zorrilla Berasategui

7 de junio de 2019

Créditos/Copyright



Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada

[3.0 España de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

FICHA DEL TRABAJO FINAL

Título del trabajo:	Diseño, desarrollo y optimización de una página web interactiva para una empresa de servicios tecnológicos
Nombre del autor:	<i>Manuel Sánchez Moreno</i>
Nombre del consultor/a:	<i>Mikel Zorrilla Berasategui</i>
Nombre del PRA:	<i>Laura Porta Simó</i>
Fecha de entrega (mm/aaaa):	06/2019
Titulación::	<i>Máster en Aplicaciones Multimedia</i>
Área del Trabajo Final:	<i>TFM Itinerario profesional</i>
Idioma del trabajo:	<i>Castellano</i>
Palabras clave	Página web, Preprocesadores CSS, Experiencia de usuario
Resumen del Trabajo (máximo 250 palabras): <i>Con la finalidad, contexto de aplicación, metodología, resultados y conclusiones del trabajo.</i>	
<p>El crecimiento y democratización de Internet ha aumentado la demanda de sitios web por parte de empresas o particulares. Por ello, el presente proyecto recorre el proceso de programación, maquetación y optimización de una página web completa para la empresa Capybara Studio.</p> <p>Es intención, por tanto, el rediseño del aspecto del sitio web actual, modernizando su estética y proporcionando una web atractiva y animada que potencie la imagen de cara al público de la empresa, cobrando especial valor la creación de una nueva identidad e imagen propia soportada en gráficos vectoriales.</p> <p>Es un factor fundamental en la evolución del proyecto el análisis de usabilidad y experiencia de usuario previo a la realización de prototipos en base a los que se construye el producto final, que es alojado en un repositorio de GitHub para un control de versiones y sobre el que se trabaja en local y es desplegado, desde la línea de comandos, por Node JS.</p>	

Para llevar a cabo la maquetación responsive y creación de una estructura sólida del proyecto se ha hecho uso del framework *front-end* Bulma.io, favoreciendo a una correcta visualización independientemente del dispositivo usado por el usuario, y el uso de SCSS como preprocesador CSS para los estilos propios.

Abstract (in English, 250 words or less):

The growth and democratization of the Internet has brought an increase of the demand for websites from companies and individuals. For that reason, this project outlines the process of programming, layout and optimization of one full website for the company Capybara Studio.

The aim is then to redesign the visual aspect of the current site, modernizing its aesthetics and providing an attractive and animated website that favours the public image of the company, focusing on the generation of a new and unique corporate image based on vector graphics

The usability analysis and customer experience previous to the realization of prototypes used for the construction of the final product is considered a fundamental factor in the evolution of this project, which has been stored in a GitHub repository for version control; and for which work has been carried out in a local server environment, supported by Node JS technology.

The achievement of a corrective responsive layout has been founded on a solid, optimal and robust structure of the markup code due to the use of [Bulma.io](https://bulma.io) framework. This has led to correct visualization of the product irrespective of the device used by the customer. This correct personalised visualization has been enhanced and supplemented by the use of CSS preprocessors, in this particular case, SCSS

Agradecimientos

Al consultor, Mikel Zorrilla Berasategui, por sus sugerencias y correcciones del proyecto durante todo el proceso de desarrollo.

A la empresa, Capybara Studio, por la confianza depositada para el desempeño de la propuesta de una nueva imagen y presencia web.

Abstract

The growth and democratization of the Internet has brought an increase of the demand for websites from companies and individuals. For that reason, this project outlines the process of programming, layout and optimization of one full website for the company Capybara Studio.

The aim is then to redesign the visual aspect of the current site, modernizing its aesthetics and providing an attractive and animated website that favours the public image of the company, focusing on the generation of a new and unique corporate image based on vector graphics

The usability analysis and customer experience previous to the realization of prototypes used for the construction of the final product is considered a fundamental factor in the evolution of this project, which has been stored in a GitHub repository for version control; and for which work has been carried out in a local server environment, supported by Node JS technology.

The achievement of a corrective responsive layout has been founded on a solid, optimal and robust structure of the markup code due to the use of [Bulma.io](#) framework. This has led to correct visualization of the product irrespective of the device used by the customer. This correct personalised visualization has been enhanced and supplemented by the use of CSS preprocessors, in this particular case, SCSS

Keywords

Website, Html5, SASS, CSS preprocessor, Web Performance Optimization, Framework, User experience

Resumen

El crecimiento y democratización de Internet ha aumentado la demanda de sitios web por parte de empresas o particulares. Por ello, el presente proyecto recorre el proceso de programación, maquetación y optimización de una página web completa para la empresa Capybara Studio.

Es intención, por tanto, el rediseño del aspecto visual del sitio web actual, modernizando su estética y proporcionando una web atractiva y animada que potencie la imagen de cara al público de la empresa, cobrando especial valor la creación de una nueva identidad e imagen propia soportada en gráficos vectoriales.

Se considera un factor fundamental en la evolución del proyecto el análisis de usabilidad y experiencia de usuario previo a la realización de prototipos en base a los que se ha construido el producto final, que ha sido alojado en un repositorio de GitHub para un control de versiones y sobre el que se ha trabajado en un entorno con servidor local soportado por la tecnología Node JS.

La consecución de una correcta maquetación *responsive* se ha visto cimentada en una estructura sólida, óptima y robusta del código de marcado gracias al uso del framework Bulma.io, lo que ha dado lugar a una correcta visualización del producto independientemente del dispositivo usado por el usuario. Esta correcta visualización personalizada ha sido potenciada y suplementada por el uso de preprocesadores CSS, en el caso concreto que atañe al proyecto abordado, SCSS.

Palabras clave

Página web, Html5, SASS, Preprocesadores CSS, Framework, Rendimiento web, Experiencia de usuario

Índice

Capítulo 1: Introducción	12
1. Definición del proyecto.....	12
1.1 Resumen de la propuesta	12
1.2 Justificación y motivación	13
1.3 Descripción de la empresa y el sitio web actual	13
2. Descripción.....	15
2.1 Consideraciones iniciales	15
2.2 Diseño y estructura.....	15
3. Objetivos generales	16
3.1 Objetivo principal	16
3.2 Objetivos secundarios.....	16
3.3 Alcance.....	16
4. Metodología y proceso de trabajo	17
5. Planificación.....	18
6. Presupuesto	22
Capítulo 2: Análisis.....	23
1. Estado del arte	23
Capítulo 3: Diseño y desarrollo	26
1. Arquitectura de la página web.....	26
1.1 Diagrama de navegación.....	26
1.2 Arquitectura de la información	27
2. Diseño gráfico e interfaces	31
2.1 Estilos	31
2.2 Usabilidad/UX.....	34
2.3 Prototipado	35
3. Creación del entorno de trabajo.....	36
3.1 Homebrew.....	36
3.2 Repositorio GitHub	36

3.3 Instalación NODE JS	37
3.4 Node SASS.....	37
4. Lenguajes de programación y APIs utilizadas	38
4.1 HTML y PHP	38
4.2 Framework Bulma.io	38
4.3 Estilos CSS.....	38
4.4 JavaScript	39
4.5 Llamadas por CDN	39
4.6 XAMMP	40
4.7 Otras herramientas utilizadas	40
Capítulo 4: Implementación y despliegue	41
1. Requisitos de instalación.....	41
2. Instrucciones de despliegue.....	42
Capítulo 5: Conclusiones y líneas de futuro.....	44
1. Conclusiones	44
2. Líneas de futuro	46
Fuentes documentales	47
Referencias	47
Bibliografía.....	48
Anexos	50
Anexo A: Entregables del proyecto.....	50

Figuras y tablas

Índice de figuras

Figura 1: Diagrama de Gantt de la planificación inicial. Fuente: Instagantt	19
Figura 2: Diagrama de Gantt de la planificación final. Fuente: Instagantt.....	21
Figura 3: Diagrama de navegación de la página web	26
Figura 4: Diagrama del directorio	27
Figura 5: Logotipo positivo Capybara Studio. Versión antigua.	31
Figura 6: Logotipo positivo Capybara Studio. Versión nueva.	31
Figura 7: Paleta de colores página web Capybara Studio 2019.	32
Figura 8: Combinación de ambas tipografías en la página web.	33
Figura 9: Ilustraciones de Capybara Studio.	34
Figura 10: Ejemplo de icono y acento SVG.	34

Índice de tablas

Tabla 1: Tabla de tareas e hitos inicial	19
Tabla 2: Tabla de tareas e hitos final	20
Tabla 3: Presupuesto orientativo.....	22

Capítulo 1: Introducción

1. Definición del proyecto

1.1 Resumen de la propuesta

El presente Trabajo de fin de Máster consiste en el desarrollo de una página web *responsive*, optimizada e interactiva sin la utilización de CMS (Gestores de contenido) o plantillas para una empresa de servicios web, software y presencia digital, huyendo del concepto clásico de sitio web empresarial.

Se pretende cumplir con el objetivo de potenciar la imagen y la presencia de la empresa mostrando de manera más atractiva sus servicios o proyectos realizados. Contará con un diseño *frontend* a medida, con imagen e ilustración vectorial propia que apoyada en diferentes animaciones y una navegación lógica, intuitiva y rápida ofrecerá una agradable experiencia al usuario.

El desarrollo del proyecto seguirá un proceso secuencial por las etapas necesarias para la creación de una página web sin ayuda de gestores de contenido:

- **Definición de la arquitectura de la información y mapa web.**
- **Elaboración de prototipos basados en la experiencia de usuario y usabilidad:** Desde los primeros *wireframes* a mano alzada hasta un *sketch* de alta definición en el que se basa la construcción del sitio web.
- **Preparación de recursos gráficos optimizados**
- **Elección de tecnologías, lenguajes de programación y maquetación:** Análisis de las necesidades del proyecto en base a la cuáles se define la manera óptima de afrontarlas.
- **Creación de un entorno de trabajo en el que aplicar el desarrollo:** Consiste en un repositorio para control de versiones y un entorno de pruebas local.
- **Desarrollo del proyecto:** Puesta en marcha de las tecnologías decididas para conseguir un producto fiel al sketch definitivo previamente aprobado.
- **Optimización y rendimiento:** Esta fase comienza junto al desarrollo del proyecto, ya que es menester cuidar todos los aspectos que componen el proyecto durante el desarrollo, pese a llevar a cabo acciones finales que potencien los resultados.
- **Validación del código:** La utilización de validadores W3 para afinar y limpiar de posibles errores los ficheros CSS y Html, aspectos fundamentales si se busca lograr valores positivos de accesibilidad.

1.2 Justificación y motivación

Como toda novedad social, lo que en su momento fue algo muy exclusivo, hoy se ha democratizado hasta el punto de que, para un negocio, la aparición de internet ha supuesto una revolución en todos los ámbitos, en especial la comunicación, y la oportunidad de ganar visibilidad [1].

Esta revolución ha provocado que muchos negocios, empresas y particulares se lancen al mundo web, creando sitios y portales en la red con la finalidad de potenciar imagen y presencia en su entorno, pero esto no siempre es así. Del mismo modo que aumentan los usuarios de internet crece la competencia, ya no es suficiente con tener una página web informativa en la que aparezcan datos de contacto y alguna imagen, hay que buscar lo mejor, diferenciarse. Para lograr esto entran en juego muchas variables nuevas, desde el diseño, al rendimiento, a la accesibilidad, estrategias de marketing..., y olvidarse de alguna de ellas en el proceso de creación puede resultar perjudicial para la empresa.

Si se busca destacar en internet, no vale con ser uno más, se tiene que ofrecer algo diferente al usuario, hay que generar una impresión llamativa, de impacto, y fidelizar, distinguirse de la competencia. Esta diferenciación puede verse apoyada por la aplicación de nuevas tecnologías dotando los sitios web de mayor interactividad, cediendo al usuario más libertad de decisión una vez esté dentro del sitio y conseguir que la experiencia dentro del sitio web sea agradable y rápida, sin olvidar que en la mayoría de los casos el fin último de la empresa o propietario de la web busca transmitir una información o potenciar su negocio. Es en esta fusión de distinción, fidelización y eficacia donde radica la dificultad y el éxito de un buen proyecto web.

El desarrollo del trabajo en cuestión encuentra justificación como propio interés laboral, aprovechando el mismo para renovar la imagen y servicio web de una empresa cuya web actual se puede catalogar como obsoleta. La empresa, Capybara Studio, opera en el sector digital ofreciendo servicios de diseño gráfico, diseño web, desarrollo software, marketing digital y analítica web, y por motivos de plantilla y volumen de trabajo no ha podido dedicar el tiempo necesario a la construcción de un sitio web de calidad.

Se considera una oportunidad de potenciar y poner en práctica los conocimientos adquiridos durante el Máster de Aplicaciones Multimedia así como otras cualidades adquirida desempeñando actividades laborales.

1.3 Descripción de la empresa y el sitio web actual

Capybara Studio es un pequeño estudio madrileño de diseño gráfico y servicios digitales, formado por tres socios que tras años ofreciendo sus servicios como *freelance* en diferentes áreas apostaron por unir sus cualidades de cara al desarrollo de proyectos más potentes. Antiguos y nuevos clientes han supuesto un aumento del volumen de trabajo, provocando que se haya dejado de lado un valor

tan esencial, más en el sector en el que se mueve la empresa, como la página web propia (www.capybarastudio.com).

El sitio web actual se despliega sobre un Wordpress basado en una plantilla de pago sobre las que se han hecho pequeñas modificaciones y aplicado la identidad gráfica. El abandono de la misma provoca una carga lenta, falta de contenido y una accesibilidad mejorable.

2. Descripción

2.1 Consideraciones iniciales

- La creación de la página web apuesta por un desarrollo a medida, prescindiendo del uso de CMS (*Content Management System*).
- Creación de prototipos de alta definición utilizando un software de creación de *wireframes*, como Axure o Adobe XD. Se ha apostado por este último, con una curva de aprendizaje sencilla y compatible con todos los programas del paquete Adobe.
- Es un valor de vital importancia que el sitio a desarrollar sea *responsive*, es decir, adaptable a todos los tipos de pantalla ofreciendo una buena experiencia desde cualquier dispositivo.
- Se pretende trabajar en local, desde un repositorio Git clonado en el dispositivo de trabajo y sobre el que se aplicará la tecnología Node JS para la compilación de recursos y ejecución de un servidor local.
- Valoración de carga de recursos en local o por CDN en función del servidor y necesidades estimadas, como frameworks y librerías.
- Se busca la optimización de recursos y lenguajes buscando el mayor rendimiento posible del proyecto, superando los validadores necesarios. Se contempla el uso de gráficos SVG como formato de mayor calidad y óptimo para el proyecto.
- Uso de librerías y lenguaje JavaScript para animaciones y funcionalidades concretas.

2.2 Diseño y estructura

- Diseño a medida sin necesidad del uso de plantillas prediseñadas, supone un gasto mayor de tiempo, pero es el modo más efectivo de lograr una réplica limpia del boceto validado.
- El aspecto de la web debe transmitir valores acordes a la empresa tales como cercanía, alegría y creatividad.
- Establecimiento de unas líneas de estilo acorde a la identidad gráfica de la empresa y recursos gráficos a utilizar.
- Utilización del *framework Bulma*, con una curva de aprendizaje sencilla, para simplificar y generar una estructura *Html* robusta y *responsive*.
- Generar archivos CSS escritos en SCSS y preprocesados mediante comandos de terminal por dependencias instaladas en el proyecto.

3. Objetivos generales

Los objetivos del presente proyecto han sido establecidos en base a las peticiones y necesidades de los socios que conforman la empresa Capybara Studio.

3.1 Objetivo principal

Desarrollo y diseño de una página web optimizada, animada e interactiva, para la empresa de servicios web Capybara Studio con el fin de mejorar la actual y potenciar su imagen.

3.2 Objetivos secundarios

- Desarrollar un diseño propio, que siga los principios de *mobile-first* y cumpla las tendencias de diseño presentes para el curso 2019/2020.
- Ofrecer a los usuarios la mejor experiencia posible siendo la web accesible, siguiendo unos principios básicos de usabilidad y una arquitectura de la información lógica
- Utilización de *Node JS* para creación de un entorno de trabajo con servidor local basado en un directorio de GitHub y para el preprocesado de archivos CSS.
- Estructurar el sitio web para futuras implementaciones de servicios y estrategias SEO.
- Llevar el control de versiones en GitHub para lograr un proyecto escalable.
- Poner en práctica de manera transversal conocimientos adquiridos a lo largo del Máster en Aplicaciones Multimedia de la UOC.

3.3 Alcance

El proyecto contempla cinco entregas:

- **Propuesta formal del proyecto**
- **Mandato del proyecto**
- **Primera entrega del desarrollo:** Prototipos, configuración del entorno de trabajo, estructura de ficheros y carpetas y preparación del material multimedia identificado.
- **Segunda entrega:** Construcción de animaciones y ficheros HTML, CSS y JS necesarios dando forma a la web.
- **Cierre:** Optimización del proyecto, elaboración de documentación y memoria.

4. Metodología y proceso de trabajo

La empresa cuenta con un sitio web desactualizado lo que ofrece la oportunidad de una refactorización de la ya existente, pero como ya se ha comentado en el documento, esta página no es atractiva ni está optimizada, por ello se apuesta por un nuevo sitio web a medida.

Esta decisión está fundamentada en que, al buscar un desarrollo muy específico, es más productiva la inversión de horas en su totalidad a un nuevo proyecto que a analizar la opción de mejorar lo que ya existe, teniendo en cuenta que los pilares sobre los que se asienta no son muy sólidos.

En este punto del proyecto surgen opciones que valorar:

- Se debe realizar una web totalmente accesible desde cualquier dispositivo, por tanto el diseño *responsive* es una condición imprescindible para la consecución de los objetivos.
- Usar o prescindir de un CMS. Se ha optado por prescindir del mismo dada las necesidades de la web, búsqueda de velocidad y diseño muy personalizado. Sin descartar una utilización de CMS como *backend* en aplicaciones futuras. El deseo de aplicar conocimientos en Html, Css y JS ha sido otro detonante, en cierto modo, de esta decisión.

El proceso de trabajo seguido ha consistido en el desglose de tareas y subtareas en función de una planificación, que han sido rigurosa y meticulosamente desglosadas en la plataforma *Asana*.

Para la consecución de cada tarea ha sido importante la gestión de las subtareas con un panel *kanban* mediante un tablón manual y *postits* con cada subtask que avanzan según el estado de desarrollo en el que se encontrara.

5. Planificación

Desglosadas las tareas en la plataforma Asana, se hace uso del software *instagantt* para asignar fechas y generar un diagrama de Gantt que ha ido variando a lo largo del proceso, debido a que han aparecido trabas y complicaciones causando variaciones importantes en la planificación inicial del proyecto.

Es un factor importante a la hora de planificar un proyecto estimar adecuadamente las horas disponibles y recursos a los que el personal o equipo de producción (en el caso del proyecto que hace referencia esta memoria un único trabajador) para evitar errores notables como los surgidos en este desarrollo. Conviene dedicar más tiempo a esta fase para cubrir cualquier imprevisto siempre estimando con datos objetivos y realistas.

Es importante que la planificación incluya como hitos parciales las entregas de todas las PEC del itinerario de la asignatura.

Planificación inicial

TAREA	DURACIÓN	INICIO	FIN
PEC 1 – Propuesta formal del proyecto	11 días	25/02/2019	08/03/2019
Propuesta formal del proyecto	11 días	25/02/2019	08/03/2019
Entrega PEC 1	0 días	08/03/2019	08/03/2019
PEC 2 – Mandato del proyecto	11 días	09/03/2019	20/03/2019
Mandato del proyecto	8 días	12/03/2019	20/03/2019
Entrega PEC 2	0 días	20/03/2019	20/03/2019
PEC 3 – Entrega 1	25 días	21/03/2019	15/04/2019
Definición, estructura y mapa web	6 días	21/03/2019	26/03/2019
Diseño sketches	13 días	27/03/2019	08/04/2019
Definición del sketch definitivo	0 días	08/04/2019	08/04/2019
Creación de espacio de trabajo	1 día	09/04/2019	09/04/2019
<ul style="list-style-type: none"> • Estructura de directorios GitHub 	1 día	09/04/2019	09/04/2019
<ul style="list-style-type: none"> • Clonar proyecto GitHub 	1 día	09/04/2019	09/04/2019
<ul style="list-style-type: none"> • Instalar Node JS 	1 día	09/04/2019	09/04/2019
<ul style="list-style-type: none"> • Creación server local basado en Node JS 	1 día	09/04/2019	09/04/2019
Preparación de materiales multimedia	6 días	10/04/2019	15/04/2019
Entrega PEC 3	0 días	15/04/2019	15/04/2019
PEC 4 – Entrega 2	28 días	16/04/2019	13/05/2019
Creación de la estructura HTML	27 días	16/04/2019	12/05/2019
Creación CSS	27 días	16/04/2019	12/05/2019

• Escribir código SCSS y preprocesarlo	27 días	16/04/2019	12/05/2019
Creación JS	27 días	16/04/2019	12/05/2019
Creación de animaciones	27 días	16/04/2019	12/05/2019
Entrega PEC 4	0 días	13/05/2019	13/05/2019
PEC 5 - Cierre	24 días	14/05/2019	07/06/2019
Optimizar proyecto web	4 días	14/05/2019	17/05/2019
Maquetar la memoria del proyecto	11 días	19/05/2019	29/05/2019
Elaborar presentación del proyecto	9 días	29/05/2019	06/06/2019
Entrega final	0 días	07/06/2019	07/06/2019
TRABAJO TOTAL	99 días	25/02/2019	07/06/2019

Tabla 1: Tabla de tareas e hitos inicial

Diagrama de Gantt inicial

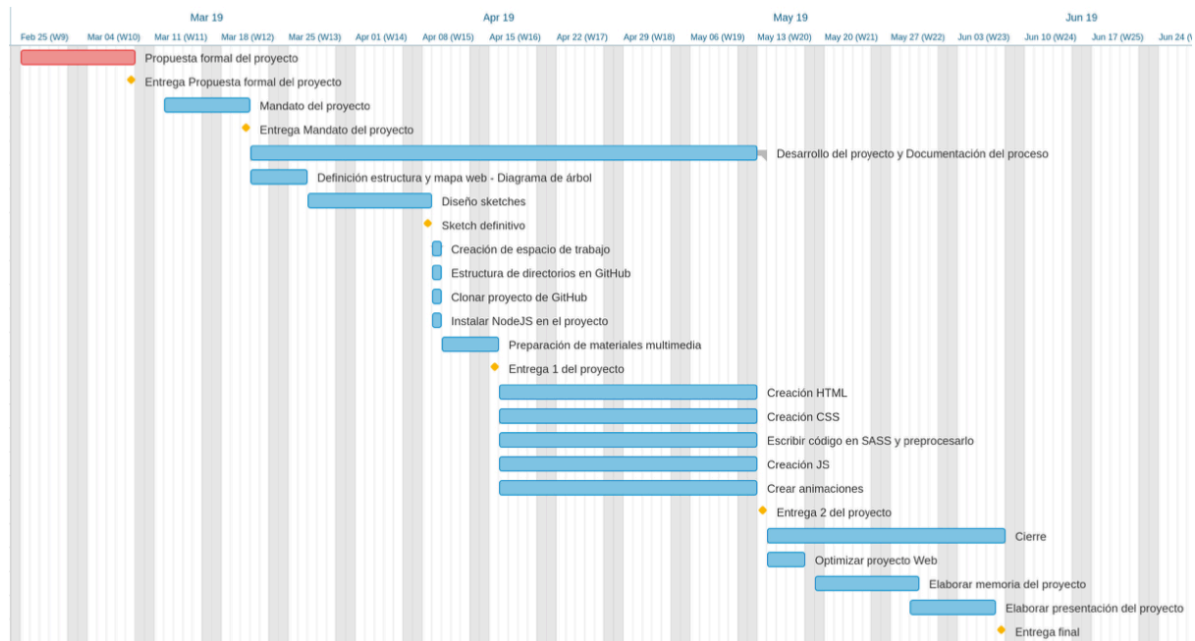


Figura 1: Diagrama de Gantt de la planificación inicial. Fuente: Instagantt

Planificación final

TAREA	DURACIÓN	INICIO	FIN
PEC 1 – Propuesta formal del proyecto	11 días	25/02/2019	08/03/2019
Propuesta formal del proyecto	11 días	25/02/2019	08/03/2019
Entrega PEC 1	0 días	08/03/2019	08/03/2019
PEC 2 – Mandato del proyecto	11 días	09/03/2019	20/03/2019
Mandato del proyecto	8 días	12/03/2019	20/03/2019
Entrega PEC 2	0 días	20/03/2019	20/03/2019
PEC 3 – Entrega 1	25 días	21/03/2019	15/04/2019
Definición, estructura y mapa web	6 días	21/03/2019	26/03/2019
Diseño sketches	13 días	27/03/2019	14/04/2019
Definición del sketch definitivo	0 días	14/04/2019	14/04/2019
Preparación de materiales multimedia	15 días	04/04/2019	18/04/2019
Entrega PEC 3	0 días	15/04/2019	15/04/2019
PEC 4 – Entrega 2	28 días	16/04/2019	13/05/2019
Creación de espacio de trabajo	1 día	22/04/2019	25/04/2019
• Estructura de directorios GitHub	3 día	23/04/2019	25/04/2019
• Clonar proyecto GitHub	1 día	23/04/2019	23/04/2019
• Instalar Node JS	1 día	24/04/2019	24/04/2019
• Creación server local basado en Node JS	2 día	24/04/2019	25/04/2019
Creación de la estructura HTML	22 días	25/04/2019	16/05/2019
Creación CSS	27 días	26/04/2019	23/05/2019
• Escribir código SCSS y preprocesarlo	27 días	26/04/2019	23/05/2019
Creación JS	27 días	26/04/2019	23/05/2019
Creación fuente propia con iconos	1 días	10/05/2019	11/05/2019
Entrega PEC 4	0 días	13/05/2019	13/05/2019
PEC 5 - Cierre	24 días	14/05/2019	07/06/2019
Creación de animaciones de entrada	7 días	23/05/2019	29/05/2019
Optimizar proyecto web	3 días	29/05/2019	31/05/2019
Maquetar la memoria del proyecto	8 días	30/05/2019	06/06/2019
Elaborar presentación del proyecto	4 días	03/06/2019	06/06/2019
• Vídeo	2 días	03/06/2019	04/06/2019
• Powerpoint	2 días	04/06/2019	06/05/2019
Entrega final	0 días	07/06/2019	07/06/2019
TRABAJO TOTAL	99 días	25/02/2019	07/06/2019

Tabla 2: Tabla de tareas e hitos final

Diagrama de Gantt inicial

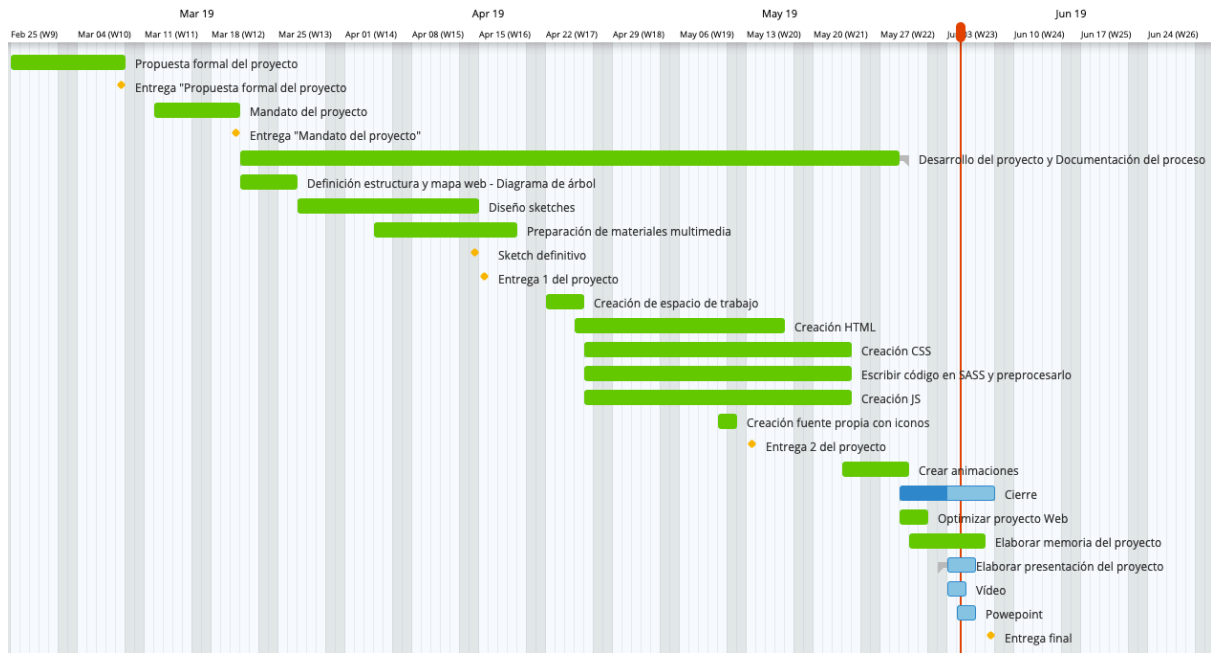


Figura 2: Diagrama de Gantt de la planificación final. Fuente: Instagantt

6. Presupuesto

El proyecto contempla el desarrollo principal de la estructura de una nueva página web para la empresa Capybara Studio. No se puede considerar completo ya que se precisa de la validación por parte de los socios de la empresa del contenido gráfico y funcionalidad para abordar su despliegue de cara al público soportado en tecnologías que gestionen en backend. Asumidas estas premisas se ha realizado una estimación del coste que hubiera supuesto el proyecto.

El gasto ha sido nulo, todo software y recursos son libres, gratuitos o bien propios, por tanto el cálculo del presupuesto estimado se basa en una aproximación de las horas invertidas:

CONCEPTO	HORAS	€/H	TOTAL €
Curva de aprendizaje de diferentes recursos	15	0	0
Planificación	15	30	450
Diseño a medida y elaboración de prototipos	40	40	1600
Producción y optimización de recursos gráficos	25	40	1000
Creación de entorno de trabajo	8	30	240
Programación HTML, CSS y JS	60	40	2400
Validadores y optimización	5	40	200
TOTAL			5800€ IVA NO INCLUIDO

Tabla 3: Presupuesto orientativo

Capítulo 2: Análisis

1. Estado del arte

El incremento de la demanda y uso de nuevas tecnologías, con internet a la cabeza, hace del diseño y desarrollo de páginas web como uno de los servicios más necesarios en el mercado. [2] Esta situación ha promovido la aparición de herramientas y software gratuitos que, apoyados de la enorme cantidad de recursos disponibles en abierto en la red (vídeos, tutoriales, documentaciones...), permiten a cualquier usuario intentar desarrollar sitios web y, con paciencia, conseguirlo.

Hoy día el lenguaje de marcado HTML5 y el lenguaje de estilos CSS3, junto con las funcionalidades que aportan PHP y *Javascript*, son los más extendidos en el panorama del diseño y desarrollo web. Sin ir más lejos, en una combinación de ellos están basados los principales gestores de contenido (*Content Management System*), tales como *Wordpress* o *Joomla*, cuya aparición ha trivializado el proceso de generar un sitio web.

Del mismo modo han aparecido ayudas para diseñadores y desarrolladores en forma de *frameworks* como *Bootstrap* o *Bulma*, facilitando la creación de páginas adaptables y de estilos vistosos sin necesidad de invertir gran cantidad de horas de trabajo y esfuerzo. Han supuesto un gran avance en cuanto a optimización de tiempo en proyectos, pudiendo generar estructuras responsivas y muy robustas con simples clases predefinidas.

La evolución de los lenguajes de programación y maquetación, junto a la aparición de *frameworks* y CMS, como apoyo para diseñadores y desarrolladores, obliga a los mismos a valorar y fundamentar la elección para el desarrollo de un proyecto. Existen varios entornos de trabajo y gestores de proyectos colaborativos, como GitHub, los que también ocupan un lugar importante en el proceso de construcción de una web.

Al igual que aumentan las herramientas y evolucionan los lenguajes y tecnologías han aparecido conceptos que se deben valorar, trabajar y cumplir si buscamos que el sitio web a desarrollar sea reconocido y competitivo en Internet.

La consecuencia de estas facilidades es la creación masiva de nuevos sitios web y una competencia elevada. Por tanto, si se busca destacar en el entorno web hay que ofrecer algo atractivo al potencial usuario que se diferencie de las plantillas y diseños estándar que ofrecen estas plataformas.

Las [3] tendencias actuales del diseño web apuestan por textos cortos y grandes, huyendo de las fuentes pequeñas y difíciles de leer; recursos gráficos con personalidad propia que huya de bancos de imágenes; tipografías manuscritas; utilización de animaciones y la ruptura de lo estándar mediante exposición de color, con diseños diferentes y dando mayor libertad de interacción al usuario. Se apuesta por una fuga de elementos intrusivos e intentar evitar el uso de menús hamburguesas.

Pese a lo volátil del sector y que las tendencias varían anualmente hay [4] varios principios que se mantendrán perennes: el *mobile-first*, todo pensado para el usuario móvil, dispositivo con más acceso a internet en la actualidad; diseños *one-page* no necesariamente de *scroll* vertical irrumpiendo con fuerza el *scroll* horizontal; y la velocidad de carga como premisa básica de cualquier web, cuánto más rápido mejor.

Teniendo en cuenta el sector en el que se enmarca el proyecto: las tecnologías y servicios de diseño y desarrollo web; esta situación es aún más acusada, una competencia elevada y creativa, y se debe tener en cuenta que indirectamente se muestra al potencial cliente cualidades de los servicios que se ofrecen.

Para destacar, a día de hoy, diseñador y desarrollador deben trabajar pensando, en primer lugar, en el usuario. [5] Yusef Hassan, en su artículo “Experiencia de usuario: Principios y métodos” define la experiencia de usuario como la necesidad de los profesionales de hacer la tecnología amigable, satisfactoria y fácil de usar, resolviendo las necesidades del visitante de manera eficiente y fluida. El crecimiento y valor de la experiencia de usuario (UX) ha propiciado la aparición de profesionales especializados en este aspecto del diseño web a día de hoy.

Son muchas cuestiones las que se deben abordar previo al desarrollo de nuestro sitio web y que se contemplan bajo los estándares de la experiencia del usuario, siempre partiendo de la máxima de diseños pensados para todo aquel que interactúe con la web, ya que dentro de este concepto encontramos fases fundamentales en el diseño y desarrollo web tales como la usabilidad, la arquitectura de la información o la accesibilidad.

La usabilidad hace referencia a la calidad de un producto según lo fácil que sea de usar, y se basa en los principios de la [6] eficacia, eficiencia, la facilidad de aprendizaje y de ser recordado. Un ejemplo básico de usabilidad se puede encontrar en la necesidad de que cualquier sitio web, a día de hoy, sea *responsive*, es decir, que pueda ser utilizado de manera sencilla en cualquier dispositivo.

La arquitectura de la información en el contexto web [7] contempla las necesidades y características de los usuarios, su entorno, y es de inestimable ayuda a la hora de definir la organización de la información y la manera de que el potencial usuario interactúa con ella. Su correcta aplicación, olvidada en muchas situaciones, aporta un valor esencial en la calidad de una página web.

[2] Actualmente existe un gran interés en el cumplimiento de unas pautas de calidad en el desarrollo web que impulsen y fomenten la inclusión digital independientemente de las capacidades de los usuarios. Este concepto de accesibilidad, uno de los pilares de la experiencia de usuario, se entiende como el acceso a la información de la web sin limitaciones por cuestión de minusvalía o deficiencia. [8] Para la mejora de este valor en la web existen los estándares de la *Web Content Accessibility Guidelines* (WCAG) creada por W3C, que explica cómo hacer el contenido más accesible, pasando por el código, imágenes y sonido.

Existen otras cuestiones que se deben tener en cuenta en el diseño web: el rendimiento, y más en concreto la velocidad. Este aspecto es lo que más influye, en un primer momento, en la experiencia del usuario. Vivimos en una sociedad que demanda una tecnología instantánea, y la demora de segundos puede suponer la pérdida de un cliente potencial para un sitio web, cada milisegundo cuenta. [9].

Este proceso, al igual que la accesibilidad, aunque se planifique en un primer paso, se irá poniendo en marcha durante el desarrollo del proyecto web. En él influye la optimización de recursos y una correcta estructura y limpieza del código, así como cuidar la cantidad de contenido a usar.

El fin de todas las páginas es ser visitadas, atraer al público y lograr su objetivo, ya sea vender, publicitar o entretener, y se busca realizarlo de la manera más atractiva y rápida posible. Por tanto, el interés de aparecer bien posicionados en los buscadores es algo común, y detrás de ello hay un proceso y unas estrategias estudiadas que ayudan al posicionamiento bajo la rama de la analítica web y el posicionamiento web. Google utiliza numerosos algoritmos que analizan y parametrizan los sitios web, y en estas situaciones es donde entra en juego el profesional que apuesta por unas u otras estrategias, pero si hay algo común en las valoraciones del buscador es que se penalizará la ausencia o mala ejecución de las tendencias y conceptos mencionados anteriormente.

Capítulo 3: Diseño y desarrollo

1. Arquitectura de la página web

1.1 Diagrama de navegación

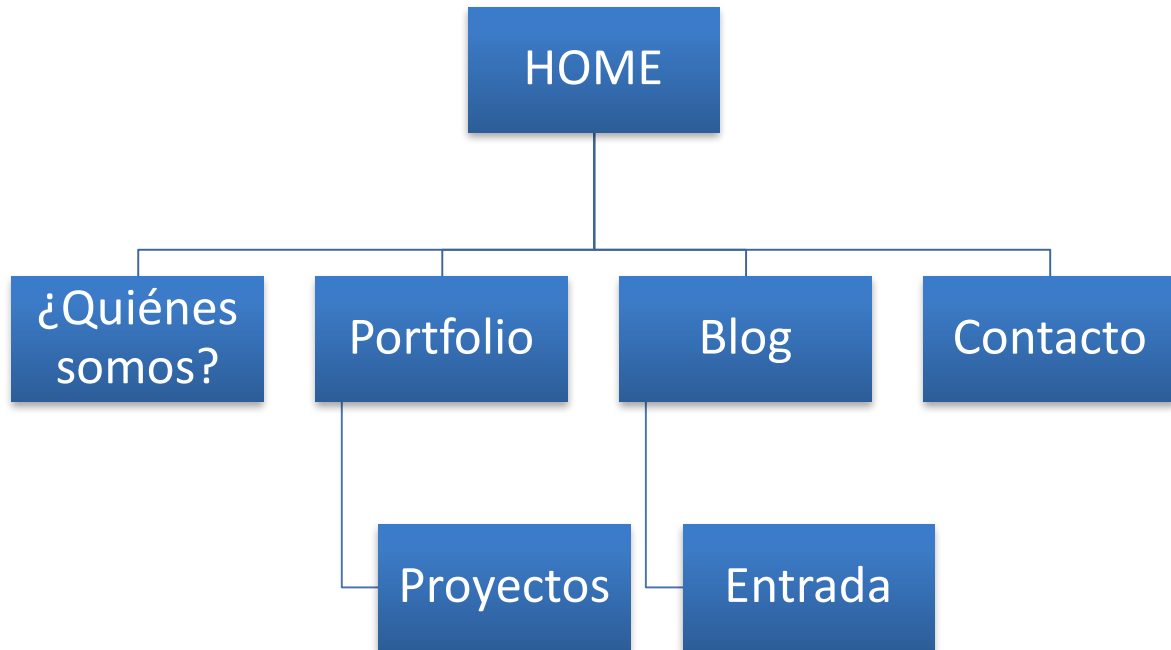


Figura 3: Diagrama de navegación de la página web

Posibles nuevas secciones y páginas secundarias, tales como políticas o aviso legal, deben ser consensuadas por el equipo director de la empresa y se procederá a su ejecución una vez se hayan decidido.

La navegación es simple, sencilla y lineal, con posibilidad de acceder a cualquier otro sitio de la web independientemente de dónde se encuentre el usuario.

1.2 Arquitectura de la información

El directorio del proyecto se estructura del siguiente modo:

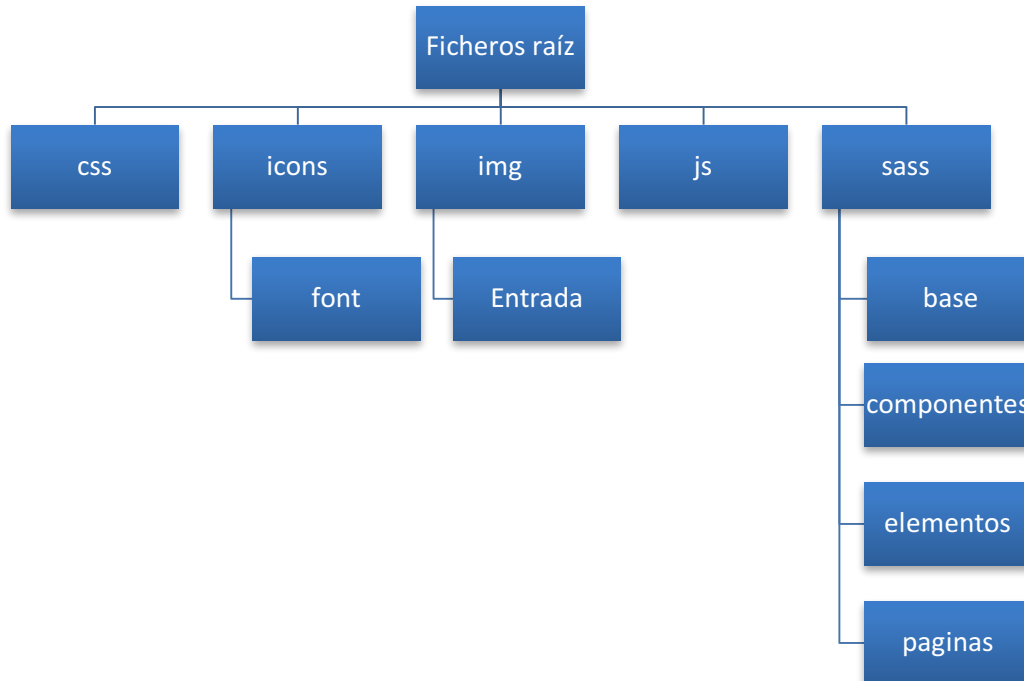


Figura 4: Diagrama del directorio

Gracias al archivo `.gitignore` se evita que se cargue en el directorio la carpeta `node_modules` que supone una enorme cantidad de archivos que actúan como dependencias no utilizadas en la mayoría de los casos.

El proyecto web de Capybara Studio se compone de los siguientes archivos:

Ficheros raíz

- **`index.php`** → Página de inicio del sitio web.
- **`about-us.php`** → Página que contendrá la información relevante de la empresa, así como los servicios que ofrece.
- **`portfolio.php`** → Espacio de la web que mostrará, con un sistema de filtrado y por orden de realización, los proyectos que desarrolle la empresa.
- **`blog.php`** → Sección para la subida gradual de noticias y post.
- **`contacto.php`** → Se incluyen diferentes formas de contacto, desde un formulario corriente a uno guiado secuencial.

- **header.php** → Contiene el *header* y los menús de navegación comunes a todas las páginas del sitio. El fichero será llamado mediante la propiedad *include*.
- **footer.php** → Fichero que contendrá el *footer* común a todas las páginas. . El fichero será llamado mediante la propiedad *include*.
- **package.json** → Archivo que registra datos del proyecto tales como dependencias instaladas.
- **manifest.json** → Archivo que contiene información y datos de la web, en vista a poder convertirse en *webapp* en un futuro.
- **README.md** → Contiene un breve manual de despliegue y ejecución del proyecto.
- **LICENSE** → Contiene la licencia definida para el proyecto.

CSS

- **main.css** → Hoja de estilos preprocesada.
- **main.css.map** → Hoja de mapa css generada al realizar la compilación con el preprocesador.
- **normalize.css** → Hoja de estilos estándar que unifica elementos y secciones comunes.

Iconos (Icons)

- **capbara-studio-icons.css** → Hoja de estilos de la tipografía icónica desarrollada.
- **config.json** → Ejecutable a través de terminal que da lugar a los recursos necesarios para la tipografía de iconos propias, en este caso a a través de Fontello.
- **Font** → Carga de la tipografía en diferentes formatos (.eot, .svg, .ttf, .woff y .woff2).

Imágenes (Img)

Directorio compuesto por todos los recursos gráficos utilizados en el sitio web.

Form

- **actions.php** → Fichero encargado de preparar la funcionalidad definitiva que requiere el formulario guiado de la sección contacto.

JavaScript (JS)

- **aos.js** → Librería encargada de ejecutar animaciones de entrada.

- **filter.js** → Script encargado del filtrado de ítems en el portfolio.
- **guided-form.js** → Funcionalidad secuencial al formulario guiado de la sección contacto.
- **main.js** → Recursos principales utilizados en la web.
- **mobile.js** → Script encargado del despliegue del menú para dispositivos móviles.

SASS

Estructura modular para una mejor organización de los estilos:

- **main.scss** → Hoja de estilos que, mediante etiquetas *import*, recoge todo el código que existe en las carpetas y estructura que forman la web, para posteriormente compilarlo.
- **Base**
 - **_all.scss** → Importa las hojas de estilos SCSS de la carpeta.
 - **cs_genericos.scss** → Estilos propios aportados a recursos generales, como tipografías o tamaños.
 - **cs_spacing.scss** → Recursos espaciadores, basados en atomic design, hechos a medida.
 - **cs_variables.scss** → Hoja de variables estilísticas personalizadas para el proyecto.
- **Componentes**
 - **_all.scss** → Importa las hojas de estilos SCSS de la carpeta.
 - **cs_animations.scss** → *Keyframes* y animaciones personalizadas.
 - **cs_footer.scss** → Estilos del pie de página.
 - **cs_guided-form.scss** → Estilos del formulario guiado.
 - **cs_nav.scss** → Estilos del menú de navegación.
- **Elementos**
 - **_all.scss** → Importa las hojas de estilos SCSS de la carpeta.
 - **cs_botones.scss** → Estilos de los botones.

- **Páginas**

- **_all.scss** → Importa las hojas de estilos SCSS de la carpeta.
- **cs_about.scss** → Estilos página “¿Quiénes somos?”.
- **cs_blog.scss** → Estilos página “Blog”.
- **cs_contacto.scss** → Estilos página “Contacto”.
- **cs_home.scss** → Estilos página “Home”.
- **cs_portfolio.scss** → Estilos página “Portfolio”.

2. Diseño gráfico e interfaces

2.1 Estilos

La paleta de colores y guías de estilo de Capybara Studio previas al proyecto se veían muy marcadas por la combinación de colores azules, grises y blancos, líneas y tipografías rectas, buscando transmitir elegancia, orden y seriedad, contrastando con el carácter desenfadado del blog. La falta de contenido novedoso y la nula actualización de secciones ha provocado una obsolescencia considerable en cuanto a apariencia.

La decisión del desarrollo de un nuevo sitio web ha evolucionado en paralelo al establecimiento de una nueva identidad, más fresca y cercana buscando empatizar en mayor medida con los usuarios, sin olvidar los colores que han sido seña de identidad de la empresa desde que comenzó su andadura. Todo ello, en base a tendencias de diseño que están en boga y supuestamente mandarían durante el curso 2019/2020.

La demanda de inmediatez, brevedad e información concisa por parte de los usuarios actuales ha sido un factor relevante en la elaboración de un diseño con menos texto, más icónico e ilustrativo.

Logotipo

Se ha realizado una pequeña adaptación del logotipo, cambiando la tipografía acorde a la que será la seña de identidad de la empresa.

Vértices suaves y redondeados para transmitir confianza y cercanía, apoyado en trazos finos que aporten elegancia y calidad, siempre respetando los colores corporativos principales.



The image shows the old logo for Capybara Studio. The word 'CAPYBARA' is written in a bold, uppercase, sans-serif font. Below it, the word 'studio' is written in a lowercase, cursive script font in a teal color.

Figura 5: Logotipo positivo Capybara Studio. Versión antigua.



The image shows the new logo for Capybara Studio. The word 'capybara' is written in a lowercase, rounded, sans-serif font. Below it, the word 'studio' is written in a lowercase, rounded, sans-serif font in a teal color.

Figura 6: Logotipo positivo Capybara Studio. Versión nueva.

Paleta de colores

La apuesta de color que se utiliza en el proyecto de Capybara Studio gira en torno a una de las tendencias principales del curso actual en el diseño web: la explosión de color huyendo de los tonos apagados mediante el uso de colores vivos para llamar la atención del usuario sobre determinadas partes del sitio web[10].

Todo apoyado en fondos blancos, limpios y elegantes que favorecen la legibilidad, prescindiendo de colores agresivos como el negro puro.

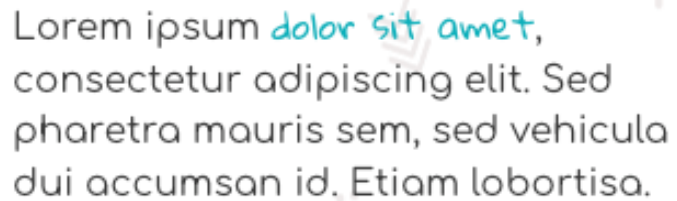


Figura 7: Paleta de colores página web Capybara Studio 2019.

Paleta tipográfica

Una forma de diferenciación importante es la utilización de tipografías llamativas, salirse de la paleta común. Valorada la situación se ha optado por el uso y combinación de dos tipografías muy diferentes:

- **Comfoorta:** De palo seco y sin serifa. Con bordes suaves y varios cuerpos de letra que dan libertad en cuanto a combinaciones.
- **Gloria Hallelujah:** De estilo manuscrito. Marca la diferencia, pero hay que aplicarla con mesura, huyendo de párrafos largos o elementos que pueda entorpecer la legibilidad.



Lorem ipsum dolor sit amet, consectetur adipiscing elit. Sed pharetra mauris sem, sed vehicula dui accumsan id. Etiam lobortisa.

Figura 8: Combinación de ambas tipografías en la página web.

Los tamaños del texto, desde títulos a párrafos gozan de tamaños fácilmente legibles que se calculan de forma dinámica en función del tamaño del dispositivo gracias a funciones que se han utilizado en el código SCSS.

Ilustraciones, iconos y elementos gráficos

En un entorno web con inmensidad de bancos de imágenes, gratuitos o de pago, desde los que adquirir y hacer uso de recursos libres de derechos cobra especial importancia la diferenciación de otras web mediante la producción propia, convirtiéndose en un valor diferencial muy a tener en cuenta según las principales tendencias de diseño.

Se ha invertido mucho tiempo en lo que se ha considerado una cuestión clave que influya para generar atracción e impacto en los usuarios, y acorde al uso de tipografía manuscrita como detalle diferencial, se apuesta por la elaboración de ilustraciones propias de carácter animado.

Esbozadas sobre papel, posteriormente han sido vectorizadas y customizadas por el software de tratamiento vectorial *Adobe Illustrator*, dando lugar a un resultado de óptima visualización y con un carácter desenfadado.



Figura 9: Ilustraciones de Capybara Studio.

Del mismo modo han sido diseñados los iconos, apostando por el trazo irregular a mano alzada generando una tipografía y hoja de estilos propia para utilizarlos en la web

El estilo, muy marcado, se ve reforzado por la utilización de trazos y formas irregulares, simulando brochazos aleatorios de pinceles con colores llamativos para acentuar y resaltar partes concretas.

Huyendo de los clásicos formatos de imagen, y apostando por la calidad y alta resolución en cualquier dispositivo se han exportado para web archivos en formato SVG, manteniendo su condición de vector.



Figura 10: Ejemplo de icono y acento SVG.

2.2 Usabilidad/UX

La búsqueda de un proyecto con niveles positivos en cuanto a usabilidad y experiencia de usuario ha supuesto uno de los aspectos más conflictivos en los comienzos del proyecto.

La idea inicial consistía en una página web con navegación y *scroll* horizontal buscando diferenciación, si bien, tras realizar un test de navegación sobre cinco personas del perfil usuario medio en un sitio web con dicha característica y unos esbozos sobre papel la idea se desechó.

El resultado fue que ninguno de los usuarios testeados fue capaz de comprender la lógica de navegación de dichas páginas, tratando de desplazarse de manera vertical e invirtiendo demasiado tiempo en navegar por la misma. Es por tanto que, pese a lo llamativo que resultaba aplacar esta propiedad, se busca hacer llegar una información, y para que el efecto sea positivo hay que buscar rapidez y eficiencia en esta labor.

Para dicho test se ha hecho uso de la web <https://www.anticafe.eu/>.

Definida la navegación vertical, se ha apostado por la presencia de modos de navegación entre páginas desde cualquier lugar de la web, gracias a la presencia de un *sticky menu* y un *footer* con navegación.

Otro punto importante ha sido el menú móvil. Si tenemos en cuenta lo monótono de los *menús de hamburguesa* se ha tomado como referencia la solución que han adoptado aplicaciones punteras como *Whatsapp* o *Telegram*. Un icono fijo, fácil de interactuar, en la esquina inferior derecha, que despliega el contenido de navegación.



Figura 11: Usabilidad menú móvil.

2.3 Prototipado

Sobre la base de los objetivos y principios de usabilidad a seguir en el proyecto se definieron múltiples ideas, esquemas y esbozos sobre papel que han asumido la función de *wireframes* o prototipos de bajo nivel, lo que se validó como prototipos de alta fidelidad.

Han existido dudas acerca del software a utilizar entre *Axure* y *Adobe XD*, si bien la familiaridad del autor con los programas del paquete Adobe (curva de aprendizaje asequible por ende) y la gratuidad de este han inclinado la balanza a su favor.

El proceso se ha regido por los principios de *mobile-first*, comenzando por tanto con la versión Smartphone en adelante, obteniendo un resultado satisfactorio.

3. Creación del entorno de trabajo

Una vez la producción gráfica ha sido finalizada y validados los prototipos definitivos se ha comenzado con la creación de un espacio de trabajo soportado por las siguientes tecnologías, a través de comandos de línea de la terminal del dispositivo.

3.1 Homebrew

Al utilizar un Macbook Pro como herramienta es recomendable la instalación de un gestor de paquetes como Homebrew facilitando notablemente el proceso de configuración de los demás componentes.

```
/usr/bin/ruby -e "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

Ejecutado este código en la terminal el propio programa marcará los pasos a seguir para su correcta instalación.

[Enlace a la documentación](#)

3.2 Repositorio GitHub

Se ha escogido GitHub como repositorio en torno al que desarrollar el proyecto, de este modo es posible llevar un control de versiones, actualizar cambios y poder trabajar desde diferentes dispositivos (esta situación no ha sido necesaria al realizar el proyecto desde el mismo dispositivo).

Una vez dado de alta, mediante comandos en línea de la terminal, se instala *git* en el dispositivo y acto seguido se clona el proyecto mediante el uso de los siguientes comandos:

```
$ brew install git  
$ git clone repositorio
```

Instalado el repositorio se podrán actualizar y subir versiones con los comandos pertinentes.

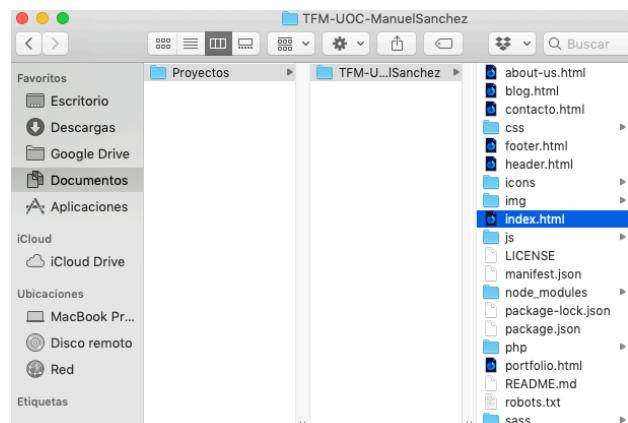


Figura 12: Repositorio GitHub instalado en local.

[Enlace a la documentación](#)

3.3 Instalación NODE JS

La instalación de *Node JS* es un aspecto fundamental, ofrece generar un entorno de trabajo multiplataforma ágil, escalable en la capa de un servidor (local en el caso del proyecto), así como la ejecución de muchas funcionalidades de valor añadido para los desarrolladores web: preprocesadores, optimización de recursos etc., entre otras muchas.

Para su correcta instalación al tener instalado el gestor de paquetes Homebrew el proceso es realmente sencillo, teniendo simplemente que ejecutar el comando marcado.

```
$ brew install node
```

Ejecutada la función se habrán instalado Node y NPM (Node Package Manager) correctamente en nuestro dispositivo, o bien únicamente en el proyecto en cuestión si se desea.

[Enlace al sitio web – Descarga e instalación manual](#)

Tras la correcta instalación, dentro del directorio ejecutamos *npm init* generando el fichero *.json* del proyecto, que registrará las dependencias instaladas entre otras funciones, y el paquete *node_modules* que cargará dependencias a instalar (contiene muchísimos archivos, es importante añadir este paquete al fichero *.gitignore* para no sobrecargar el directorio).

3.4 Node SASS

Node-SASS: Ejecutando el commando *\$ sass input.scss output.scss* preprocesa el código SCSS del proyecto unificándolo en un único fichero CSS.

```
$ npm install node-sass
```

[Enlace a la documentación](#)

Es posible instalar el *framework Bulma.io* y múltiples librerías de iconos, hojas de estilos y otras muchas funcionalidades, pero para el proyecto actual se ha optado por la llamada a varios de estos recursos por CDN, prescindiendo del uso de la terminal.

4. Lenguajes de programación y APIs utilizadas

4.1 HTML y PHP

Los ficheros del proyecto siguen una estructura lógica del lenguaje de marcado por excelencia, HTML, con etiquetas semánticas, sin elementos sobrantes o div vacíos, una práctica muy común en proyectos que no están desarrollados a medida.

Se ha generado una estructura robusta apoyada en el *framework front-end Bulma.io*, aportando estilos predefinidos de manera ágil y sencilla.

Validado el contenido HTML se han convertido los ficheros a PHP de cara a ganar velocidad y escalabilidad una vez se despliegue el proyecto.

Se utilizan *includes* del *header* y del *footer* para modularizar el proyecto y su mantenimiento sea, por ende, menos costoso.

4.2 Framework Bulma.io

Siendo *Bootstrap* el *framework* a nivel *frontend* puntero en el sector, en el proyecto se ha hecho uso de *Bulma.io*. Como puntos a favor se han catalogado la ligereza, limpieza de las clases y fácil estructura, unido a la inquietud por conocer un nuevo servicio de estas características.

No es necesario el uso de *javascript* para la maquetación del proyecto, si bien para determinados servicios como *spinners* o el formulario escalonado se ha precisado añadir las librerías pertenecientes a las extensiones del *framework*.

4.3 Estilos CSS

El resultado el fichero *main.css* que engloba todos los estilos personalizados creados para este proyecto es el resultado del preprocesado del código SCSS escrito. Las clases creadas son fruto de una adaptación al estilo del autor de la metodología BEM.

El uso de SASS con sintaxis SCSS como preprocesador ha permitido estructurar los estilos de forma modular, muy escalable y de fácil acceso, huyendo de las clásicas hojas de estilos interminables, pudiendo afectar a partes concretas para las que se ha dedicado un fichero concreto.

Otro punto positivo es la posibilidad de tener una hoja propia con clases y variables generales para el proyecto, teniendo dichos valores controlados de manera rápida y eficaz.

Se han utilizado funciones para calcular dinámicamente el tamaño de tipografías, así como para generar una hoja de espaciadores estilo *atomic design* propia.

Un aspecto muy presente en la hoja de estilos del proyecto ha sido el uso de numerosas pseudoclasas, como *:before* o *:after* para la personalización de elementos decorativos muy concretos, tales como los *hover* de los botones y enlaces.

4.4 JavaScript

Se ha desarrollado código javascript, dividido de forma modular, para:

- Sistema de filtrado del portfolio
- Despliegue del menú móvil
- Marcado active de los ítem de menú
- Sticky menú
- Spinners, que cargan a modo decorativo antes de cada visualización de contenido.

4.5 Llamadas por CDN

Valorada la situación actual del servidor en la que se alojaría la web, sin demasiada potencia y mucha carga, se ha considerado más ligero y adecuado para el proyecto hacer llamadas a los servicios externos utilizados por CDN, de tal modo que se ejecute en sus servidores más potentes y con una velocidad de respuesta muy buena.

Se han llamado los siguientes servicios:

- Bulma CSS y JS en las páginas que haya sido necesario
- JQuery
- Normalize.css
- Modernizr.js
- Aos.css y aos.js

4.6 XAMMP

Cambiados los ficheros HTML a PHP ha surgido la complicación de la imposibilidad de ejecutarse en local con el servicio *http-server*.

Analizada la situación se ha optado por la descarga de MAMP y ejecución como servidor local en el que corre PHP, basado en Apache.

4.7 Otras herramientas utilizadas

- **Fontello:** Herramienta online para generar tipografías de iconos que, ejecutada desde la terminal, ha creado los ficheros necesarios para que se carguen en el proyecto.
- **Firefox Developer:** Ha sido el navegador principal desde el que se ha trabajado, con múltiples herramientas propias de desarrollador.
- **Brackets:** Editor de código sencillo.
- **Adobe XD:** Prototipado
- **Adobe Illustrator:** Preparación de material gráfico
- **Validadores W3C:** Con valoraciones positivas para el código HTML y CSS3.
- **Google PageSpeed:** Valoración excelente del proyecto actual, pero es un valor a juzgar una vez el proyecto se despliegue al completo en el servidor y dominio definitivo.

Capítulo 4: Implementación y despliegue

1. Requisitos de instalación

Para un correcto despliegue del entorno de trabajo del sitio web de Capybara Studio se precisará de conocimientos básicos en determinadas áreas como el manejo de editores de texto así como el dominio del uso de la terminal del dispositivo de trabajo por línea de comandos.

Será necesaria la instalación de componentes y servicios que nos permitan ejecutar códigos y acciones sin necesidad de que el proyecto se encuentre alojado en un servidor público, tales como:

- Disponer de un server local que ejecute PHP, como pueden ser XAMPP o MAMP.

Recomendable si se busca escalar y trabajar sobre el proyecto:

- Disponer de GitHub instalado en el dispositivo de trabajo.
- Disponer de Node JS instalado en el dispositivo de trabajo si se pretende seguir escalando el proyecto.

En caso de utilizar dispositivos con sistema operativo Mac es recomendable tener instalado o instalar el gestor de paquetes Homebrew

Este entorno, permitirá un flujo de trabajo óptimo y más rápido si se pretende seguir con el desarrollo del mismo o visualizarlo en local.

En caso de que el cliente desee publicar el contenido actual de manera pública el único requisito sería disponer de un servidor al que apunte un dominio al que subir los contenidos del paquete de ficheros.

2. Instrucciones de despliegue

Opción 1: Visualización en servidor

- Opción más sencilla en caso de disponer de un hosting asociado a un dominio al que poder subir, por el programa de FTP deseado a la ruta elegida los ficheros del proyecto.
- Accediendo a la URL del proyecto a través del navegador se desplegará el aspecto gráfico y su funcionalidad.

Opción 2: Visualización en servidor local

- Instalar en el dispositivo MAMP o servicio que de soporte a un servidor local con Apache.
- Abrir el programa e iniciar el servidor web ejecutando Apache.
- Establecer la ruta "localhost" en la que se encuentra el proyecto a desplegar.

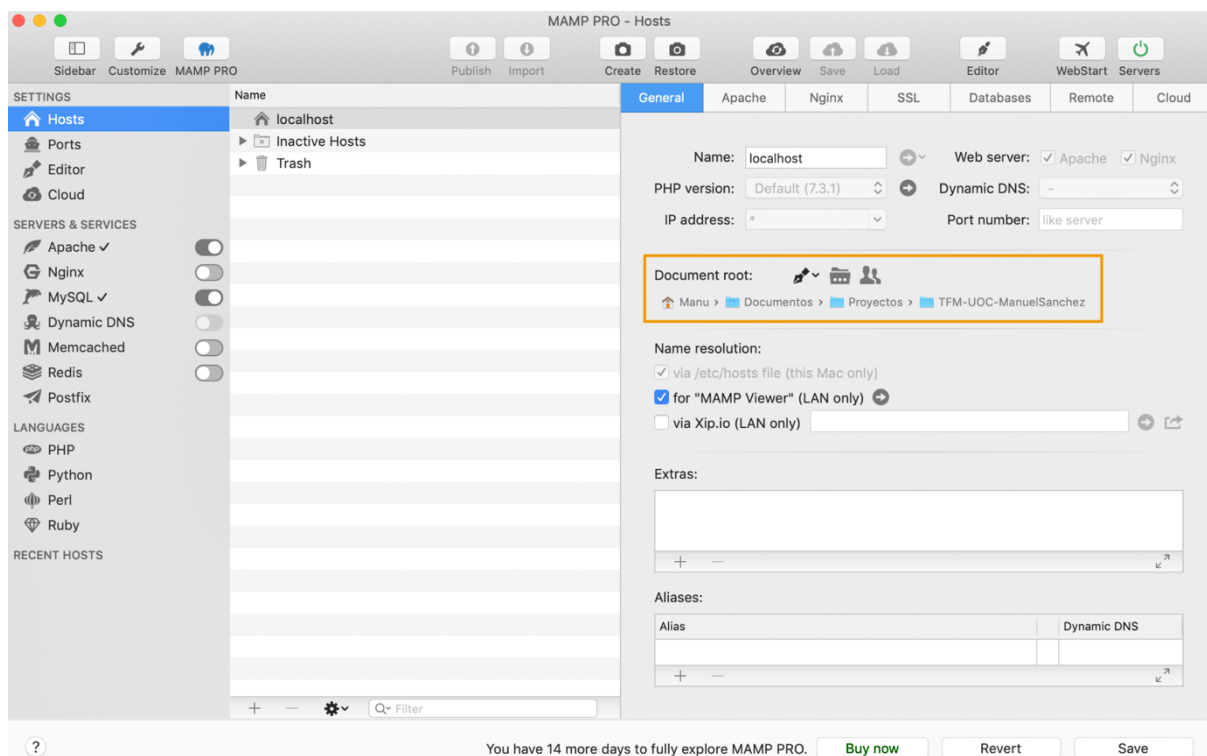


Figura 13: Asignar la ruta del proyecto

- Ejecutar MAMP y visualizar en el navegador.

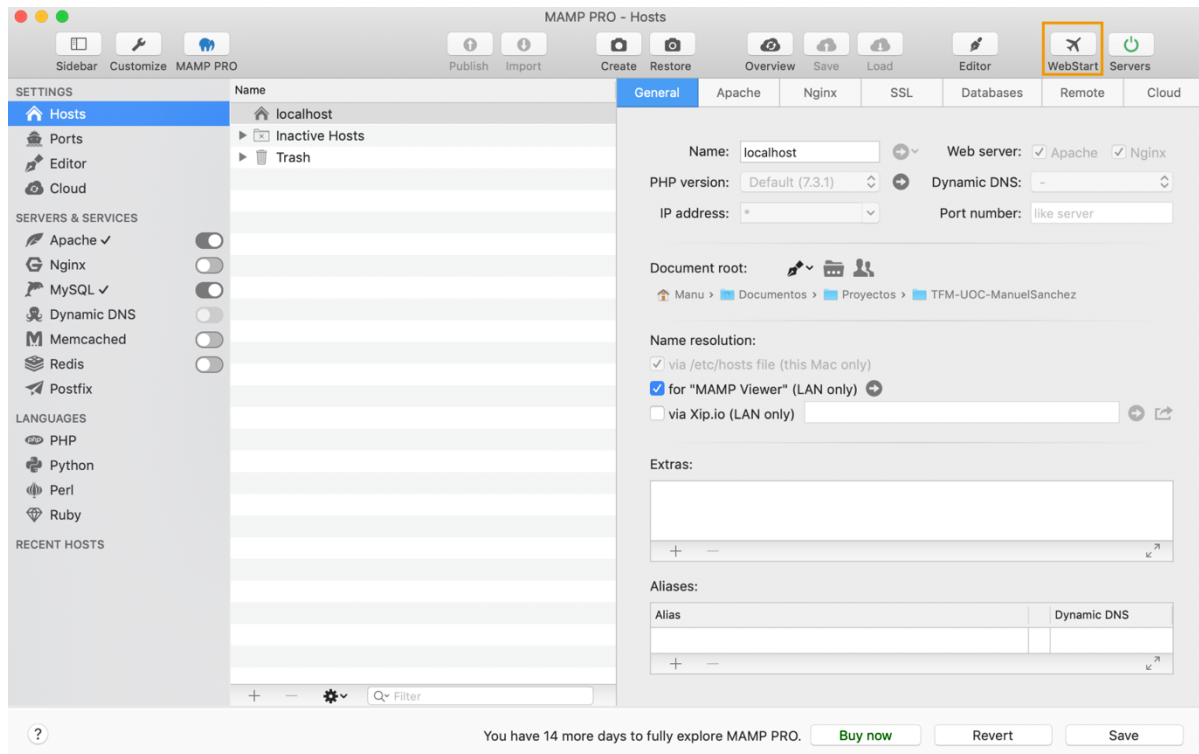


Figura 14: Desplegar la web en el navegador

En caso de querer continuar con el desarrollo del proyecto será necesaria la instalación de Node JS y la dependencia Node-SASS para el preprocesado de código SCSS.

Capítulo 5: Conclusiones y líneas de futuro

1. Conclusiones

La posibilidad de desarrollar la página web de la empresa Capybara Studio ha supuesto un reto y una oportunidad ilusionante de poner en práctica e interiorizar nuevos conceptos aprendidos desarrollando un sitio a medida sin la utilización de gestores de contenido ni plantillas elaboradas por terceros.

El principal motivo de renuncia al uso de gestores de contenido, más allá de poner en práctica conocimientos de maquetación y desarrollo, ha sido la intención de generar un proyecto limpio, con el código justo para que la velocidad sea óptima una vez se despliegue en el servidor y sobre el *framework backend* que los socios de Capybara Studio estime oportuno.

Un punto a mejorar en el futuro tras la realización del proyecto ha sido la planificación, hay que ser realistas, invertir el tiempo necesario en estimar de cuanto tiempo se va a disponer y de cuántos y qué recursos se podrá hacer uso, siempre contemplando que pueden surgir diferentes imprevistos durante el desarrollo. En el caso del proyecto realizado esta fase no se ha hecho correctamente, ya que se han realizado estimaciones cortas pensadas para un trabajador con disponibilidad laboral completa.

La utilización de metodologías ágiles es un facto positivo para este tipo de proyectos, que apoyado de diferentes software como *Asana* o *Trello* permiten una estructura fácil y rápida de entender, siendo necesario poner fechas límite de entrega.

La utilización de recursos vectoriales aporta un valor enorme a los sitios web, considerando tiempo muy bien invertido el empleado en la vectorización de ilustraciones.

Uno de los aspectos, si no el más positivo extraído del proyecto, ha sido poder realizar un código SCSS propio, a media. Resulta sencillo y rápido de modificar y es realmente escalable de cara a futuras ampliaciones.

Los objetivos planteados al inicio del proyecto no han sido logrados en plenitud, pese a realizar un proyecto que se ha considerado positivo, de calidad y con potencial de futuro, se esperaba lograr un producto más animado con animación HTML5 y CSS3 de algunos elementos que no han podido más que afectarse con ligero movimiento gracias a una librería. Esta situación viene derivada de la mala planificación inicial al realizar una estimación contando que el autor dispondría de más tiempo para invertir.

Dicha planificación ha tratado de seguirse, pero diferentes imprevistos han impedido mantener todas las fechas iniciales y durante el proceso han aparecido tareas a desempeñar que bien se han adelantado en la planificación o no se habían contemplado en un inicio.

La metodología seguida ha sido satisfactoria permitiendo una consecución lógica, ordenada y sin perder el hilo conductor del trabajo, apoyada en el software Asana que, a buen seguro, seguirá presente en futuros trabajos individuales o en equipo.

Ha sido necesaria la aplicación de tecnologías no contempladas en la fase inicial para lograr los objetivos, tales como el uso de *frameworks frontend*, situación que al plantear la estructura y comienzo del desarrollo se valoró como negativa ya que suponía prescindir de estructuras robustas y optimizadas que se ofrecen a los desarrolladores sin suponer una carga excesiva de rendimiento. Las decisiones y cambios asumidos han sido considerados beneficiosos para el resultado del proyecto

Cabe destacar que el sector web obliga a los profesionales a no dejar de formarse, las tecnologías avanzan y es importante mantener los proyectos activos, con vida evitando que queden obsoletos.

2. Líneas de futuro

Acciones a realizar a corto plazo:

- Introducción de la información y contenido definitivo reemplazando el texto Loren Ipsum.
- Despliegue en el servidor y dominio sobre un framework PHP a elegir entre Symfony y Laravel, para gestionar el backend de cara al blog y automatización del contenido generado en portfolio.
- Animación del *pattern* de fondo.
- Configuración de Google Analytics.
- Completar políticas de la web y página de error 404.
- Minificación de archivos definitivos.

Mejoras a largo plazo:

- Estrategia SEO.
- Elaborar un panel de usuario para clientes con mantenimientos.
- Repaso y optimización de accesibilidad.
- Animación individual de cada ilustración SVG.

Fuentes documentales

Referencias

- [1] Dentzel, Z. (s.f.). El impacto de internet en la vida diaria - OpenMind. Recuperado 16 mayo, 2019, de <https://bit.ly/2JaA2WH>
- [2] Ana Gómez Codutti, Sonia I. Mariño, Pedro L. Alfonso. (2016). Una propuesta integradora de Mantenimiento Correctivo aplicada al Diseño Web Adaptativo y Accesibilidad Web. Scientia et Technica Año XXI.
- [3] Ruiz González, I. (2019, 2 enero). Tendencias en diseño web para este 2019. <https://bit.ly/2HyAf84>
- [4] García, L. (2018, 10 diciembre). Tendencias en diseño web 2019: lo que viene para quedarse. <https://bit.ly/2BzAvzJ>
- [5] Hassan Montero, Y. (2015). Experiencia de Usuario: Principios y métodos.
- [6] Nielsen, J. (2003). Usability 101: Introduction to Usability. <https://bit.ly/1jNaWu>
- [7] Ronda León, Rodrigo (2008). Arquitectura de Información: análisis histórico- conceptual. <https://bit.ly/1IYm4r>
- [8] WCWAI. (s.f.). Web Content Accessibility Guidelines (WCAG). <https://bit.ly/2sGJGZs>
- [9] Victores, O. (2018, 16 mayo). Investigación sobre rendimiento web - SEMrush 2018. <https://bit.ly/2WjkZj3>
- [10] García, L. (2018a, 10 diciembre). Tendencias en diseño web 2019: lo que viene para quedarse. Consultado 14 mayo, 2019, de <https://bit.ly/2QQHZUJ>

Bibliografía

MacDonald, Matthew. Creación Y Diseño Web : Edición 2012. Madrid: Anaya Multimedia, 2011. Print. Títulos Especiales.

Gutiérrez, R. T. (2018). Understanding the role of digital commons in the web; The making of HTML5. Telematics and Informatics, 35, 1438–1449.

MONJO, Antònia (2009). Diseño web y de interfaces multimedia. Barcelona: FUOC

Brown, D. M. (2007). Communicating Design: Developing Web Site Documentation for Design and Planning. New Riders, Berkeley, CA. 2007. ISBN 0-321-39235-3

PREECE, Jennifer; Rogers, Yvonne; Sharp, Helen; (2011). Interaction design. Beyond human-computer interaction. 3rd edition. United Kingdom: Wiley.

Juliá, S. (s.f.). La importancia de tener una página web para tu empresa. Consultado 7 marzo, 2019, de <https://bit.ly/2TFjais>

Cifuentes, L. (2017, 8 febrero). La importancia de tener un sitio web para tu negocio. Consultado 6 marzo, 2019, de <https://bit.ly/2AAbUXy>

Rodriguez, J. R. 2013. El trabajo final como proyecto. [aut. libro] R. Clarisó y J. R. Rodriguez. Trabajo final de máster. Barcelona: FUOC, 2013.

Sintes Marco, B. (s.f.). Historia de la Web. Páginas web HTML y hojas de estilo CSS. Consultado 7 marzo, 2019, de <https://bit.ly/2UoJ2wc>

ROST, A. (2004). “Pero, ¿de qué hablamos cuando hablamos de interactividad?”. En Congresos ALAIC/IBERCOM 2004, 15 oct. Consultado (7 mayo 2017) en: <http://bit.ly/23WYexO>

García Matilla, L. (2017, 13 julio). DISSENYWEB: USUARIS, INTERFÍCIES I MÒBILS. <https://bit.ly/2UMRwNV>

Ruiz Ruiz, L. (2014, 1 abril). Desarrollo del sitio Web para la Sociedad Latinoamericana de Primatología, SLAPrim. <https://bit.ly/2ug588T>

The Geco, C. (2018, 4 abril). El diseño web en la actualidad y la importancia de un buen profesional. <https://bit.ly/2TX7w2N>

Node.js. Documentación | Node.js. Consultado 23 abril, 2019, de <https://bit.ly/2ERjOkm>

Npm. npm | build amazing things. Recuperado 23 junio, 2019, de <https://bit.ly/1GegdNf>

Bulma.io. Documentation. Recuperado 30 abril, 2019, de <https://bit.ly/2wCQ8TO>

Sass. Sass: Syntax. Recuperado 27 abril, 2019, de <https://bit.ly/2QObbM9>

World Wide Web Consortium W3C. Consultado 20 mayo, 2019, de <https://bit.ly/2QPA2PJ>

Homebrew. Homebrew. Consultado 13 mayo, 2019, de <https://bit.ly/2QU8Lfe>

Git. Git - Instalando Git. Consultado 11 mayo, 2019, de <https://bit.ly/2HW5SYq>

Anexos

Anexo A: Entregables del proyecto

En la entrega del proyecto se presenta un archivo comprimido que contiene:

- Archivo comprimido con todos los archivos web y códigos fuente
- PDF prototipos
- Presentación de diapositivas del proyecto
- Memoria