



Héctor Tornos Faure

Trabajo Obligatorio ARQUITECTURA DE SOFTWARE

Curso 2018-2019

B.Sc. (Hons.) Tecnologías de la información para la
empresa

Validated by University of Wales

CENTRO SAN VALERO
SEAS, Centro de Formación Abierta
ZARAGOZA

Contenido

Introduccion	1
Esqueleto	1
Primeras unidades	1
Ajax y JQuery	1
Repositorio Github	2
Manual.....	3
Pantalla de inicio.....	3
Navbar	3
Tarjetas.....	3
Parte baja	3
Detalle de una unidad.....	4
Ejemplo de ejercicio	4
Ejercicio FeedBack	5
Ejercicio Ajax.....	6
Conclusiones extraidas	9
Puntos fuertes.....	9
Resolucion de los ejercicios.....	9
Uso de framework.....	9
Mejoras en algunos ejercicios.....	10
Integración	10
Puntos débiles.....	10
Escasez de ejemplos	10
Poca presencia de JQuery	11
No reutilización de código:.....	11

Introducción

Sirva el siguiente texto como introducción del trabajo obligatorio de Arquitectura de Software, asignatura de Bachelor TIE en la escuela SEAS. A continuación, voy a describir brevemente cómo se ha realizado el proyecto y qué se puede ver en esta página web. Por supuesto existe una completa memoria donde se trata el tema de una manera más exhaustiva.

Esqueleto

Se ha escogido un framework de CSS para realizar el esqueleto de la misma, con el fin de proporcionarle mejor aspecto. En las primeras unidades han realizado ejercicios sencillos de JavaScript y no se ha profundizado en estilos ni en JQuery, motivo por el cual he escogido contar con la 'ayuda' de un framework como Bootstrap. Cuando queremos incluir elementos en el DOM de la página, esta librería tiene muchas clases ya creadas y simplemente hemos de escoger que la más nos interesa. Por supuesto en temas posteriores dentro de los ejercicios individuales he creado y personalizado mis propios diseños y estilos en CSS.

Primeras unidades

Lo que he intentado hacer, ha sido adaptar todos los ejercicios de los tres primeros temas a una página web completa. No solamente he resuelto los ejercicios por sí mismos, sino que también los he enclavado en la estructura de la página. De este modo todas las funciones de cada unidad están en un solo archivo que se carga al inicio de la página para poder llamar a cada método cuando sea necesario, y además evitamos tener los scripts mezclados con el código HTML.

Ajax y JQuery

En el caso de los ejercicios de Ajax, he desarrollado las funcionalidades dentro del ámbito de la sección correspondiente al ejercicio. El usuario podrá observar como no se ha utilizado Ajax para realizar el esqueleto de la página, y que esta recarga en su totalidad para mostrar las distintas secciones, suponiendo por un lado una pérdida de tiempo y de velocidad de respuesta y, por el otro, demasiada repetición de código. Empecé a desarrollar la página antes de llegar a estas

unidades por lo que los comienzos de la misma se realizaron sin tener en cuenta tecnologías como Ajax.

Repositorio GitHub

Para controlar las versiones y tener un backup del proyecto, he usado un respaldo subiéndolo a mi cuenta de GitHub de manera pública de forma que cualquiera puede descargarse el proyecto desde la url: <https://github.com/hectornos/trabajobachelor.git> y el ejercicio de Feedback desde: <https://github.com/hectornos/feedbackTO.git>

Manual

Pantalla de inicio

La pantalla de inicio está dividida en diferentes secciones cada una de la cual proporciona diferentes informaciones con respecto al trabajo que se ha realizado. A continuación, las voy a describir una por una con más detalle:



Navbar

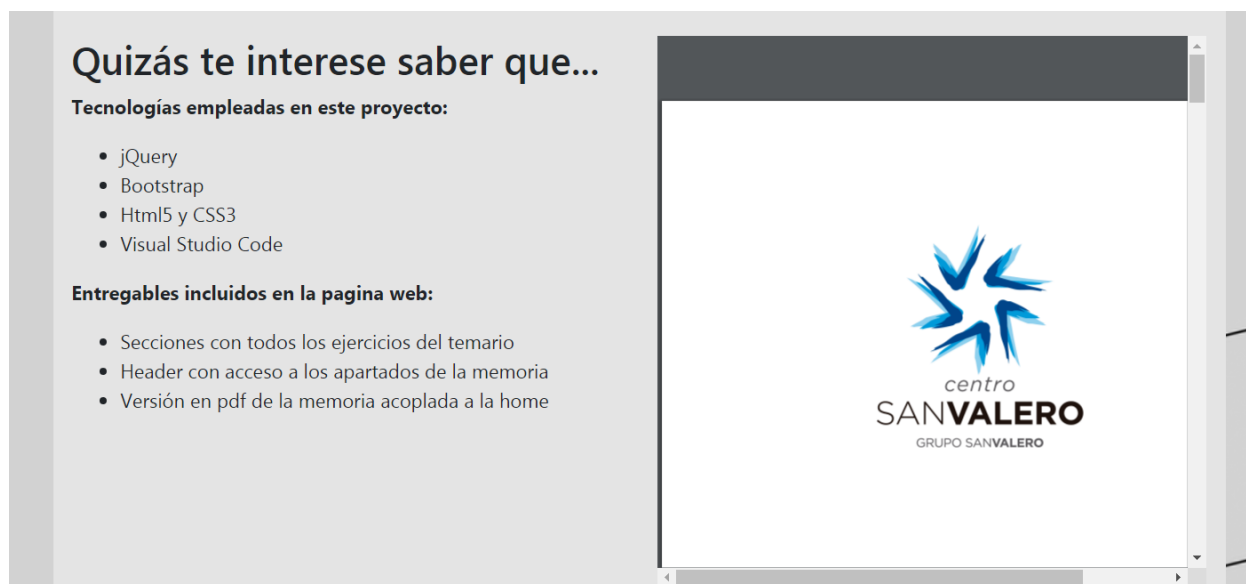
Es una barra superior propia de Bootstrap y que se ha fijado a la parte de arriba de la ventana mediante el modificador `'.fixed-top'`, en la parte de la izquierda tenemos el título del proyecto y el logotipo de la escuela, si hacemos clic (en cualquier parte de la web) podremos volver al inicio rápidamente. En la parte derecha de la barra, tenemos 3 enlaces rápidos a diferentes partes de la memoria del trabajo, y además un desplegable que nos da acceso al índice de cada unidad con los ejercicios de cada una de estas, resueltos.

Tarjetas

Las tarjetas son introductorias de cada unidad, si hacemos clic en la imagen (que ilustra un poco de qué va el temario en esa unidad concreta) accederemos al índice de cada unidad didáctica. Además, vemos una breve explicación de qué podemos encontrarnos en la misma.

Parte baja

En la parte de abajo del índice, se ha incluido alguna información extra de qué tecnologías se han empleado en el desarrollo de la página y más importante, la versión en pdf del proyecto incrustada en la propia página y con su propio scroll. Por supuesto con el menú contextual se puede descargar una copia.



Detalle de una unidad

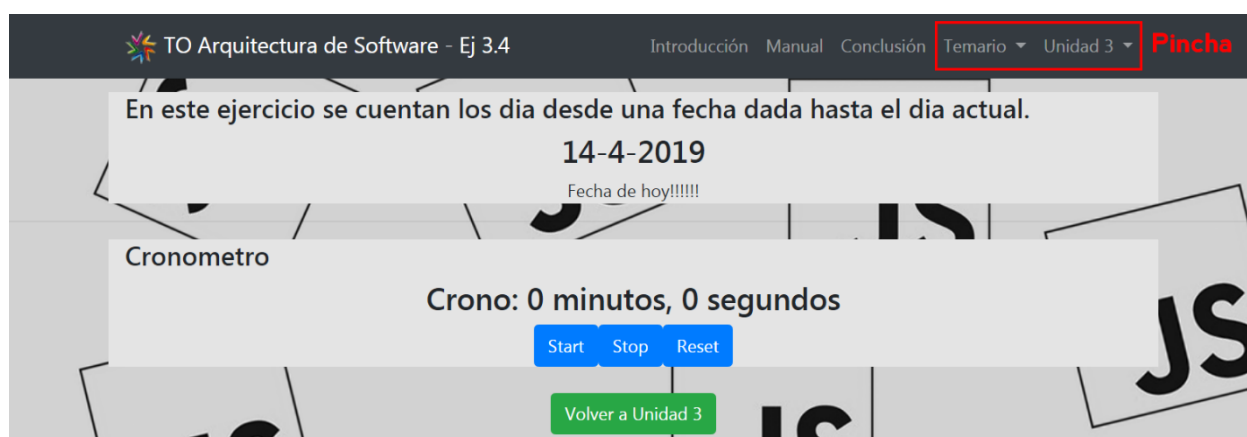
Si accedemos a cualquiera de las cinco unidades en que está dividido el temario, a través del enlace en el texto o en la imagen introductoria de la tarjeta correspondiente, llegamos a un índice donde en una columna se presentan todos los ejemplos interactivos que se han programado y se pueden probar. Por supuesto seguimos teniendo la navegación de la navbar superior, de forma que podemos ir a cualquier otra unidad, al manual o al índice rápidamente.



Ejemplo de ejercicio

Es una barra superior propia de Bootstrap y que se ha fijado a la parte de arriba de la ventana mediante el modificador `'fixed-top'`, en la parte de la izquierda tenemos el título del proyecto y

el logotipo de la escuela, si hacemos clic (en cualquier parte de la web) podremos volver al inicio rápidamente. En la parte derecha de la barra, tenemos 3 enlaces rápidos a diferentes partes de la memoria del trabajo, y además un desplegable que nos da acceso al índice de cada unidad con los ejercicios de cada una de estas, resueltos.



En la parte inferior de cada ejercicio hay un enlace directo al menú de la unidad de la que forma parte. Como se puede ver en la parte superior se puede navegar entre ejercicios de la unidad y unidades del temario, y si se pincha en 'Arquitectura de Software' se vuelve al índice.

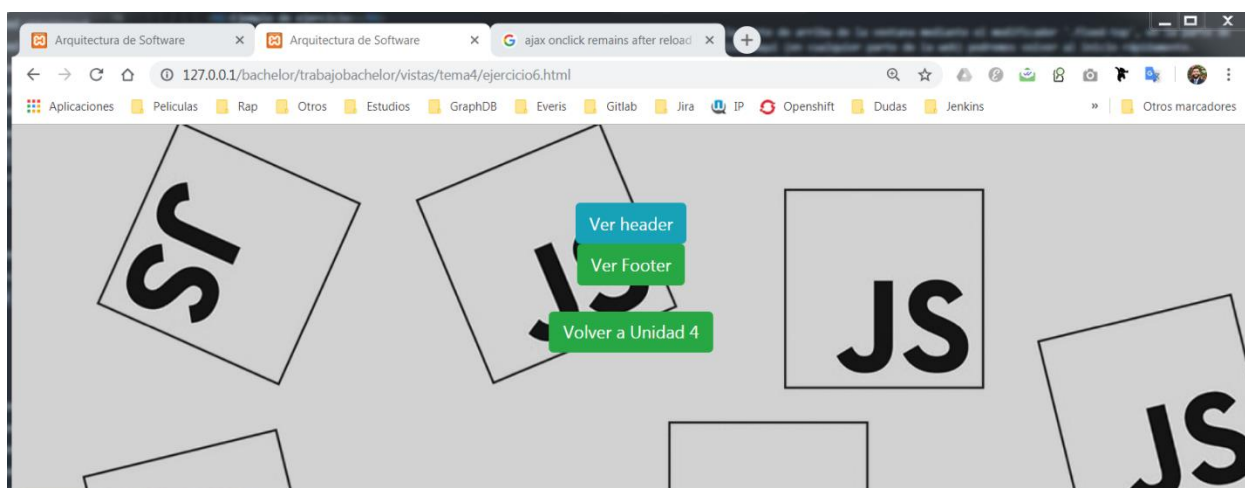
Ejercicio FeedBack

He añadido la resolución del ejercicio de Feedback en la web, para que quede integrado y como parte necesaria para comprender los temas que se explican en la unidad 5. Aunque el sistema de navegación es parecido, es un proyecto a parte por lo que no podremos ver las unidades ni los ejercicios.

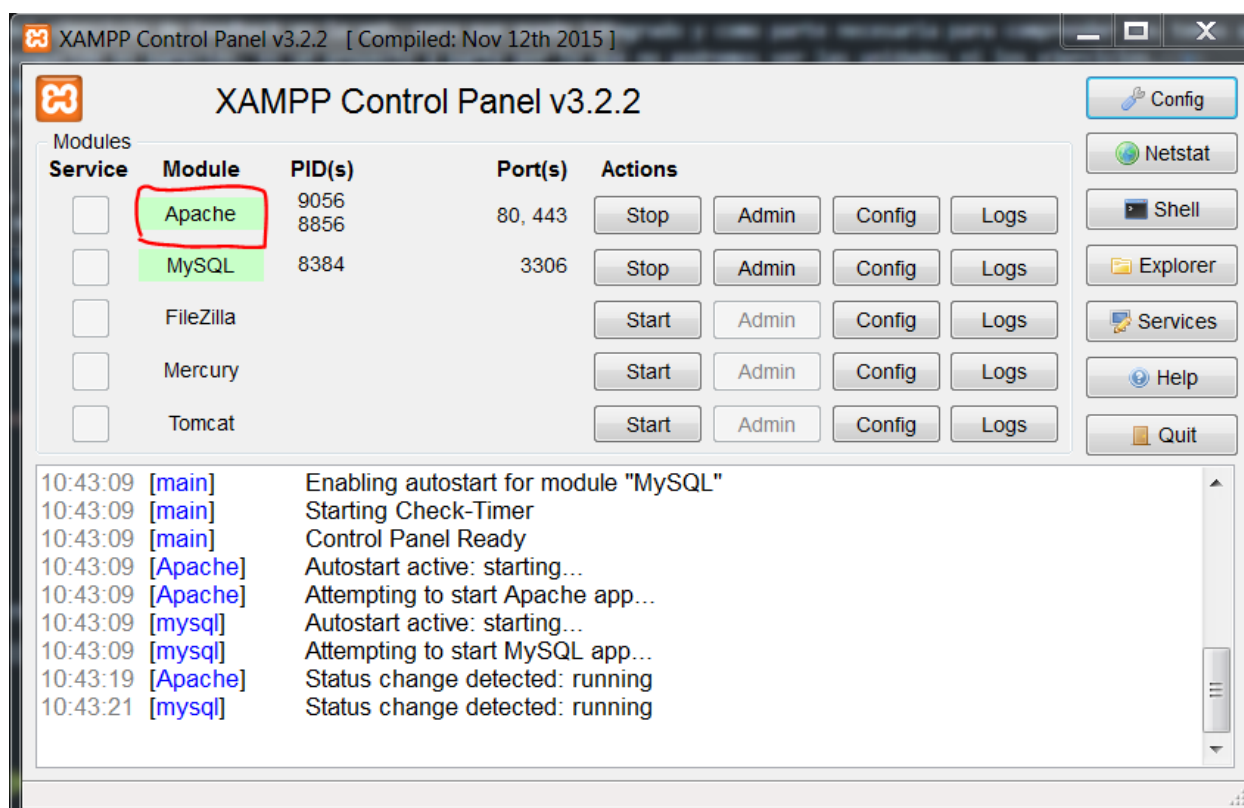


Ejercicio Ajax

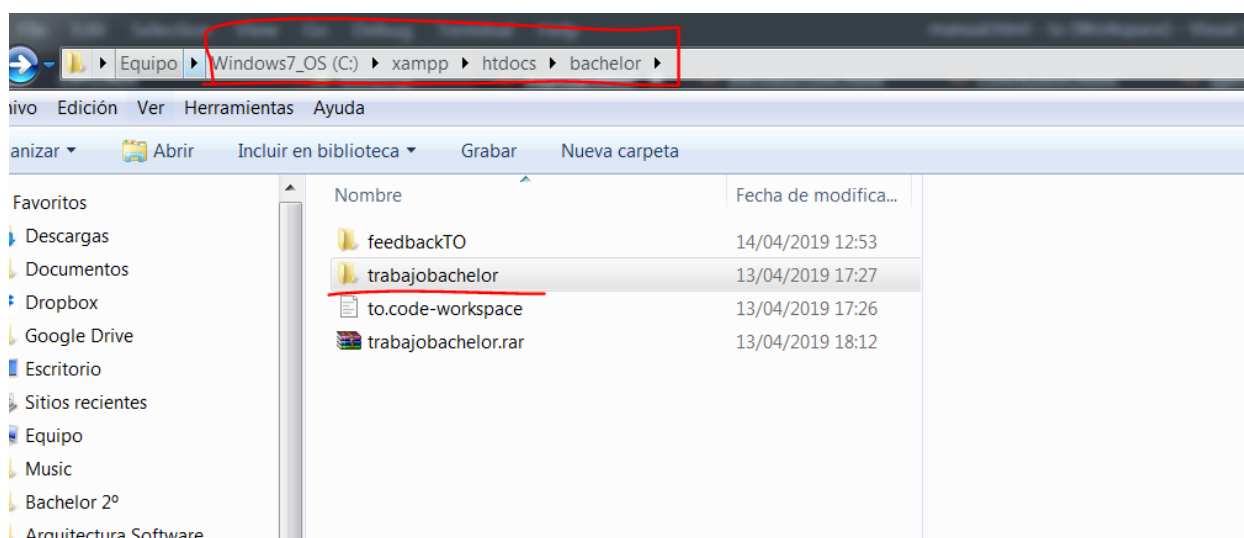
Ajax es una tecnología que necesita que la web se ejecute tras un servidor, el cual funciona de tal forma que se recargan algunos elementos de la página de manera asíncrona y sin necesidad de recarga. He preparado un ejercicio de ejemplo donde partimos de una página vacía con dos botones, si pulsamos en cualquiera de los dos podremos cargar el navbar superior y el footer, sin recargar la página.



A modo de servidor he configurado un Apache con XAMPP para Windows. De esta manera si el usuario quiere ver cómo funciona, debe ejecutar un servidor local y exponer el proyecto a través de dicho servidor. Voy a explicar rápidamente como hacer esto con la herramienta mencionada:



Con la configuración por defecto es suficiente, se ejecuta el software y se le dice que inicie el servidor Apache, que es un servidor web y nos permite exponer nuestro código estático (html+css) pero también obtener ciertas funcionalidades Ajax, por ejemplo.



Por último, emplazamos nuestro proyecto en la carpeta `c:/xampp/htdocs` que es la carpeta que usa el servidor por defecto. Es muy importante colocar en esta carpeta tanto el trabajo obligatorio como el ejercicio de feedback, dentro de la misma carpeta. Si se descargan del repositorio, se deben clonar dentro de la misma ruta, si se usa el archivo comprimido subido a la plataforma, todo está dispuesto para que funcione satisfactoriamente.

Con estas explicaciones estás preparado para ver cómo se navega y se recorren todos los ejercicios del temario. Para más información te recomiendo que revises el código y entiendas de qué forma se han desarrollado todas las funcionalidades.

Conclusiones extraídas

En este apartado voy a hacer una crítica del proyecto desarrollado, de forma que se destaquen las virtudes y los defectos del mismo y así tener un mejor entendimiento de cómo se ha realizado cada uno de los apartados. Lo voy a estructurar en dos partes, 'puntos fuertes' y 'puntos débiles' para que el lector pueda contraponer ambas secciones y sacar sus propias conclusiones.

Puntos fuertes

Resolución de los ejercicios

Aunque todos los ejercicios estaban resueltos en la plataforma, yo he intentado resolverlos por mí mismo. Se puede observar que la estrategia para realizarlo es distinta a la propuesta. Entiendo que copiando y pegando la resolución de los ejercicios se consiguen las siguientes cosas negativas:

- Aprender poco de HTML y nada de JavaScript: Ya que el único punto en que podemos demostrar que sabemos desarrollar código es a la hora de estructurar y codificar nuestro proyecto estático en HTML, pero no realizando las funciones de JavaScript que añaden funcionalidad extra a la página.
- Perder tiempo: Intentando copiar código de archivos en pdf que están protegidos. No tiene sentido perder tiempo copiando manualmente un código que ha desarrollado otro.
- Perder la paciencia: Si intentas integrar ejercicios que están resueltos de manera independiente en un proyecto completo, puedes perder los nervios, tienes que intentar crear una estructura de carpetas y una jerarquía de las mismas que te permita reciclar la mayor parte de código posible y, teniendo claras las referencias, llamar a la función necesaria en cada caso, separando HTML de JavaScript.

Uso de framework

Aunque los framework facilitan el uso de lenguajes de programación y proporcionan funciones ya hechas, pienso que haber utilizado Bootstrap 4 en mi proyecto es algo positivo. En primer lugar, hay que tener claro primero cómo funciona html+css para usar un framework como el mencionado, y por otro lado nos permite dar un aspecto mucho más profesional al sitio e incluir

una estructura 'responsive'. Por último, mencionar que nos permite hacer ciertas modificaciones sobre la hoja de estilos principal y por supuesto añadir nuestros estilos propios. Yo los he recogido todos en el archivo 'custom.css'.

Mejoras en algunos ejercicios

He intentado añadir funciones extra a los ejercicios conforme los resolvía, si se indaga un poco en el código se verá que en algunos ejercicios se cuenta con soluciones extra (por ejemplo, en el ejercicio del cronómetro, o en el que visualiza la posición del ratón). En todos esos ejercicios consideré que para entender los contenidos que se practicaban había que ir más allá de lo que proponía el ejercicio en su enunciado.

Integración

Todos los ejercicios están integrados en el esqueleto de la página. Se puede acceder a todos los ejercicios cómodamente y desde distintos puntos de la página, de forma que recorrer todo el sitio web es cómodo y rápido. Además, el diseño de todas las secciones es hermano de forma que te da la sensación de que estás visualizando un resultado homogéneo.

Puntos débiles

Escasez de ejemplos

He intentado que los ejercicios resueltos contemplen todas las lecciones que se estudian en el temario, pero por motivos de tiempo no he podido incluir todos los ejemplos presentes en el pdf general. Por otro lado, ha sido mi intención incluir todos los aspectos que se mencionan ya sea en la resolución general del ejercicio o añadiendo pequeñas funciones que la complementan. De todas maneras, me gustaría comentar qué conceptos he intentado recoger.

- Formularios: Incluyendo todos los tipos de inputs, y distintos tipos de validaciones con o sin expresiones regulares.
- Sobrescribir clases: Clase Object, clase String...
- CSS: Usando Bootstrap y también estilos propios, incluyendo todos los tipos de inputs, y distintos tipos de validaciones con o sin expresiones regulares.
- Lógica: Switch, for, if...Incluyendo todos los tipos de inputs, y distintos tipos de validaciones con o sin expresiones regulares.

- Objetos JSON: No solamente los objetos propios del DOM sino también creando objetos Json propios (ejercicio facturas).
- Ajax: He intentado demostrar que comprendo cómo se utiliza Ajax y para qué sirve y lo he integrado en el diseño de la página.
- Cronómetros y timeouts: Incluso compartiendo variables entre funciones, las que disparan o controlan y aquellas que se usan cada cierto intervalo. Carrousel.
- Arrays: Trabajando con arrays de distintas maneras.
- Ajax: He intentado demostrar que comprendo cómo se utiliza Ajax y para qué sirve y lo he integrado en el diseño de la página.

Poca presencia de JQuery

Aunque es de utilidad, he preferido dedicar mi tiempo a las funciones propias de JS sin usar ningún complemento.

No reutilización de código:

Como Ajax aparece en la parte final del temario, he diseñado la estructura principal de la página sin reciclar código, de forma que se puede apreciar que el header y el footer se repiten en cada vista. Para reutilizar partes de código o utilizar plantillas hace falta el uso de un servidor, y es una solución que te proporciona Ajax, pero yo estaba habituado a resolver con PHP o un lenguaje de lado servidor

Referencias

Alarcon, J. (2017). *Cómo usar los pseudo-elementos CSS :before y :after con imágenes en HTML - JASoft.org*. [online] JASoft.org. Available at: <https://www.jasoft.org/Blog/post/como-usar-los-pseudo-elementos-css-before-y-after-con-imagenes-en-html.aspx> [Accessed 19 Apr. 2019].

Alvarez, M. (2013). *Entendiendo AJAX en jQuery*. [online] DesarrolloWeb.com. Available at: <https://desarrolloweb.com/articulos/entendiendo-ajax-jquery.html> [Accessed 19 Apr. 2019].

Basalo, A. (2015). *7 formas de incluir Bootstrap en nuestro proyecto*. [online] Academia Binaria. Available at: <https://academia-binaria.com/formas/> [Accessed 19 Apr. 2019].

Daza, Y. (2016). *Componiendo funciones en JavaScript*. [online] Medium. Available at: <https://medium.com/entendiendo-javascript/componiendo-funciones-en-javascript-6fd956bc7f71> [Accessed 19 Apr. 2019].

Ibañez, A. (2018). *Cómo los navegadores web dibujan las páginas HTML/CSS, explicado con viñetas / Microservos (Ordenadores)*. [online] Microservos.com. Available at: <https://www.microservos.com/archivo/ordenadores/navegadores-web-paginas-html-css.html> [Accessed 19 Apr. 2019].

León, D. (2015). *Organizando un sitio web*. [online] Htmlquick.com. Available at: <http://www.htmlquick.com/es/tutorials/organizing-website.html> [Accessed 19 Apr. 2019].

Saquete, R. (2018). *¿Cómo afecta JavaScript y AJAX a la indexación en Google?*. [online] Human Level. Available at: <https://www.humanlevel.com/articulos/indexabilidad/como-afecta-javascript-y-ajax-a-la-indexacion-en-google.html> [Accessed 19 Apr. 2019].