

Planificación del proyecto

Importante: Estamos vinculados por github para trabajar de forma más dinámica con el html, etc. Así que, para añadir modificaciones, trabajamos cada uno en su rama y online compartimos pantalla y las modificaciones que más nos gusten, se añaden a la rama principal.

Día 31: día de repaso (ver clases, tutoriales, etc. de javascript)
Junio

Día 1: día de planificación, primeros pasos (hacer html), reparto de tareas y creación de un cronograma.

Día 2: diseño y maquetación de las diferentes páginas (grid y flex). Empezamos a compartir y debatir ideas sobre cómo queremos que funcione la página. Hemos consultado con otros grupos para saber sus visiones y planteamientos, y así también vemos otros puntos de vista más allá de los nuestros.

Día 3: finalización de las tres maquetas, investigación sobre la creación de páginas similares y repaso de conceptos de javascript que puedan ser útiles. Aurora comienza el css de la página de producto, Raúl se ha encargado de la maqueta de la página de supermercados y Fátima se ha encargado de la maqueta de la página principal.

Fin de semana (4 y 5 de junio): Investigar sobre AddEventListener y localStorage para poder incorporar ya JS a partir del lunes.

Día 6: Comienzo de manejo de javascript, Raúl ha empezado a crear un objeto con los datos de los productos, supermercados y precios, Fátima ha investigado sobre local storage y ha encontrado algunos tutoriales útiles sobre ejercicios similares, además, ha peleado un rato con la página principal y el uso de flexbox para centrar los elementos. Aurora está haciendo que la barra de navegación de todas las páginas y que los elementos de la página de los productos sea responsive con JQuery.

Día 7: Hasta las 11, cada uno hace investigación por su parte sobre DOM, JSON y local storage. Tras el descanso de las 11, ponemos ideas en común. Aurora tiene la tarea de hacer responsive todas las páginas y vincular la página principal con la de productos, y comenzará también a pensar en cómo hacer que la base de datos que hizo Raúl y cómo hacer para ir añadiendo los productos a la lista. Raúl va a seguir explorando el tema de JSON y el manejo de datos a partir de objetos. Fátima comienza a generar con JS la lista de supermercados (con nombre y logo). El objetivo es que el/la usuario/a pueda introducir el nombre de un supermercado y éste aparezca reflejado en una lista debajo del formulario. No obstante, tenemos en cuenta que no todos los supermercados trabajan con nuestro servicio de distribución, así que cuando se introduzca el nombre de un supermercado que no sea Mercadona, Carrefour, Lidl o Alcampo, saldrá una alerta o un mensaje en la página que diga "Lo siento, ese supermercado no se encuentra entre nuestros distribuidores".

Día 8: Fátima ha generado el script de la página de supermercados, de manera que a través del formulario ahora se genera una lista de supermercados, que además contiene la opción de eliminar cada opción. Raúl comenzará hoy a hacer el script de la página principal, donde se tienen que agregar todos los productos elegidos por el usuario, con las características especificadas en el enunciado de la tarea. Raquel se ha unido al equipo.

Día 9: Intentamos cada uno por nuestra cuenta hacer que el localStorage funcione dentro de la página de supermercados. Raquel consigue que se registre el array pero al usuario se le devuelve en pantalla “undefined” por cada uno de los supermercados escritos.

Día 10: En clase, logramos que en pantalla salga cada uno de los valores del array de supermercados. Pensamos cómo hacer que los productos frecuentes se asocien a los supermercados

Día 11: Conseguimos eliminar datos concretos del localStorage en la página de la lista de sùpers. Planteamos la funcionalidad de productos favoritos. Ensayos sobre la función para asegurar su correcto funcionamiento en la interacción del usuario.

Día 13-Día 15: Hemos estado planteando la funcionalidad de la página de productos favoritos. Cómo traer la información del localStorage de los supermercados preferentes y poder unificarlo a los productos que se van incluyendo. Hemos tenido dificultades con esta página respecto a la consecución de las acciones de las diferentes funciones. Raquel trata de aplicar Bootstrap a lo realizado, y hace cambios en la maquetación tanto en HTML como CSS.

Importante: Hemos tenido reunión con Juan y el planteamiento inicial tiene que cambiar. Hemos puesto en común ideas de cómo hacerlo, para repartir tareas. Fátima va a seguir con la página de Supermercados y se pondrá a almacenar la información reflejada en el local storage. Raúl está investigando sobre cómo podemos enlazar esa información con la página de productos, de forma que se generen diferentes columnas, dependiendo del número de supermercados introducidos en la anterior página, y diferentes filas dependiendo de cuántos productos se introduzcan. Aurora va a generar el html de la página de producto y empezar con el script de esa página.

Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
30	31 repaso	1 maqueta	2 maqueta	3 maqueta	4 repaso	5 repaso
6 js	7 repaso	8 (reunión)	9 js	10 js	11	12
13	14	15	16	17	18	19

Primer paso: hacer la maqueta del html de la página de los productos. Con un índice que divida las secciones (verduras, lácteos, carne, pescado...) y la descripción de cada producto debajo de la imagen del mismo. Idealmente, habrá también un botón de “añadir” (a modo de “añadir al carro” en Amazon) para que se guarde directamente en la lista de la compra. Investigaremos más sobre esto. Planificamos la maquetación.

Segundo paso: Una vez que se ha decidido cómo es la configuración de las tres pantallas y su distribución y utilidad (un formulario en la página principal y la de supermercado y una lista de productos frecuentes en la página de producto), incluimos el css para que sea *responsive*.

Tercer paso: Planteamos qué cosas se pueden hacer en la página utilizando javascript. Tenemos claro que necesitamos que la página principal tenga un formulario que genere una lista con los productos que el usuario ha elegido, pero no sabemos si añadir además un buscador para que el usuario pueda añadir productos que, por ejemplo, no se encuentran en la página de productos frecuentes.

Cuarto paso: Tras la reunión con Juan, teniendo claro lo que tenemos que hacer, es importante entender bien cómo funciona el local Storage y empezar a almacenar todos los datos que tenemos.



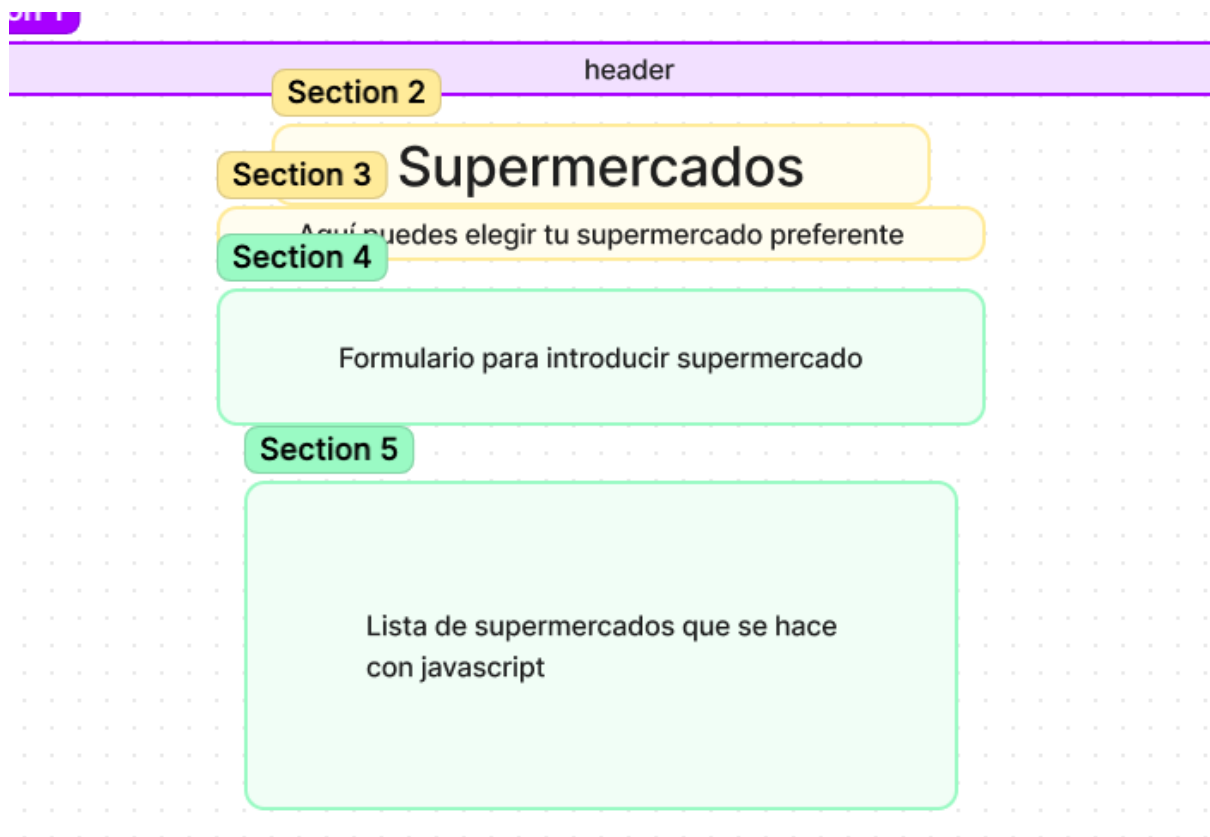
Página principal: Formulario donde el usuario pueda poner alguna palabra relacionada con el producto y que le salgan varias opciones abajo (como en la página de producto), un desplegable con los supermercados disponibles y un botón, o imagen, donde se invite a que el usuario visite la página de los productos frecuentes.

Idea: en la página principal, en la lista que sea ya el resultado, ordenarlo por los mismos supermercados (ej: 2 productos del mercadona, después 3 del alcampo, etc).

Probable cambio de idea: hacer que el usuario ponga los datos él mismo en las páginas del supermercado, productos frecuentes y la lista. Preguntar al “cliente” por si acaso.

Página de supermercados: Como en el ejercicio de superhéroes, debe ser un formulario en el que se incluya información sobre el supermercado y luego aparece una lista con los distintos supermercados. Como trampa, para no complicarnos, podemos poner que si no se

trata de un supermercado Lidl, Alcampo, Carrefour, Mercadona, que no consta dentro de los distribuidores y, por tanto, no aparece luego en la página de producto.



Características CSS (susceptibles de cambios):

Propuesta de tipografía: 'Charis SIL' (<link href="https://fonts.googleapis.com/css2?family=Charis+SIL:wght@700&display=swap" rel="stylesheet">), para h1 y h3.

h1:

```
font-family: 'Charis SIL', serif;
font-size: 45px;
text-align: center;
margin-left: 200px;
margin-right: 200px;
padding: 30px;
color: white;
gradiente: background: linear-gradient(90deg,
rgba(175,92,22,0.9794292717086834) 0%,
rgba(184,84,14,0.9626225490196079) 32%, rgba(255,211,0,1)
100%);
```