

Práctica 4: Frontend

Crear una aplicación frontend

Objetivo: Proveer interactividad a las prácticas anteriores usando JavaScript y manipulación de DOM para obtener una aplicación funcional del lado del **cliente** y realizar operaciones soportadas por solicitudes de HTTP hacia el lado del **servidor**.

Descripción de la práctica

Se completará la parte del frontend (*práctica 1*) referente a la aplicación de e-commerce. Los contenidos de la práctica son los siguientes:

1. Configuración inicial
2. Mostrar la lista de productos
3. Agregar productos al carrito de compras
4. Redirección al carrito de compras
5. Carga del carrito de compras
6. Edición de productos
7. Eliminación de productos del carrito de compras
8. Administración de productos
9. Entregables
10. Evaluación

Resumen de acciones

1. Hacer una petición al backend de los productos y mostrar el listado de productos
2. Agregar los productos actuales al carrito de compras
3. Leer la lista de productos en el carrito de compras
4. Cargar el carrito de compras con los datos del servidor / sessionStorage
5. Modificar los productos del carrito de compras
6. Eliminar productos del carrito de compras
7. CRUD de productos

A todo esto, se le añade que se deben activar/desactivar botones o inputs en algunos momentos, ocultar o mostrar elementos del HTML, solicitar confirmación para realizar una acción, entre otras.

Configuración inicial

Usa el código de la práctica 1 para realizar la actual (si no la tienes, ya vas tarde).

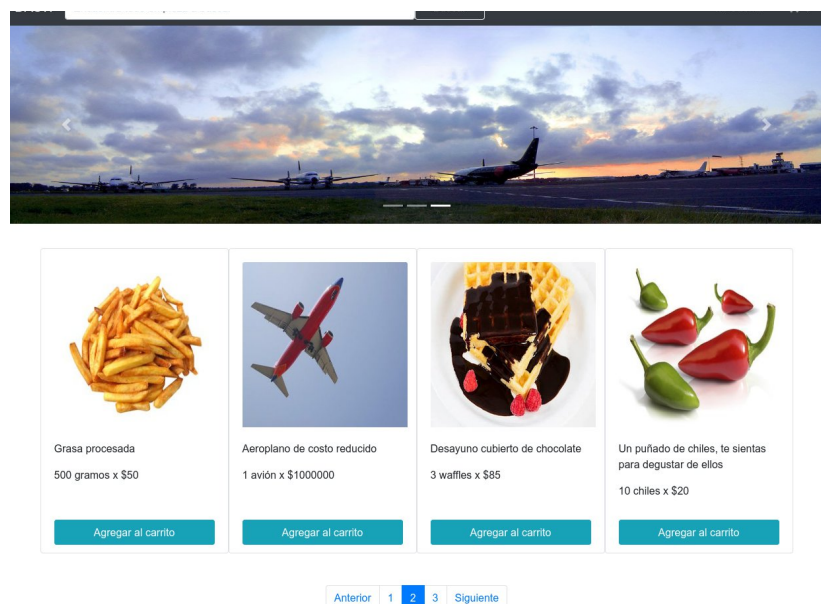
- Podrás quitar los botones que no se necesiten; de igual manera, puedes agregar nuevos en caso de ser necesarios.
- Tienes libertad de modificar tu HTML *siempre y cuando se cumplan todos los requerimientos*
- **Todo el código de JavaScript debe estar en archivos externos con extensión .js**
 - *Estos archivos deben ser importados en el HTML*
 - *Se recomienda (más no se obliga) que cada HTML tenga su propio archivo .js para facilitar el desarrollo*

Mostrar la lista de productos

Lo primero que se debe realizar es la conexión la página principal (**Home**), la cual contiene el listado de productos. Para esto, se hará uso del **Endpoint** en el servidor que lee la lista de productos. **NOTA:** Para hacer llamadas de **HTTP** desde el frontend se deben usar las librerías de **AJAX** (**fetch** o **XMLHttpRequest**).

Posteriormente se tendrá que mostrar la lista de productos usando manipulación del **DOM**. Si realizaste la parte opcional de conversión a **HTML** de la *práctica 2*, puedes usar esas funciones aquí, en caso contrario, créalas para mostrar los productos.

Es importante tratar de usar el paginado para cargar únicamente de 4 en 4 los productos. Apóyate del **DOM** para verificar en cuál página está el usuario y carga el subarreglo correspondiente (*pregunta o investiga* cómo manejar este mecanismo).



Asegúrate de añadir un botón para agregar al carrito cada elemento y que el mismo abra un *modal* para seleccionar la cantidad a agregar.



Agregar productos al carrito de compras

Para hacer el manejo del carrito de compras, se hará uso de la *memoria del navegador* para guardar la información correspondiente. Investiga cómo usar **sessionStorage** y agrega en un archivo de **.js** algunas funciones para manejar el carrito de compras. Se recomienda crear uno para leer, uno para borrar y otro para inicializar.

Una vez que se hayan creado estos métodos, agrega los *eventos* necesarios a las tarjetas de productos para que, al dar *agregar* en el modal, se actualice correctamente el carrito de compras que está guardado en **sessionStorage**.

Redirección al carrito de compras
















Al dar click en el carrito de compras, se deberá cargar la lista de productos correspondientes. Para esto, es posible usar el **Endpoint** correspondiente, teniendo en cuenta la actualización de la información contenida en **sessionStorage**. Si la respuesta es exitosa, se deberá redirigir a la sección de carrito de compras usando **`window.location.href = "url"`**. Conviene que la redirección sea *relativa* ("*shopping_cart*").

Carga del carrito de compras

Al cargar esta página, se deben mostrar todos los productos guardados en el carrito de compras. Para esto, se puede emplear un mecanismo similar al que se usó en la página principal.

Asegúrate de cargar el total de compras correctamente, así como toda la información correspondiente a cada producto.






Resumen de compra

| | |
|--|---|
| Rebanada de pizza  <div> <div>Cantidad: <input type="text" value="2"/>   </div> <div>Precio: <input type="text" value="40"/> \$ m.n.</div> </div>  | Total de compra <hr/> Rebanada de pizza: 2 x 40 Malteada: 1 x 15 Controles de simulador: 5 x 1500 <hr/> Monto a pagar: \$7595 <div>Pagar</div> <div>Cancelar</div> |
| Malteada  <div> <div>Cantidad: <input type="text" value="1"/>   </div> <div>Precio: <input type="text" value="15"/> \$ m.n.</div> </div>  | |
| Controles de simulador  <div> <div>Cantidad: <input type="text" value="5"/>   </div> <div>Precio: <input type="text" value="1500"/> \$ m.n.</div> </div>  | |

Edición de productos

Al presionar el botón de editar (ícono de lápiz), se debe habilitar el input de cantidad y mostrar 2 botones nuevos que sustituirán el botón previo: uno para confirmar el cambio y el otro para cancelar la operación.

Asegúrate de *mantener el valor previo en caso de cancelación de la edición*. Además deberás actualizar el carrito de compras con el nuevo valor en caso de confirmar el cambio. Recuerda validar en caso de que la cantidad sea 0 (eliminar) o algún valor negativo.

| |
|---|
| Malteada  <div> <div>Cantidad: <input type="text" value="1"/>   </div> <div>Precio: <input type="text" value="15"/> \$ m.n.</div> </div>  |
|---|

Eliminación de productos del carrito de compras

Se debe añadir un manejador de eventos al botón y se debe asegurar la eliminación del elemento correspondiente en el carrito de compras. Recuerda actualizar el **HTML** para que se refleje el cambio.

Administración de productos

Esta parte es ***opcional***. Se debe crear una nueva página de **HTML**, donde un administrador podrá ver, modificar y eliminar los productos actuales. El formato es libre, pero se deben conectar correctamente todas las operaciones del CRUD de productos.

Entregables

1. Archivo .zip (no .rar ni otros formatos) con los archivos utilizados
2. Vídeo *narrado* demostrando la práctica y todas las funcionalidades solicitadas en las secciones anteriores; se recomienda que te apoyes de este documento para demostrar cada parte de la misma. ***Sin video, no hay calificación***
 - a. El vídeo ***no debe ir dentro del zip***

Evaluación

| Requerimiento | Puntuación |
|---|------------|
| 1. Se muestra el listado de productos correctamente con los elementos del servidor | 15 |
| 2. El paginado permite cargar 4 productos por página | 10 |
| 3. Es posible agregar elementos al carrito de compras | 15 |
| 4. El carrito de compras contiene la información necesaria para hacer los cálculos del total de compra | 10 |
| 5. Al entrar a la ventana de carrito de compras se cargan correctamente los productos agregados y se muestra el total de compra actualizado | 15 |
| 6. Al dar click en el botón de editar se habilita el input y se muestra la opción de confirmar o cancelar el cambio | 15 |
| 7. Al dar click en el botón de borrar se elimina el elemento del carrito de compras | 10 |

| | |
|---|-----------|
| 8. El carrito de compras se guarda correctamente en el almacenamiento interno (sessionStorage) | 10 |
| 9. Existe una sección para agregar productos nuevos (se debe validar que se llenen los campos correspondientes). Además, se permite la edición y borrado de productos existentes. | 30 |

Máxima calificación de la práctica: 100 puntos.

Consideraciones adicionales

- El alcance de la práctica llega hasta lo descrito únicamente, por ende, si se desean realizar implementaciones adicionales y no funcionan, habrá una penalización por ello
- El vídeo debe ser conciso; puedes explayarte tanto como desees, siempre y cuando lo que expliques sea relevante para la práctica. Procura no producir algo extenso
- Si no estás seguro de algo o te parece ambigua alguna instrucción, **PREGUNTA**
 - No hagas asunciones, es posible que tu entendimiento de algún punto sea distinto a lo que se pide