



Universidad Cenfotec

Fundamentos de Programación Web

Proyecto final:
Aplicación Web con JavaScript

Carrito de Compras:
Vivero Pétalo Vivo

Docente:
Francisco Jose Jiménez Bonilla

Estudiante:
Gabriela Gómez Trejos

Fecha de entrega:
04 de Diciembre del 20525



Carrito de Compras de Pétalo Vivo

El carrito de compras fue desarrollado como parte del sitio web Pétalo Vivo con el objetivo de permitir al usuario seleccionar productos del catálogo, gestionar cantidades, calcular costos finales y avanzar hacia un proceso de pago claro y eficiente. La aplicación mejora la experiencia del usuario mediante interacción dinámica, cálculos automáticos, almacenamiento persistente y retroalimentación visual con microinteracciones. Todo su funcionamiento es completamente intuitivo y está programado en JavaScript, cumpliendo los requisitos del curso.

Indicadores Utilizados

1. Programación JS (funciones, operadores, condicionales, ciclos, arreglos)

La aplicación utiliza JavaScript puro para manejar la lógica del carrito. Incluye:

- Arreglos para almacenar los productos agregados al carrito (**cart[]**)
- Condicionales para validar cantidades, envío y estado del carrito.
- Ciclos **forEach** para recorrer los productos y mostrarlos en pantalla.
Funciones como **addToCart()**, **removeFromCart()**, **updateQuantity()**, **calculateTotal()** y **displayCartItems()**.

2. Eventos y métodos de programación

La aplicación usa diversos eventos:

- **onclick** para agregar productos o modificar cantidades.
- El **addEventListener('submit')** para formularios de newsletter, contacto y encuesta.
- **DOMContentLoaded** para inicializar el carrito y cargar productos.

3. Cálculos financieros

El carrito realiza cálculos automáticos:

- **Subtotal:** suma de precios × cantidad.
- **IVA (13%):** **subtotal * 0.13.**
- **Costo de envío:** 0 o 2,500 según selección.
- **Total final:** subtotal + IVA + envío.

4. Búsquedas

El proyecto incluye una barra de búsqueda funcional en la página de productos:

searchProducts();

Que busca coincidencias en nombre o descripción del producto.

5. Filtrado de datos

Se implementa un filtro por categoría que modifica dinámicamente los productos mostrados:

filterByCategory('plantas', btn);

Lo cual permite que el usuario explore contenido más relevante sin recargar la página.

6. Controles de formulario programados

La aplicación tiene formularios programados en JavaScript, Todos son manejados con eventos **submit**.

- Formulario de **suscripción**.
- Formulario de **contacto**.
- Formulario de **encuesta**.

7. Validaciones con micro interacciones

Se utiliza **SweetAlert2** para mostrar mensajes animados: Producto agregado al carrito, Carrito vacío, Suscripción exitosa y Mensaje enviado.

8. Uso de programación JSON y LOCALSTORAGE

El carrito usa almacenamiento persistente:

Guardado:

```
localStorage.setItem('cart', JSON.stringify(cart));
```

Carga al iniciar la página:

```
let cart = JSON.parse(localStorage.getItem('cart')) || [];
```

Además se almacena: Envío elegido, Total del pedido y el Resumen del carrito.

9. Soluciones programadas (pantalla / SweetAlert / correo electrónico)

Se utilizan varios tipos de soluciones:

- **Por pantalla:** actualización del carrito, totales y productos.
- **SweetAlert2:** alertas personalizadas con animación.
- **Correo electrónico:** Suscripción (template_urulkzp, Contacto y encuesta (template_geaw92e) usando EmailJS. Se ha usado el mismo template para dos funcionalidades por la limitante presentada por la aplicación EmailJs.

10. Base de datos JSON

La tienda carga productos desde un archivo JSON real:

```
const response = await fetch('../JSON/productos.json');
```

Ese archivo funciona como base de datos para:

- Nombre
- Descripción
- Imagen
- Categoría
- Precio