

CODO-A-CODO-2023

COMISION 23520 PYTHON

Integrantes:

Muller Mayra

Marchioli Fernando

Castro Cristian

Brito Zelarayan Paula Guadalupe

Descripción del proyecto: E-Commerce de Snacks- página web dedicada a venta de snacks basado en productos con carrito de compra desde la cual la persona interesa puede elegir entre los distintos productos o simplemente basados en sus preferencias acordes a ofertas combos y descuentos de los mismos, luego los cuales puede agregarlo al carrito de compras que le dirá el nombre del producto y el monto a pagar o en consecuencia en caso de elegir más de 1 producto ofrecerá los productos y el monto total a pagar.

Planificación y Organización: el proyecto se basó en un trabajo por las cual se “dividió” las tareas en distintas partes: un header con distintas opciones como la barra de búsqueda de productos, un carrito de compra y productos, un carrusel con productos, una sección con próximos productos a traer, un pie de página con información de la página web incluyendo un API que muestra el clima actual etc.; esa división cada integrante fue haciendo una parte del proyecto y luego se integró todo de forma consistente y lo más prolija posible para la formal implementación de la página web.

Cosas que quedan por integrar al proyecto:

- hacer que el carrito a la hora de mostrar nombre del producto y monto a pagar lo haga sin mostrar un mensaje alert en js sino que lo haga en un “container” hecho en css .
- hacer un logo propio para el e-commerce.
- a la hora de usar el buscar productos de acuerdo al tipo que desee ya sea ofertas descuentos o combos debería enviarlo a otra página que contenga lo mismo, pero en los productos solo sea los solicitados y no que lo muestre en la misma pagina.
- hacer responsive la página principal ya que parte de la página no lo es debido al carrusel y los próximos productos a traer.
- hacer un menú navbar a la hora de verlo desde un móvil.
- mostrar detalles de los productos por ejemplo peso, marca etc.
- a la hora de elegir entre descuentos combos u ofertas cuando se muestren los productos muestre la cantidad a reducir de descuento o que oferta es específicamente, complementario

a esto se produzca lo mismo en la función en JavaScript para que muestre en tiempo real lo dicho ya mencionado(es decir si hay descuento muestre el % a descontar el nuevo precio y al final el monto a pagar teniendo en cuenta el descuento)

Complicaciones en el proyecto:

Caso del compañero que se encargó de hacer la API del clima y el carrusel:

Sugerencias:

5. Pie de página (Sección "pie-pagina"):

- No me gusta: "...parece provenir de un script llamado "apiclimate.js"."
- La información meteorológica NO se muestra en tiempo real, porque no se actualiza constantemente. Se actualiza una vez, al cargar la página.

...

* Para mostrar la información meteorológica, primero se obtiene las coordenadas mediante geolocalización. Esta acción requiere el permiso del usuario para acceder a la ubicación del dispositivo. A partir de estas coordenadas se obtiene por medio de una solicitud a una API (<https://openweathermap.org/api>) la información meteorológica correspondiente a la ubicación actual. El código necesario para realizar esta función se encuentra en el archivo "apiclimate.js".

(Después me di cuenta que la descripción completa está en: los archivos JavaScript(js): - Apiclimate.js)

6. Scripts:

(Te faltó mencionar "validar.js")

-Contacto.html:

...

También muestra la ubicación de la empresa en un mapa de Google, mediante la utilización de un Iframe.

(Uno de los requisitos era incluir un IFRAME.)

-Apiclimate.js

...la API navigator.geolocation (navigator.geolocation, no es una API)

-Carousel.js

Esto está deshabilitado, porque era lo que generaba el conflicto con la búsqueda.

"4. Los usuarios pueden interactuar con el carrusel girándolo manualmente haciendo clic y arrastrando."

La música también la deshabilité.

"5. El fondo musical se reproduce en segundo plano, y los controles de música (reproducir/pausar) son opcionales."

Casos apartes:

- En el carrito de compras hacer que el button "añadir al carrito" este dentro del producto y logre efecto tomo mucho tiempo entender como era.
- Hacer que se muestren determinada cantidad de productos sin usar grid.
- Efectos de css en productos también costo entenderlos y hacerlos funcionar.
- Lograr que más de 1 página sea responsive utilizando media queries , a veces había partes de la página que eran responsive y otras no.
- Manipular dom para que los productos vayan desde js hasta el HTML (antes teníamos todo en HTML y luego tuvimos que cambiar todo para hacer uso del dom).
- Error en nombres de variables que causo que algunas cosas no funcionaran ejemplo **mostrarofertasButton** en lugar de **mostrarOfertasButton** luego no mostraba ofertas.
- No iniciar variables antes de usarlas
- Errores en lógicas de programación hacia que la sintaxis estuviera bien pero por ejemplo las condiciones para que se filtren productos estaba mal entonces la barra de búsqueda funcionaba pero no mostraba los productos.
- Etiquetas y elementos mal anidados.
- Errores en los selectores: en vez de usar `.` a veces se usaba `#` sin saber que eso hacía que no funcione lo que uno quería.
- Error en el orden de las funciones: ejemplo primero las funciones de buscar productos y luego el arrays de productos hacia que funcionara la lista de productos, pero no la búsqueda de ellos.