



## Parcial No. 1

### Enunciado

Se requiere desarrollar una aplicación web adaptativa (*responsive*) la cual permita realizar pedidos en un restaurante (ver figuras 1 y 2).

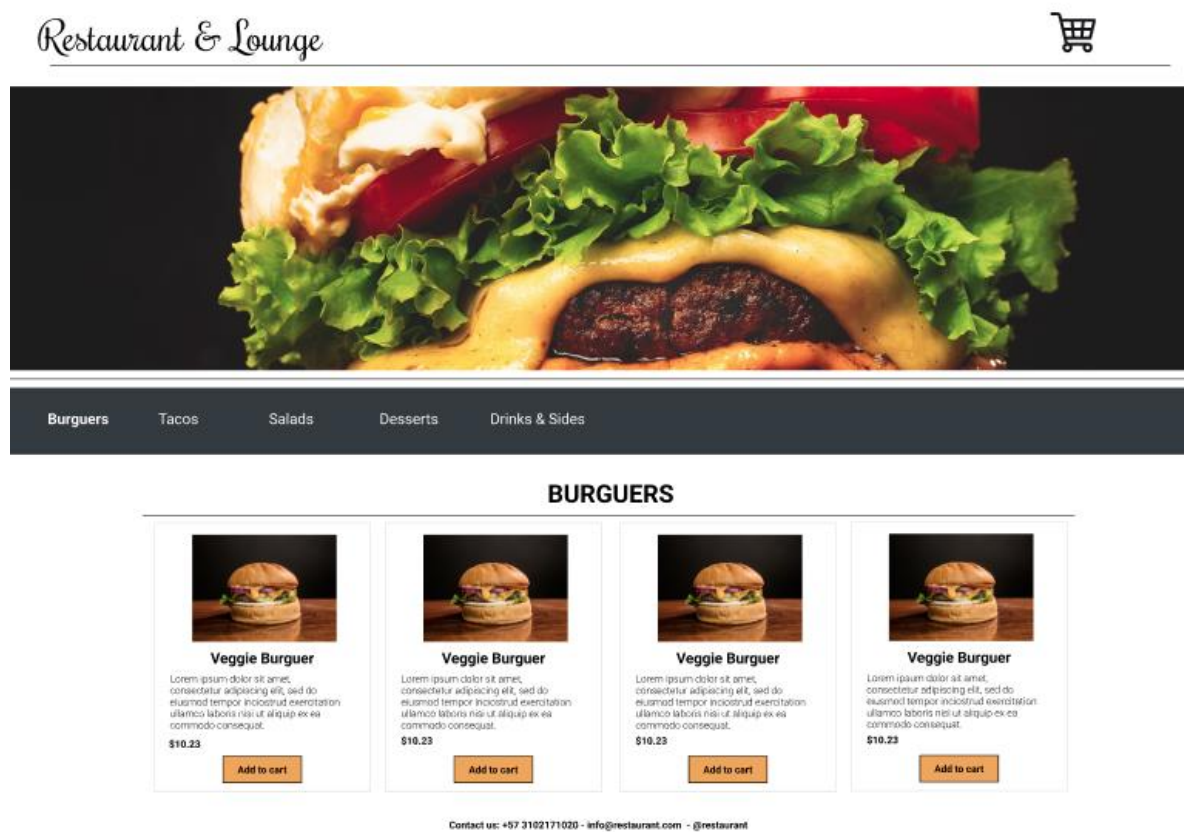


Figura 1. Página principal de la aplicación (versión *desktop*)

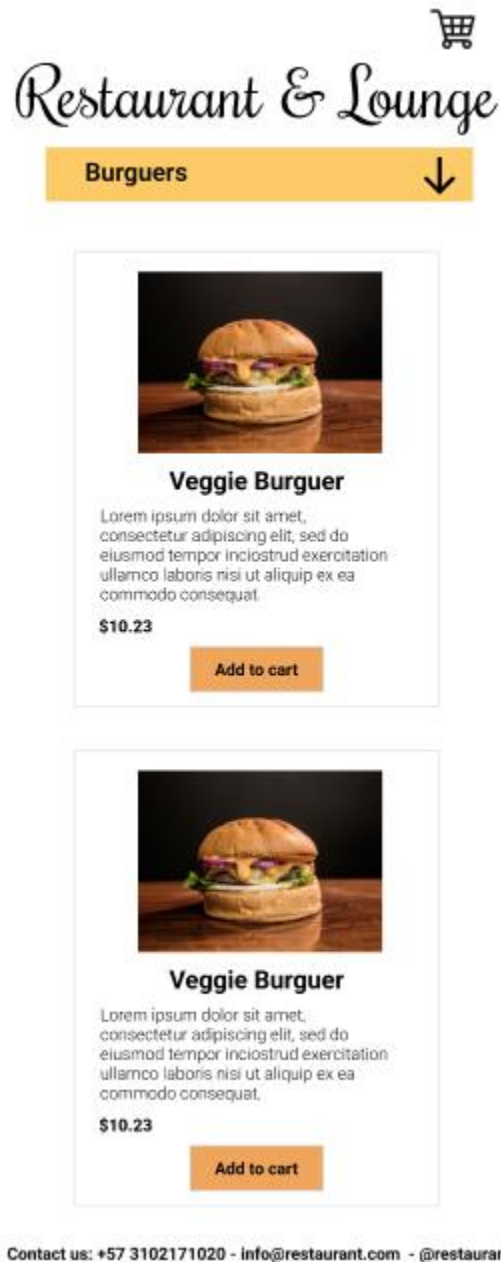


Figura 2. Página principal de la aplicación (versión *mobile*)

### Funcionalidades

1. Cuando el usuario hace clic en cada categoría del menú, se debe mostrar, en la parte central de la página los productos asociados a esa categoría.
2. Cuando el usuario agrega un producto al carro de compras, el texto ubicado en la parte derecha del carrito de compras debe mostrar la cantidad de productos agregados.

3. Cuando el usuario hace clic en el carro de compras, se debe desplegar, en la sección central de la página, el resumen del pedido junto con el valor a pagar (ver Figura 3).

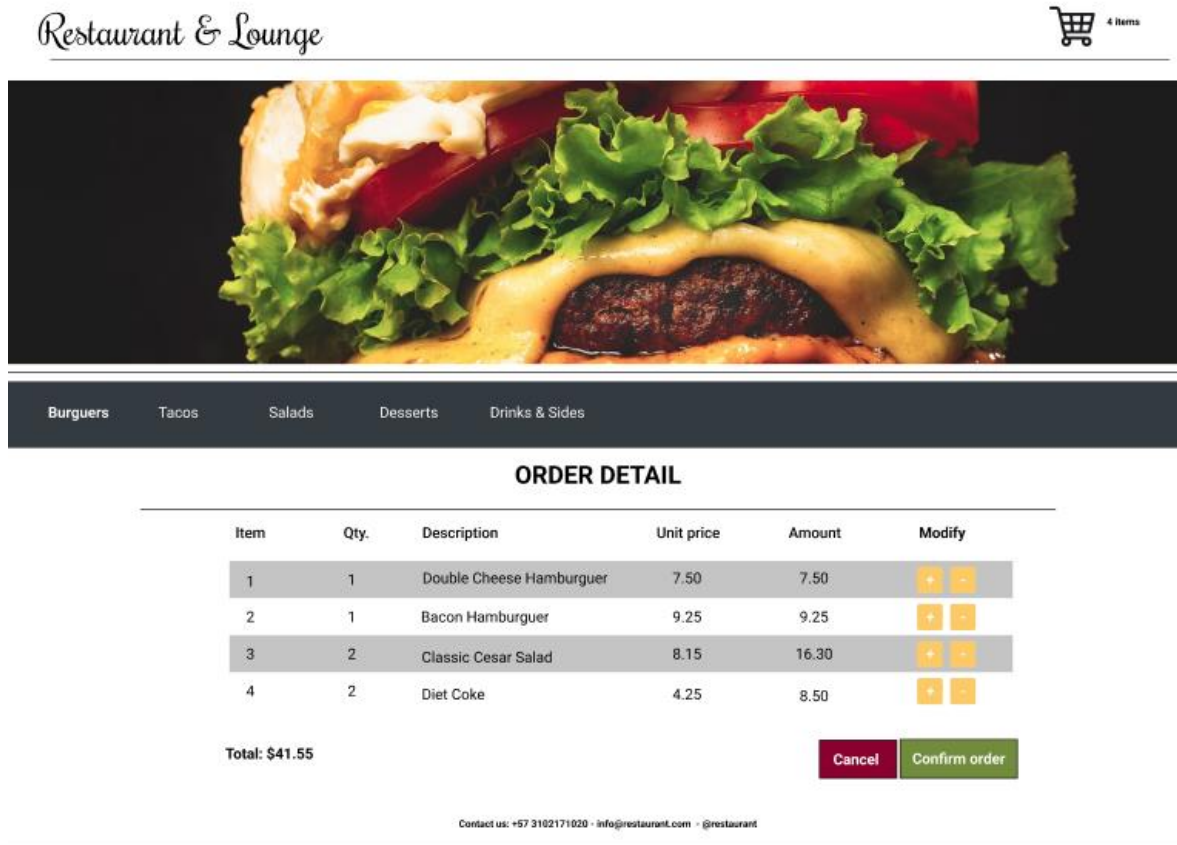


Figura 3. Detalle de una orden (versión *desktop*)

4. Cuando el usuario hace clic en los botones para modificar la orden, las cantidades asociadas a cada product deben aumentar o disminuir (según el botón que se presione), y los valores (*amount* y *total*) deben actualizarse.
5. Cuando el usuario cancela la orden se debe mostrar un mensaje de confirmación. En caso afirmativo, se retiran todos los productos de la orden; en caso negativo, se mantiene la orden sin modificación.

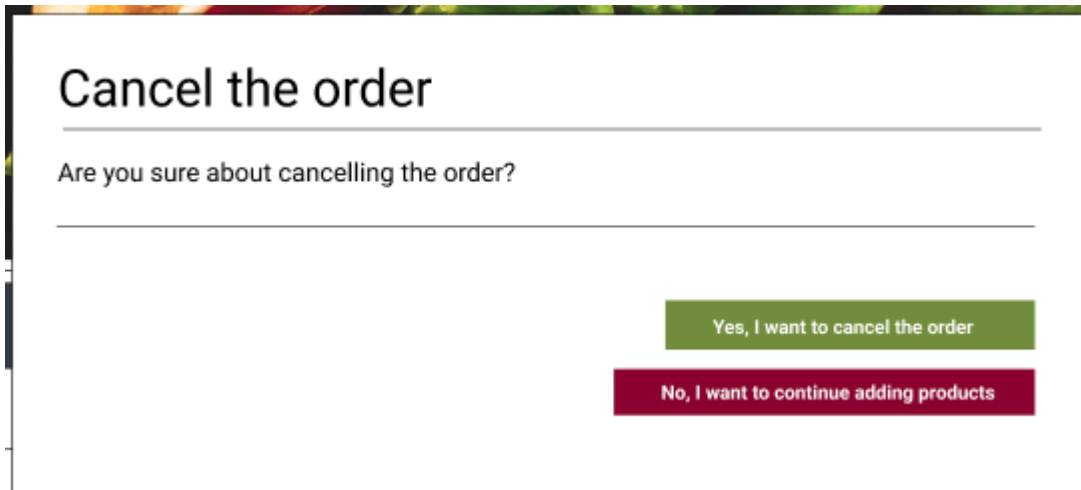


Figura 3. Mensaje para confirmar la cancelación de la orden

6. Cuando el usuario confirma la orden se muestra por consola el listado de productos (ver Figura 4).

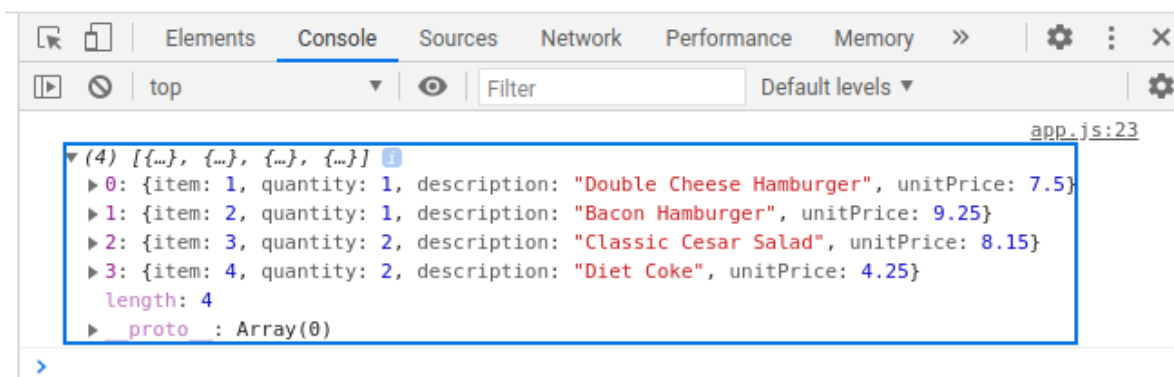


Figura 4. Listado de productos en la consola

## Origen de los datos

Los datos de las categorías y los productos de cada categoría están detallados en el siguiente enlace:

<https://gist.githubusercontent.com/josejbocanegra/9a28c356416badb8f9173daf36d1460b/raw/5ea84b9d43ff494fcbf5c5186544a18b42812f09/restaurant.json>

## Consideraciones técnicas y de diseño

La aplicación será desarrollada usando únicamente HTML, CSS, Bootstrap y JavaScript. No se usará ningún otro *framework* de desarrollo.

Para este parcial el aspecto visual de la aplicación es de suma importancia. Asegúrese de que la interfaz es **exacta** a la presentada en el siguiente proyecto de Figma:

<https://www.figma.com/file/ovVIXZ35YgGKSxPTrqabXy/Parcial?node-id=0%3A1>

En el enlace anterior se detalla el aspecto visual para dos tamaños de pantalla (*desktop* y *mobile*). Use los mismos colores, tamaños, tipografía, imágenes y disposición de los elementos

## Fechas de entrega

Las funcionalidades 1 y 2 deben entregarse a más tardar, el martes 15 de marzo a las 13:45. Para esta entrega el aspecto visual no se evaluará

Las funcionalidades 3, 4, 5 y 6 deben entregarse a más tardar el miércoles 16 de marzo a las 23:59. Para esta entrega se evaluarán todos los elementos de la rúbrica anexa. Los plazos son definitivos y no se extenderán por ninguna circunstancia.

## Entregables

Como entregable (i) suba el código de la aplicación a su repositorio personal; (ii) haga un *release* con el tag 1.0.0 y el nombre parcial1-entrega1 (para la primera entrega) y con el tag 2.0.0 y el nombre parcial1-entrega2 (para la segunda entrega); (iii) adjunte como respuesta a la actividad de BS el zip de los *releases* y en los comentarios de la actividad indique la URL del repositorio. La primera respuesta se debe subir como máximo el martes 15 de marzo a las 13:45. La segunda respuesta se debe subir como máximo el miércoles 16 de marzo a las 23:59.

### Rúbrica de evaluación

Elemento	Puntaje
El diseño de la aplicación es <b>exacto</b> al descrito en el proyecto de Figma (colores, tamaños, tipografía, imágenes, disposición de los elementos)	30
El menú muestra las categorías descritas en el JSON suministrado	10
Cuando se hace clic sobre cada categoría se muestra, en la parte central de la página, el nombre de la categoría, el listado de los productos asociados con ella y se detalla, por cada producto, el nombre, descripción, precio, fotografía y un botón para agregar cada producto al carro de compras.	10
Cuando se hace clic en el botón “Add to car” el texto ubicado al lado derecho del carrito de compras se actualiza con el número de ítems agregados.	5
Cuando se hace clic sobre el carro de compras se muestra el detalle del pedido, con los valores correctos (unitarios y totales).	10
Cuando se modifica la orden las cantidades y los valores (amount y total) se actualizan.	10
Cuando el usuario cancela la orden se debe mostrar un mensaje de confirmación.	5
En caso afirmativo, se retiran todos los productos de la orden; en caso negativo, se mantiene la orden sin modificación.	5
Cuando se confirma la orden se muestra por consola el listado de los productos.	5
El código HTML y CSS se valida correctamente vía <a href="https://validator.w3.org/">https://validator.w3.org/</a> y <a href="https://validator.w3.org/unicorn/?ucn_task=full-css&amp;ucn_lang=es">https://validator.w3.org/unicorn/?ucn_task=full-css&amp;ucn_lang=es</a>	5
El código JS se valida correctamente ejecutar ESLint tomando como referencia el archivo de configuración disponible en <a href="https://github.com/isis3710-uniandes/tutoriales/wiki/%5BNode.js%5D-Instalacion#configuraci%C3%B3n-eslint-recomendada">https://github.com/isis3710-uniandes/tutoriales/wiki/%5BNode.js%5D-Instalacion#configuraci%C3%B3n-eslint-recomendada</a>	5
<b>Total</b>	<b>100</b>