



### <u>Objetivo</u>

Determinar si el/la candidat@ cuenta con los conocimientos para interactuar con una base de datos a través de una aplicación web, implementando cliente y servidor.

## El desafío consiste en:

- Construir una tienda online.
- → Utilizar la base de datos que se encuentra en la hoja 2 de éste desafío..
- → Desplegar productos agrupados por la categoría a la que pertenecen.
- → Generar por separado backend (API REST) y frontend (aplicación que la consuma)
- → Agregar un buscador, el cual <u>tiene que estar implementado a nivel de servidor</u>, mediante una Api Rest <u>cuyo lenguaje</u> y <u>framework puede ser de libre elección</u>. Es decir, los datos de productos deben llegar filtrados al cliente.
- → Desarrollar la aplicación de cliente con vanilla javascript/Vanillajs.
- → ¿Puedo usar react y similares? La respuesta es NO, sólo usar vanilla javascript/ Vanillajs. Si lo desarrollas con react o similares, quedará el ejercicio sin efecto.
- → ¿Puedo usar librerías o componentes específicos?, Si, tales como; boopstrap, material, Jquery, entre otros.
- → Disponibilizar la aplicación en un hosting como a modo de ejemplo, puede ser; Heroku, Netlity, Aws u otro.
- → Disponibilizar el código en Github.

Si quieres agregar algo más, <u>es opcional</u>, tal como, un carrito de compra, filtros por atributo, ordenar productos y paginación. Es tú decisión!!

#### **BASE DE DATOS**

Accede a la base de datos. Es de uso obligatorio

Motor: MySQL

• Host: mdb-test.c6vunyturrl6.us-west-1.rds.amazonaws.com

• Usuario: bsale\_test

Contraseña: bsale\_testNombre db: bsale\_test

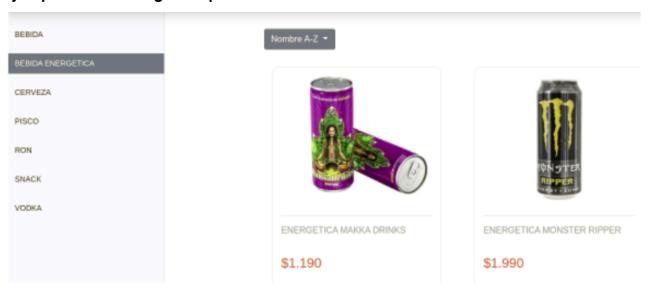
**Tip para la conexión**: El servidor está configurado para que todas aquellas conexiones inactivas por más de 5 segundos sean abortadas, por lo que se requiere controlar la reconexión.

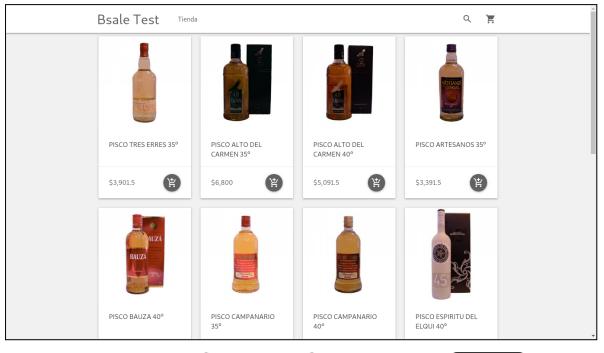
#### <u>Tablas</u>

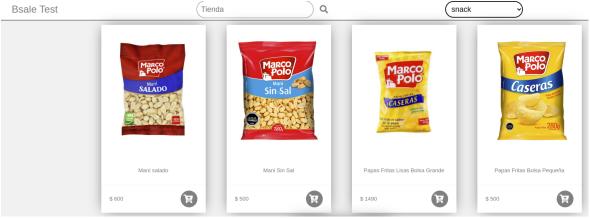
product			
id	Identificador único del producto (int)		
name	Nombre del producto (varchar)		
url_image	URL de la imagen asociada al producto (varchar)		
price	Precio de venta del producto (float)		
discount	Porcentaje de descuento del producto (int)		
category	Identificador de la categoría (int)		

category		
id	Identificador único de la categoría (int)	
name	Nombre de la categoría (varchar)	

# Ejemplos de las imágenes que se encuentran en la BD







## MÁS CLARO QUE EL AGUA...

#### ¿Qué evaluaremos en este desafío?

Categoría	Acción que evaluaremos	Peso		
Excluyente	Se debe utilizar control de versiones	Obligatorio		
Excluyente	La aplicación debe tener deploy	Obligatorio		
Programación	El código debe ser limpio, ordenado y seguir buenas prácticas	400/		
Programación	La aplicación debe ser eficiente y controlar errores	40%		
Documentación *	Documentar la aplicación	20%		
Funcionalidad	Que se entregue buena usabilidad y experiencia al usuario.	40%		
	TOTAL	100%		

## Te queremos ayudar.... ¿Qué es documentación?

- Explicar el ejercicio, en términos de qué lenguaje se usa y detalles, su objetivo, detalle del Apirest y Frontend.
- Revisa acá un <u>Ejemplo de documentación</u>
- Redactar en el Readme del repositorio de ambos proyectos (back y front).

# Para terminar, al finalizar mi desafío, ¿Qué debo enviar a Bsale?

ngresa e	n <u>este</u>	tormu	<u>liario</u> , i	os siguie	entes iink	5
		_				

Link deploy frontend
Link repositorio backeno
Link repositorio fronteno

## Importante mencionar!

<u>Este ejercicio, no tiene ningún fin comercial ni estratégico</u>, sólo queremos medir tus habilidades y conocimientos.

Muchas Gracias y Mucho éxito!!!

