

DESAFÍO



Objetivo

Determinar si el candidato está en condiciones de interactuar con una base de datos a través de una aplicación web, implementando cliente y servidor.

Este ejercicio, no tiene ningún fin comercial ni estratégico, sólo queremos medir tus habilidades...

Ejercicio

Construir una tienda online que despliegue productos agrupados por la categoría a la que pertenecen, generando por separado backend (API REST) y frontend (aplicación que la consume) y utilizando la base de datos que se disponibiliza para su desarrollo.

Además, hay que agregar un buscador, el cual **tiene que estar implementado a nivel de servidor**, mediante una Api Rest cuyo lenguaje y framework puede ser de libre elección. Es decir, los datos de productos deben llegar filtrados al cliente.

Opcionalmente, puedes implementar filtros por atributo, ordenar productos y paginación.

La aplicación de cliente tiene que estar desarrollada con vanilla javascript (javascript puro), sin ningún framework, si puedes usar librerías o componentes específicos, tales como; bootstrap, material, JQuery, entre otros.

Finalmente, disponibilizar la aplicación y el repositorio con el código en un hosting como a modo de ejemplo, puede ser; Heroku, Netlify u otro. También puedes levantar el sitio en una cuenta AWS.

Acceso a la base de datos de uso obligatorio

Base de datos

- Motor: MySQL
- Host: mdb-test.c6vunyturrl6.us-west-1.rds.amazonaws.com
- Usuario: bsale_test
- Contraseña: bsale_test
- Nombre 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 te recomendamos utilizar alguna estrategia de "keep alive"

Tablas

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

BEBIDA

BEBIDA ENERGETICA

CERVEZA


PISCO

RON

SNACK


VODKA

Nombre A-Z



ENERGETICA MAKKA DRINKS

\$1.190




ENERGETICA MONSTER RIPPER

\$1.990


Bsale Test

Tienda




PISCO TRES ERRES 35°

\$3,901.5




PISCO ALTO DEL CARMEN 35°

\$6,800




PISCO ALTO DEL CARMEN 40°

\$5,091.5




PISCO ARTESANOS 35°


\$3,391.5




PISCO BAUZA 40°



PISCO CAMPANARIO 35°



PISCO CAMPANARIO 40°




PISCO ESPIRITU DEL ELQUI 40°

Bsale Test


Tienda

snack




Mani salado

\$ 600




Mani Sin Sal

\$ 500



Papas Fritas Lisas Bolsa Grande

\$ 1490



Papas Fritas Bolsa Pequena

\$ 500

CONOCE LA PAUTA DE EVALUACIÓN

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

- **¿Qué es documentar?** implica explicar ejercicio, Apirest, Frontend y Readme en el repositorio de ambos proyectos (back y front).

Ejemplo de documentación:

<https://apichile.bsalelab.com/lista-de-endpoints/productos-y-servicios/stocks>

Mucho éxito!!!