

# Indicaciones Test - Frontend Developer

## Enunciado

Usted es contratado para desarrollar una landing page y un sistema de gestión de ventas. La empresa que lo contrata tiene sucursales a nivel LATAM y necesita llevar un control de sus proveedores, vendedores, clientes, productos y ventas.

Un proveedor tiene un RUT, nombre, dirección, teléfono y página web.

Un vendedor tiene RUT, nombre, apellidos, dirección, teléfono, fecha de nacimiento, e-mail.

Un cliente también tiene RUT, nombre, apellidos, dirección y teléfono.

La dirección se entiende por calle, número, comuna y ciudad.

Un producto tiene un id único, nombre, precio, stock y puede estar en alguna sucursal.

Una sucursal o Branch office representa a un país de LATAM, ejemplo: Chile, Perú, Argentina, etc.

Por razones de contabilidad, se debe registrar la información de cada venta con un id único, fecha, vendedor, cliente, sucursal y monto total de la venta.

A nivel de detalle de venta se debe guardar cantidad, producto, precio unitario, subtotal, donde el subtotal es igual a cantidad multiplicado por precio. El monto total es igual a la sumatoria de los subtotales.

La moneda de la venta se establece de manera dinámica según el país.

## Lógica del frontend solicitada

1. El administrador de la empresa hace login desde la landing opción Login Ver "UI.pdf" > primera página
2. Una vez autenticado redireccionar al dashboard de nueva venta Ver "UI.pdf" > segunda página
  - a. **New Sale** es la vista principal desde donde se registran las ventas
  - b. El campo **Client** debe permitir buscar entre los clientes registrados y seleccionar uno.
  - c. Si el cliente no existe se debe poder crear y registrar en el sistema, posteriormente, seleccionarla.
  - d. El campo **Branch office** debe permitir buscar entre las sucursales registradas y seleccionar una. Ejem: Argentina, Chile, Perú, etc.
  - e. Cuando se selecciona una sucursal, el campo **Currency** se debe autocompletar en función del país. Ejem: Argentina: AR, Chile: CLP, Perú: PEN, etc.
  - f. En el bloque **Details** se debe poder agregar uno o muchos detalles de producto
    - Un detalle de producto consiste en los siguientes campos:  
**Name:** Nombre del producto,  
**Quantity:** Cantidad de la compra,  
**Price:** Precio de venta,  
**Subtotal:** Quantity \* Price
    - El campo **Name** debe permitir buscar entre los productos pertenecientes a la sucursal y seleccionar uno.
    - Cuando se seleccione un producto, el campo **Price** se debe autocompletar
  - g. Debe calcular el total de venta: **Total**= sumatoria de la columna **Subtotal**
  - h. Debe guardar la venta
3. Debe ver una vista listado de ventas con un botón ver detalle y mostrar todos los datos de una venta: ejem: datos generales de la venta, cliente, sucursal, moneda y el detalle de la venta

## Objetivos

1. Plasmar la solución en un proyecto desde cero utilizando React, puede utilizar Tailwind CSS y Next.
2. Para tener una BBDD poblada, se puede apoyar de cualquier recurso de tercero, lo ideal tener:
  - a. Al menos 1000 productos \*
  - b. Al menos 10 sucursales \*
  - c. Al menos 1000 clientes \*
  - d. Un par de ventas

También es válido si lo crea en un array, JSON, etc dentro del proyecto.

## Sobre la evaluación.

Subir su proyecto a un repositorio con los pasos para poder ejecutarlo y enviarlo por correo.

Se evaluará desde la presentación del proyecto como un usuario final, UI, UX, animaciones, validaciones, pasando a nivel del código las buenas prácticas y finalmente la lógica de negocio solicitada en este TEST. **Se evaluará todo.**

## Plazo.

El plazo de entrega son 3 días.

## Nota

Si la documentación de los pasos para ejecutar el proyecto no son claros y el proyecto no se logra ejecutar, el postulante quedará automáticamente descalificado.