

# Trabajo Practico Final Laboratorio II

Alumno: Alvarez Gonzalo

División: 2ª

Recuperatorio TP3:

Al iniciar el programa nos encontramos con el sistema de stock, se ve de la siguiente manera:

The screenshot shows a window titled "Sistema de Stock" with a close button (X) in the top right corner. On the left side, there is a vertical menu with three buttons: "Cargar Producto" (highlighted with a blue border), "Eliminar Producto", and "Vender Producto". The main area of the window contains a data grid with the following headers: "Id", "Marca", "Tipo", and "Precio". The first row of the grid has an asterisk (\*) in the "Id" column. Below the grid, there are four buttons: "Guardar Stock XML", "Cargar Stock XML", "Guardar Stock JSON", and "Salir". At the bottom left of the window, the text "Recaudacion:" is visible.

Contiene los siguientes botones y funcionalidades:

- Cargar Producto: El botón cargar producto abre un formulario para poder cargar el producto al stock y mostrarlo en el data grid view, se ve de la siguiente manera:

The screenshot shows a dialog box titled "Cargar Producto" with a close button (X) in the top right corner. The dialog box is divided into two main sections. The left section contains three input fields: "ID", "Marca", and "Precio". The right section contains a "Tipo del Producto" section with two radio buttons: "Electrodomestico" and "Instrumento". Below these radio buttons is a "Producto" label. At the bottom of the dialog box, there are two buttons: "Agregar" and "Cancelar".

Si seleccionamos Electrodoméstico, tendremos las siguientes opciones:

**Cargar Producto**

ID:

Marca:

Precio:

Tipo del Producto

☒ Electrodomestico

☐ Instrumento

Producto

- Celular
- Televisor
- Lavarropa
- Heladera

Si seleccionamos Instrumento, tendremos las siguientes opciones:

**Cargar Producto**

ID:

Marca:

Precio:

Producto

☐ Electrodomestico

☒ Instrumento

Tipo de Producto:

- Guitarra
- Saxo

- **Eliminar Producto:** El botón eliminar producto elimina el producto seleccionado en la data grid view y del stock.
- **Vender Producto:** El botón vender producto abre un formulario donde se puede ingresar un cliente y venderle el producto seleccionado en el data grid view, se ve de la siguiente manera:

**Vender Producto**

Datos del Cliente

Nombre:

Apellido:

Producto seleccionado:

ID: 1  
 Marca: Samsung  
 Precio: 30000  
 Tipo Electrodomestico: Celular

Si seleccionamos Generar Ticket se vende el producto y se genera un archivo .txt con los datos del cliente y el producto vendido.

- Guardar Stock XML: El botón guardar stock xml guarda en los archivos stockElectromesticos.xml o stockInstrumentos.xml según corresponda el stock disponible.
- Cargar Stock XML: El botón cargar stock xml permite al programa cargar el stock leyendo un archivo con extensión .xml.

Por defecto viene con los siguientes productos cargados:

Id	Marca	Tipo	Precio
1	Samsung	Celular	30000
2	Gibson	Guitarra	80000
3	LG	Televisor	45000
4	Patrick	Heladera	85000
5	Yamaha	Saxo	120000

- Guardar Stock JSON: El botón guardar stock json guarda en los archivos stockElectromesticos.json o stockInstrumentos.json según corresponda el stock disponible.
- Recaudación: Modifica su valor cada vez que se vende un producto al cliente.
- Data Grid View: Permite ver el stock disponible.

## Temas utilizados:

1. **Excepciones:** Se implementa a lo largo de todo el proyecto donde sea necesario. Excepciones propias utilizadas en FrmCargarProducto, en el método Ingresar.
2. **Test Unitarios:** Se implementa para probar funcionalidades de Producto y sus derivados en el proyecto Test Unitarios.
3. **Tipos Genéricos:** Se implementa en la clase Stock y en FrmInicio
4. **Interfaces:** Se implementa en IArchivo.
5. **Archivos y serialización:** Se implementa en la clase Archivo y SerializadorXml\_Json. FrmInicio utilizar la serialización y FrmVenderProducto la escritura de un archivo .txt

## Correcciones dadas (Corregidas):

Gestión de stock - me falta la parte de atención al público. Si bien hay un botón para vender y un label de recaudación, estaría bueno una vuelta de tuerca para cumplir con la consigna. Faltan algunas validaciones. Si agrego un producto sin seleccionar el tipo (instrumento o electrodomestico) se cierra el form y no pasa nada. Ni se agrega al datagrid (lo cual estaría mal) ni hay un label que me dice que me falta algo. Faltan exceptions propias. Si no hay stock cargado de alguno de los tipos de productos, no es necesario que generes un archivo que queda vacío. Y falta txt.