

# **Trabajo Práctico**

# **Integrador: mateAR**

## **(Segunda Iteración)**

**Alumno: Mondino Facundo.**

**Legajo: 14496.**

**Profesores: Pablo Pioli, Juan Pablo Ferreyra.**

**Cátedra: Diseño de Sistemas.**

**Carrera: Ingeniería en Sistemas de Información.**

**Año 2021.**

## INIDICE:

<b>Dominio</b>	3
<b>Introducción</b>	3
<b>Modelo de negocio</b>	3-4
Problemas a resolver	
Vista interna (BPMN)	
<b>Diseño orientado a objetos</b>	4 a 6
Diagrama de CU	
Diagrama de clases	
Diagrama de secuencia	
<b>Prototipado de interfaz</b>	6 a 8
Diseño de arquitectura	8
Persistencia	9 a 12
<b>Verificación y validación</b>	12 a 22
Pruebas de aceptación	
Pruebas unitarias	
<b>Conclusión</b>	22

## DOMINIO:

El **dominio elegido** para realizar el desarrollo del Trabajo Practico Integrador de la materia se basa en **mates y elementos** que se relacionan con ellos.

## INTRODUCCIÓN:

Durante el desarrollo del documento se dará a conocer **cuál** es el **trabajo** que se **realizó** para lograr la obtención de la **solución** para la **serie de requerimientos** solicitados por la cátedra, que se detallaran más adelante en la sección de modelo de negocio, problemas a resolver.

Las **responsabilidades** del trabajo que se debe realizar fueron **divididas en grupos** formados por tres alumnos de la cátedra. Cada grupo cuenta con un **conjunto de requerimientos** que se les fue asignado durante **dos iteraciones**.

A pesar de que cada uno cuenta con responsabilidades distintas, **el trabajo** que realizan **se relaciona** ya que todos trabajan con el **mismo dominio**. Esto hace posible el **intercambio de información** entre los distintos grupos de trabajo, que se verá en la sección de persistencia.

Para la solución al problema del dominio planteado se pensó un diseño que se verá plasmado en la **documentación realizada**, durante el desarrollo del trabajo, y puesta en práctica en el **código fuente** realizado.

A continuación, se mostrará el **trabajo** efectuado, **documentado** por **secciones** que tratan en **detalle** lo que se realizó en cada una durante el **segundo cuatrimestre de la materia**.

## MODELO DE NEGOCIO:

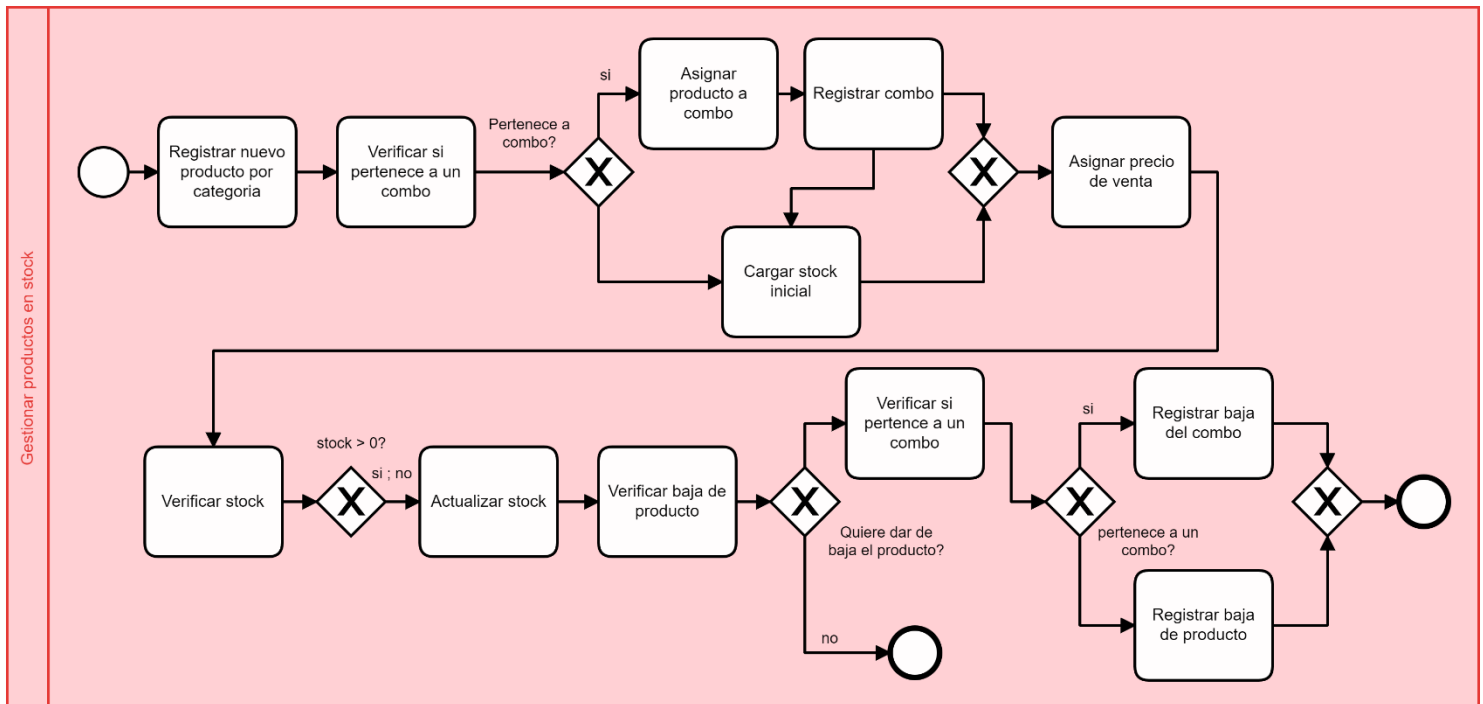
### Problemas a Resolver

Los **problemas detectados** para encontrarles una solución y resolverlos surgen del **administrador**, quien se encargará de ejecutar la solución. Los mismos son:

- El administrador desea poder registrar un nuevo producto con todos sus atributos.
- El administrador desea poder registrar un nuevo combo con todos sus atributos y los productos que incluye el mismo.
- El administrador desea poder actualizar la cantidad de unidades en stock de un producto determinado que ya fue registrado anteriormente.
- El administrador desea poder registrar la baja de un producto que ya fue registrado anteriormente.

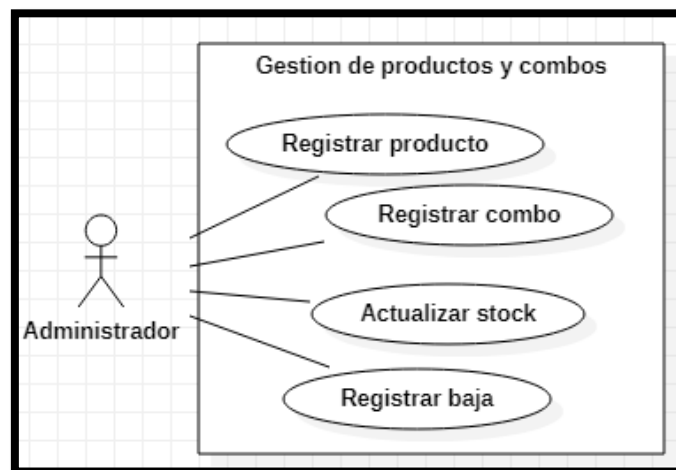
- El administrador desea conocer los productos que están en oferta, para esto se debería permitir cargar una oferta con una fecha de vigencia, por ejemplo, ofertas para el día del padre, ofertas para el día de la tradición, etc.
- El administrador desea poder permitir definir precio en función de la cantidad de unidades de un mismo producto a comprar.

### Vista interna (BPMN)

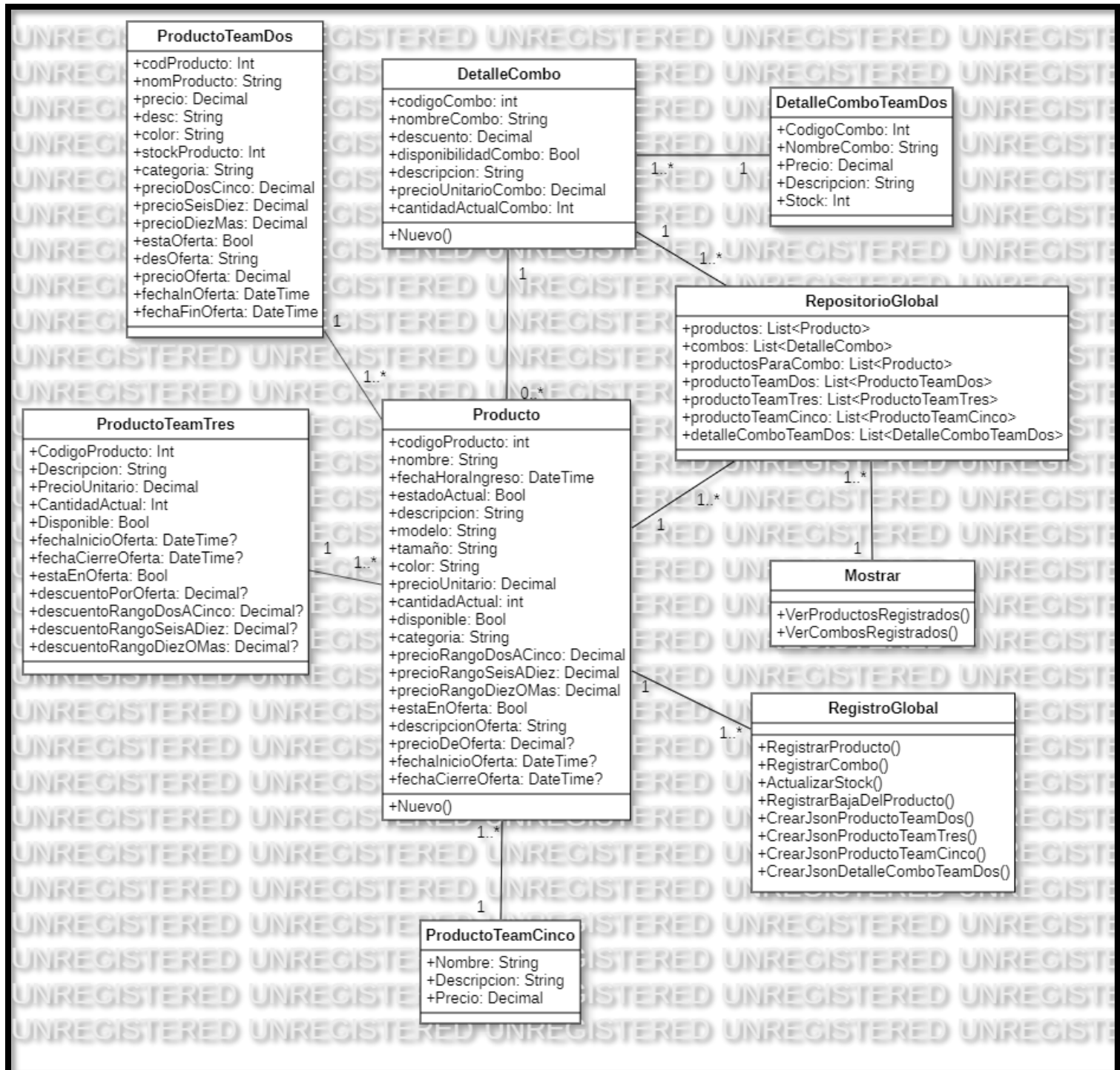


### DISEÑO ORIENTADO A OBJETOS:

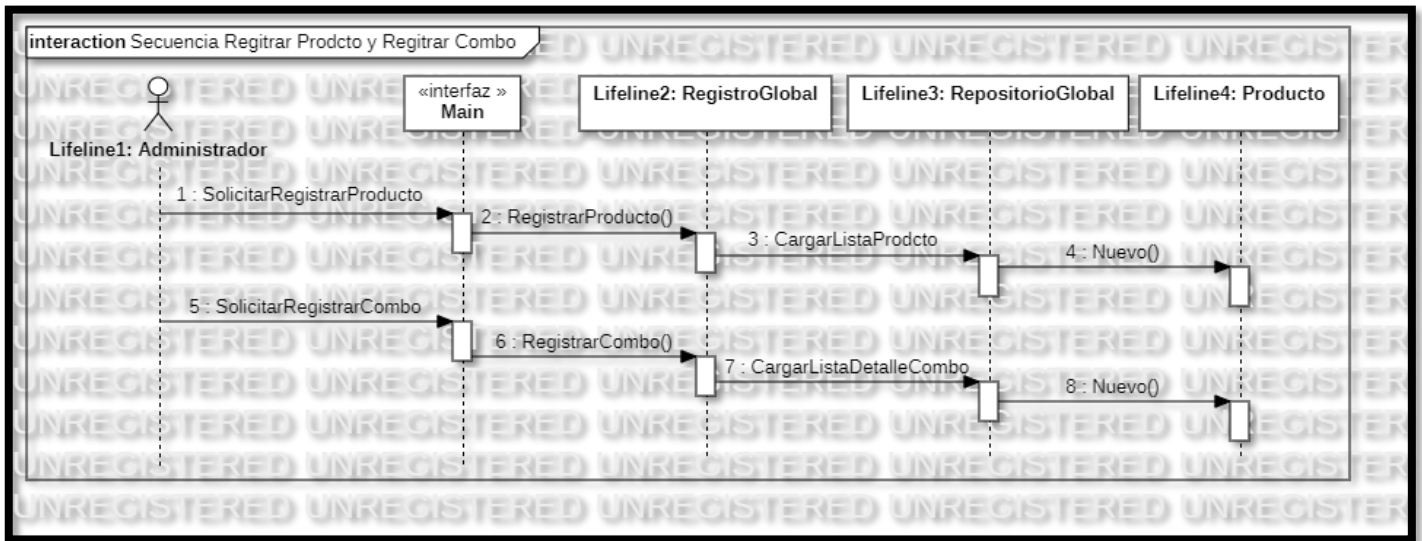
#### Diagrama de CU



## Diagrama de clases



## Diagrama de secuencia



## PROTOTIPADO DE INTERFAZ



**mateAR**

CódigoProducto

Categoría ▶

Modelo

Tamaño

Color

Nombre

FechaIngreso

EstadoActual

Descripción

PrecioUnitario

CantidadActual

Disponible

REGISTRAR PRODUCTO

Categoría A

Categoría B

Categoría C

<- CódigoProducto

Modelo ->

**mateAR**

CódigoCombo

NombreCombo

Descuento ▶

DisponibilidadCombo

NombreProductos

PrecioUnitarioCombo

REGISTRAR COMBO

<- NombreCombo

DisponibilidadCombo ->

## mateAR

### ACTUALIZAR STOCK

INGRESE EL CODIGO DE PRODUCTO

INGRESE LA CANTIDAD

## mateAR

### REGISTRAR BAJA DEL PRODUCTO

INGRESE EL CODIGO DE PRODUCTO

## DISEÑO DE ARQUITECTURA:

Con la complejidad del proyecto con el que se trabajó, puede decirse que se utilizó un diseño de n capas.

Se trabajo con tres capas.

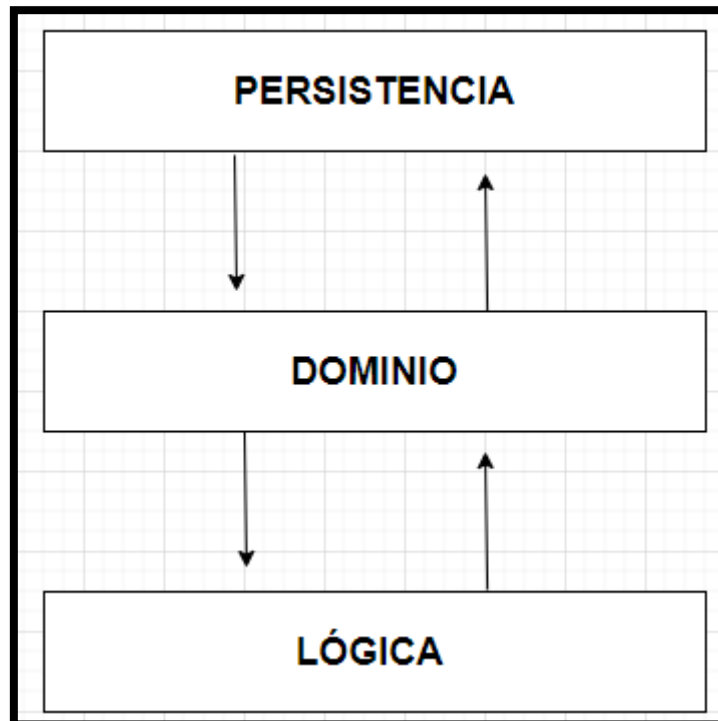
-La primera de ellas es la **capa lógica**, la cual representa el main.

-La segunda es la **capa de dominio**. Esta corresponde a las clases del dominio con el que estamos trabajando.

-La tercera, y ultima, es la **capa de persistencia**, donde allí se encuentra la clase **RepositorioGlobal** y los **DTO** de cada uno de los teams con los que se intercambiò información.



A continuacion se adjunta un **esquema del diseño**. Este representa el diseño de la arquitectura en n capas del proyecto:



## PERSISTENCIA:

En esta sección se explicará **cómo** fueron **guardados** los **datos** durante el desarrollo del trabajo.

Como se explicó anteriormente, una de las decisiones de diseño que se tomaron para almacenar los datos que se fueron cargando por consola, fueron la utilización de distintas **listas** que están definidas en la clase **RepositorioGlobal**.

```
public static List<Producto> productos = new List<Producto>();
```

La lista **productos** fue creada con el objetivo de almacenar todos los objetos de tipo **Producto** que se fueron instanciando. Tal y como se visualiza en el código fuente, cada producto esta formado por atributos cuyos valores se cargan por consola. Todos estos datos se registran en la lista.

```
public static List<DetalleCombo> combos = new List<DetalleCombo>();
```

La lista **combos** fue creada con la finalidad de almacenar todos los objetos de tipo DetalleCombo que se fueron instanciando. Cada DetalleCombo cuenta con atributos cuyos valores son ingresados por consola.

```
public static List<Producto> productosParaCombo = new List<Producto>();
```

La lista **productosParaCombo** fue creada con el objetivo de almacenar todos los nombres que identifican a los productos que forman al combo que se esta registrando. Dicha lista, es utilizada para realizar los calculos para poder definir el precio unitario del combo y almacenarlo en el atributo **precioUnitarioCombo**.

Además, se hizo uso de otras listas que se encargan de almacenar datos que hacen referencia a los distintos **patrones DTO** que se utilizaron para **intercambiar información** con los demás teams de trabajo.

El intercambio se realizó con los grupos: Mostrar los productos y sus respectivos combos (team dos), Carrito de compra (team tres) y Cuenta personal (team cinco).

El mismo se realizó a traves de un formato de intercambio de datos llamado **Json**.

### Intercambio de información con el team dos: (DTO)

ProductoTeamDos
+codProducto: Int
+nomProducto: String
+precio: Decimal
+desc: String
+color: String
+stockProducto: Int
+categoria: String
+precioDosCinco: Decimal
+precioSeisDiez: Decimal
+precioDiezMas: Decimal
+estaOferta: Bool
+desOferta: String
+precioOferta: Decimal
+fechaInOferta: DateTime
+fechaFinOferta: DateTime
+Nuevo()

DetalleComboTeamDos
+CodigoCombo: Int
+NombreCombo: String
+Precio: Decimal
+Descripcion: String
+Stock: Int
+Nuevo()

### Intercambio de información con el team tres: (DTO)

ProductoTeamTres
+CodigoProducto: Int
+Descripcion: String
+PrecioUnitario: Decimal
+CantidadActual: Int
+Disponible: Bool
+fechaInicioOferta: DateTime
+fechaCierreOferta: DateTime
+estaEnOferta: Bool
+descuentoPorOferta: Decimal
+descuentoRangoDosACinco: Decimal
+descuentoRangoSeisADiez: Decimal
+descuentoRangoDiezOMas: Decimal
+Nuevo()

### Intercambio de información con el team cinco: (DTO)

ProductoTeamCinco
+Nombre: String
+Descripcion: String
+Precio: Decimal
+Nuevo()

Se adjunta un **ejemplo de uno de los archivo Json** que se fueron generando para los intercambios. En este caso se adjunta el Json de productos que se le envio a cada uno de los integrantes del team dos.

```
[{"codProducto":1,"nomProducto":"Mate original Stanley","precio":5900.0,"desc":"Mate termico de acero inoxidable","color":"Verde","stockProducto":50,"categoria":"Mates","precioDosCinco":5723.00,"precioSeisDiez":5605.00,"precioDiezMas":5487.00,"estaOferta":true,"desOferta":"Oferta dia de la madre","precioOferta":5310.00,"fechaInOferta":"2021-10-30T00:00:00","fechaFinOferta":"2021-12-20T00:00:00"},
```

```
{"codProducto":2,"nomProducto":"Termo original Stanley","precio":13900.0,"desc":"Termo de de acero inoxidable, con sistema antigoteo y con tapon cebador","color":"Verde","stockProducto":30,"categoria":"Termos","precioDosCinco":13483.00,"precioSeisDiez":13205.00,"precioDiezMas":12927.00,"estaOferta":true,"desOferta":"Oferta dia del
```

padre", "precioOferta":12510.00, "fechaInOferta":"2021-10-30T00:00:00", "fechaFinOferta":"2021-12-20T00:00:00"},

```
{ "codProducto":3, "nomProducto":"Bombilla pico de loro", "precio":690.0, "desc":"Bombilla Alpaca Brasileira con correcta disipacion de calor", "color":"Plateado", "stockProducto":50, "categoria":"Bombillas", "precioDosCinco":669.30, "precioSeisDiez":655.50, "precioDiezMas":641.70, "estaOferta":true, "desOferta":"Oferta dia de la tradicion", "precioOferta":621.00, "fechaInOferta":"2021-10-30T00:00:00", "fechaFinOferta":"2021-12-20T00:00:00" }
```

Cabe recalcar que los demas archivos Json estan cargados en el repositorio Git.

## VERIFICACIÓN Y VALIDACIÓN:

### Pruebas de aceptación

#	Nombre (función que se prueba)	Valores ingresados	Resultado esperado	Resultado observado	Resultado de la prueba (ok o si falló) si falló es un bug	Tipo error	Prioridad
1	Iniciar la ejecución del programa con un valor entero mayor o igual que 1 pero menor o igual que 10.	opcionAingresar = 1	Debería permitir ejecutar la opción del menú mostrado por pantalla y cargar los datos pedidos en la opción 1, Registrar Producto.	Permite la ejecución de la opción 1, lo mismo para las demás ingresando un valor menor a 10.	Ok	-	-
2	Iniciar la ejecución del programa con un valor entero mayor que 10.	opcionAingresar = 11	Debería arrojar error en el ingreso de la opción a ejecutar, ejecutando un valor de opción por default.	No se valida el ingreso de las opciones del menú.	Fallo	Validación ausente	baja

3	Continuar con la ejecución del programa, es decir, ejecutar otra opción del menú habiendo ejecutado alguna opción anteriormente, ingresando un valor entero distinto de 1 y 2.	Opción = 3	Debería arrojar error en el ingreso de la opción a elegir en el momento que el usuario decide si continuar con la ejecución de alguna otra opción o simplemente finalizar con la ejecución del programa solicitándole el ingreso nuevamente del valor de opción.	Permite continuar ejecutando alguna de las opciones del menú talque se realiza la validación correspondiente al caso.	Ok	-	-
4	Registrar nuevo producto con un valor para el código del producto igual a un número no entero.	<b>codigoProducto = 2,6</b> categoría = Termo modelo = Termo clasico 1.4 lts tamaño = 11 x 36 cm. color = Verde nombre = Termo original Stanley fechaIngreso = 23/10/2021 EstadoActual = false descripción = Termo de de acero precioUnitario = 13900 cantidadActual = 66 disponible = true	Debería arrojar error en el ingreso del código del producto ya que el atributo está declarado como un tipo de dato entero, por ende, tendría que solicitar el ingreso nuevamente del mismo.	No se valida el ingreso del código del Producto.	Fallo	Validación ausente	Alta
5	Registrar nuevo producto con un valor para el precio Unitario del producto igual a un numero entero.	codigoProducto = 2 categoría = Termo modelo = Termo clasico 1.4 lts tamaño = 11 x 36 cm. color = Verde nombre = Termo original Stanley fechaIngreso = 20/12/2021 EstadoActual = false descripción = Termo de de acero <b>precioUnitario = 13900</b> cantidadActual = 66 disponible = true	Debería permitir el ingreso de un número entero como precio unitario del producto, ya que, si bien éste fue declarado con tipo de dato decimal, el programa trabaja con centavos 00 por ende no generaría error en los cálculos que se haga uso del dato.	Permite el ingreso de un dato entero para el código del producto.	Ok	-	-
	Registrar nuevo producto con un valor para la	codigoProducto = 2 categoría = Termo	Debería arrojar error en el ingreso de la cantidad	No se valida que la cantidad actual del producto sea			

6	cantidad actual del producto igual a un numero menor que cero.	modelo = Termo clasico 1.4 lts tamaño =11 x 36 cm color = Verde nombre = Termo original Stanley fechaIngreso = 20/12/2021 EstadoActual = false descripción = Termo de de acero precioUnitario = 13900 <b>cantidadActual = -1</b> disponible = true	actual del producto, solicitando el ingreso nuevamente del mismo.	mayor o igual que 0 unidades.	Fallo	Validación ausente	Media
7	Registrar nuevo producto con un valor para la cantidad actual igual a cero, de manera tal que no haya disponibilidad de unidades de ese producto.	codigoProducto = 2 categoría =Termo modelo = Termo clasico 1.4 lts tamaño =11 x 36 cm color = Verde nombre = Termo original Stanley fechaIngreso = 20/12/2021 EstadoActual = false descripción = Termo de de acero precioUnitario = 13900 <b>cantidadActual = 0</b> disponible = true	Debería permitir cambiar de manera automática el valor de la variable de estado que representa si está o no disponible el producto como false, ya que el valor por default de la variable es true.	Permite cambiar el estado actual en stock de la variable de estado que representa la disponibilidad del producto.	Ok	-	-
8	Registrar nuevo producto con un valor para el precio unitario del producto usando un separador de centavos igual a “,” (coma).	codigoProducto = 2 categoría =Termo modelo = Termo clasico 1.4 lts tamaño =11 x 36 cm color = Verde nombre = Termo original Stanley fechaIngreso = 20/12/2021 EstadoActual = false descripción = Termo de de acero <b>precioUnitario = 16789,66</b> cantidadActual = 66 disponible = true	Debería permitir el ingreso de un número no entero como precio unitario del producto cuyo separador de centavos sea coma ya que fue declarado con un tipo de dato decimal.	Permite el ingreso de un número no entero con centavos, separándolos con comas.	Ok	-	-
	Registrar nuevo producto con un valor para el precio unitario del producto usando un separador de	codigoProducto = 2 categoría =Termo modelo = Termo clasico 1.4 lts tamaño =11 x 36 cm color = Verde	Debería arrojar error en el ingreso del precio unitario del producto usando un punto como separador	No se valida el ingreso del precio unitario de producto con separador de centavos punto.			

9	centavos igual a "." (punto).	nombre = Termo original Stanley fechaIngreso = 20/12/2021 EstadoActual = false descripción = Termo de de acero <b>precioUnitario = 16789.66</b> cantidadActual = 66 disponible = true	de centavos, podría generar error en los cálculos que se haga uso del dato ya que el atributo fue declarado con un tipo de datos decimal.		Fallo	Validación ausente	Media
10	Registrar nuevo producto con un valor para el precio unitario del producto usando el punto como separador de miles.	codigoProducto = 2 categoría = Termo modelo = Termo clasico 1.4 lts tamaño = 11 x 36 cm color = Verde nombre = Termo original Stanley fechaIngreso = 20/12/2021 EstadoActual = false descripción = Termo de de acero <b>precioUnitario = 16.789,66</b> cantidadActual = 66 disponible = true	Debería permitir el ingreso de un número que represente el precio unitario del producto haciendo uso de un separador de miles.	Permite el ingreso del precio unitario del producto usando un separador de miles para el dato.	Ok	-	-
11	Registrar nuevo producto con un valor para el precio unitario del producto sin hacer uso del punto como un separador de miles.	codigoProducto = 2 categoría = Termo modelo = Termo clasico 1.4 lts tamaño = 11 x 36 cm color = Verde nombre = Termo original Stanley fechaIngreso = 20/12/2021 EstadoActual = false descripción = Termo de de acero <b>precioUnitario = 16789,66</b> cantidadActual = 66 disponible = true	Debería permitir ingresar un valor para el precio unitario del producto usando el punto como separador de miles.	Permite el ingreso del ingreso del precio unitario del producto usando el punto como separador de miles.	Ok	-	-
	Registrar nuevo combo formado por productos que no fueron registrados previamente, cargandolos a través del codigo	Previamente se registro el productoUno con codigo 1.  codigoCombo = 156 nombreCombo = ComboDos	Debería arrojar error cuando el usuario está añadiendo al combo el producto de código 2 ya que no fue registrado	No se valida el ingreso de un código de producto que no existe cuando se lo quiere añadir al combo.			

12	que identifica a cada uno de los productos.	<b>codigoDelProductoParaCombo = 1</b> <b>codigoDelProductoParaCombo = 2</b> descuentoCombo = 0.10 precioUnitarioCombo = 2500,56 cantidadActualCombo = 66	anteriormente, se tendría que informar por pantalla al usuario un mensaje que le explique lo sucedido.		Fallo	Validación ausente	Media
13	Registrar nuevo combo ingresando un valor numerico entero menor a 100 pero mayor o igual que 0, para el descuento que se le aplicará al combo.	Previamente se registro el productoUno con codigo 1 y el productoDos con codigo 2.  codigoCombo = 156 nombreCombo = ComboDos codigoDelProductoParaCombo = 1 codigoDelProductoParaCombo = 2 <b>descuentoCombo = 0,10</b> precioUnitarioCombo = 2500,56 cantidadActualCombo = 66	Debería permitir ingresar un descuento para el combo con un valor de tipo de dato decimal.	Permite el ingreso de un numero con tipo de dato decimal para el descuento del combo.	Ok	-	-
14	Registrar nuevo combo ingresando un valor numerico entero que represente el descuento que se le aplicará al combo.	Previamente se registro el productoUno con codigo 1 y el productoDos con codigo 2.  codigoCombo = 156 nombreCombo = ComboDos codigoDelProductoParaCombo = 1 codigoDelProductoParaCombo = 2 <b>descuentoCombo = 10</b> precioUnitarioCombo = 2500,56 cantidadActualCombo = 66	Debería arrojar error en el ingreso del descuento del combo, mostrando por pantalla un mensaje que le informa al usuario el ingreso nuevamente del valor correcto para que no genere conflicto en los cálculos en donde se utiliza.	No se valida el ingreso del descuento de un combo.	Fallo	Validación ausente.	Media
	Registrar nuevo combo ingresando un valor para el descuento que se le aplicará al combo haciendo	Previamente se registro el productoUno con codigo 1 y el productoDos con codigo 2.  codigoCombo = 156	Debería arrojar error en el ingreso del descuento informando al usuario que ingrese nuevamente el valor para que no	Permite el ingreso de valor de descuento haciendo uso del punto, pero genera error en los cálculos donde se utiliza este dato, ya			



15	uso del punto. Ej: (0.10%)	<p>nombreCombo = ComboDos codigoDelProductoPara Combo = 1 codigoDelProductoPara Combo = 2 <b>descuentoCombo = 0.10</b> precioUnitarioCombo =2500,56 cantidadActualCombo =66</p>	se genere ningún error en los cálculos donde se hace uso del dato.	que fue declarado con un tipo de dato decimal.	Fallo	Validación ausente	Media
16	Registrar nuevo combo con estado disponible, formado por productos cuya cantidad actual de unidades en stock sea mayor que 0.	<p>Previamente se registro el productoUno con codigo 1 con <b>cantidad actual en stock 66</b>. Ademas se cargo el productoDos con codigo 2 y <b>cantidad actual 77</b>.</p> <p>codigoCombo = 156 nombreCombo = ComboDos <b>disponibilidadDelCom bo = true</b> codigoDelProductoPara Combo = 1 codigoDelProductoPara Combo = 2 descuentoCombo = 10 precioUnitarioCombo =2500,56 cantidadActualCombo =66</p>	Debería permitir añadir los productos que el usuario quiera cuando se verifique que la cantidad en stock de cada uno supera las 0 unidades.	Permite crear el combo con los productos que el usuario ingreso, ya que se realizó la validación correspondiente que verifica que cada producto tenga cantidad de unidades en stock mayor que 0.	Ok	-	-
17	Registrar nuevo combo ingresando el descuento que se le aplicará con un valor decimal que sea igual a 0,00.	<p>Previamente se registro el productoUno con codigo 1 y el productoDos con codigo 2.</p> <p>codigoCombo = 156 nombreCombo = ComboDos codigoDelProductoPara Combo = 1 codigoDelProductoPara Combo = 2 <b>descuentoCombo = 0.00</b> precioUnitarioCombo =2500,56 cantidadActualCombo =66</p>	Debería arrojar error en el ingreso del descuento del combo, ya que en este caso no se estaría efectuando un descuento porque el valor ingresado es nulo, por ende, es ilógico.	No se valida el ingreso del descuento que se le aplicará al combo tanto para valores nulos como también para valores negativos de descuento.	Fallo	Validación ausente.	Media

18	No permitir continuar con la ejecución del programa luego de haber añadido un producto al combo que se está registrando, si se ingresa un valor entero igual a 3.	opcion = 3	Debería arrojar error en el ingreso del valor de la opción, solicitando el ingreso del mismo nuevamente.	Permite mostrar por pantalla un mensaje informando el ingreso nuevamente de la opción a elegir, por ende, se realiza la validación correspondiente ya que solo deja ingresar 1 o 2 dependiendo de la opción que se elija.	Ok	-	-
19	Registrar nuevo combo con un valor, para la cantidad actual de unidades de ese mismo combo, igual a un numero menor que cero.	<p>Previamente se registro el productoUno con codigo 1 y el productoDos con codigo 2.</p> <p>codigoCombo = 156 nombreCombo = ComboDos codigoDelProductoPara Combo = 1 codigoDelProductoPara Combo = 2 descuentoCombo = 0,10 precioUnitarioCombo = 2500,56 <b>cantidadActualCombo = -1</b></p>	Debería arrojar error en el ingreso de la cantidad actual del combo solicitando el ingreso nuevamente del mismo.	No se realiza la validación correspondiente para que la cantidad de unidades del combo sea mayor o igual que 0.	Fallo	Validación ausente.	Media
20	Registrar nuevo producto ingresando la fecha de ingreso con un separador del día, mes y año usando definido mediante un "-" (guion medio).	<p>codigoProducto = 1 categoría = Termo modelo = Termo clasico 1.4 lts tamaño = 11 x 36 cm color = Verde nombre = Termo original Stanley <b>fechaIngreso = 20-10-2021</b> EstadoActual = false descripción = Termo de de acero precioUnitario = 16.789,66 cantidadActual = 66 disponible = true precioUnitario = 16789,66 cantidadActual = 66</p>	Debería permitir ingresar la fecha en la que se registra el producto con el formato de día-mes-año usando el separador "-" ya que el programa la trabaja automáticamente con el separador "/".	Permite ingresar la fecha en la que se va a cargar el producto usando un separador guion para el día, mes y año.	Ok	-	-
	Registrar nuevo producto cuya	codigoProducto = 1 categoría = Termo	Debería arrojar error en el ingreso	No se valida que el día en el que se			

21	fecha de ingreso este formada por un día mayor a 31.	modelo = Termo clasico 1.4 lts tamaño =11 x 36 cm. color = Verde nombre = Termo original Stanley <b>fechaIngreso = 36/10/2021</b> EstadoActual = false descripción = Termo de de acero precioUnitario = 13900 cantidadActual = 66 disponible = true	de la fecha, solicitando que la misma vuelva a ser ingresada.	registra el ingreso del producto sea menor o igual a 31.	Fallo	Validación ausente	Alta
22	Registrar nuevo producto cuya fecha de ingreso este formada por un mes mayor a 12.	codigoProducto = 1 categoría =Termo modelo = Termo clasico 1.4 lts tamaño =11 x 36 cm. color = Verde nombre = Termo original Stanley <b>fechaIngreso = 20/15/2021</b> EstadoActual = false descripción = Termo de de acero precioUnitario = 13900 cantidadActual = 66 disponible = true	Debería arrojar error en el ingreso de la fecha, solicitando que la misma vuelva a ser ingresada.	No se valida que el mes en el que se registra el ingreso del producto sea menor o igual a 12.	Fallo	Validación ausente	Alta
23	Registrar nuevo combo con un valor para el código del combo igual a un número no entero.	Previamente se registro el productoUno con codigo 1 y el productoDos con codigo 2.  codigoCombo = 156,98 nombreCombo = ComboDos codigoDelProductoPara Combo = 1 codigoDelProductoPara Combo = 2 descuentoCombo = 0,10 precioUnitarioCombo =2500,56 cantidadActualCombo = 66	Debería arrojar error en el ingreso del código del combo ya que está declarado como un tipo de dato entero solicitando el ingreso nuevamente del mismo.	No se valida el ingreso del código del combo.	Fallo	Validación ausente	Alta
24	Registrar nuevo combo con un valor para el código del combo	Previamente se registro el productoUno con codigo 1 y el productoDos con codigo 2.	Debería arrojar error en el ingreso del código del combo, solicitándole al	Permite el ingreso del código del combo con un valor entero negativo, pero el	Fallo	Validación ausente	Baja

	igual a un número entero negativo.	<b>codigoCombo = -150</b> nombreCombo = ComboDos codigoDelProductoPara Combo = 1 codigoDelProductoPara Combo = 2 descuentoCombo = 0,10 precioUnitarioCombo = 2500,56 cantidadActualCombo = 66	usuario el ingreso nuevamente del mismo.	programa fue pensado bajo la idea de que el código de un combo se represente mediante valores enteros positivos, por ende, no se realiza la validación correspondiente.			
25	Registrar nuevo producto con un valor para el código del producto igual a un número entero negativo.	<b>codigoProducto = -1</b> categoría = Termo modelo = Termo clasico 1.4 lts tamaño = 11 x 36 cm. color = Verde nombre = Termo original Stanley fechaIngreso = 20/15/2021 EstadoActual = false descripción = Termo de de acero precioUnitario = 13900 cantidadActual = 66 disponible = true	Debería arrojar error en el ingreso del código del producto, solicitándole al usuario el ingreso nuevamente del mismo.	Al igual que en el caso 25, se permite el ingreso del código del producto con un valor entero negativo, pero el programa fue pensado bajo la idea de que el código de un producto se represente mediante valores enteros positivos, por ende, no se realiza la validación correspondiente.	Fallo	Validación ausente	Baja
26	No permitir continuar con la ejecución del programa luego de haber ingresado un valor igual a 3 cuando el usuario decide si el producto que está registrado, tendrá o no oferta.	decisionParaOferta = 3	Debería no permitir con la ejecución del programa, solicitándole al usuario el ingreso nuevamente de la opción.	Deja continuar con la ejecución del programa generando error en la carga de datos. Se rompe el programa.	Fallo	Validación ausente	Alta
27	Setear en NULL los atributos referidos a la oferta del producto, en caso de que el usuario decida no considerar el producto que está	decisionParaOferta = 2	Debería asignar de manera automática NULL los atributos que hacen referencia a la oferta del producto.	Los atributos de oferta del producto son nulos cuando no tiene oferta.	Ok	-	-

	cargando con oferta.						
28	Registrar un nuevo producto con descuento de oferta igual a 0,00.	descuentoOferta = 0,00	Debería arrojar error en el ingreso del descuento de oferta ya que no se estaría aplicando descuento al precio unitario del producto, ya que sería nulo. Es ilógico.	Permite el ingreso de un descuento de oferta igual a 0,00.	Fallo	Validación ausente	Media
29	Registrar un nuevo producto con descuento de oferta igual a un valor decimal negativo -0,10.	descuentoOferta = 0,10	Debería arrojar error en el ingreso del descuento de oferta ya que esta situación generaría error en el cálculo codificado para aplicarle la rebaja correspondiente al precio del producto.	Permite el ingreso de un descuento de oferta igual a -0,10.	Fallo	Validación ausente	Media
30	Registrar un nuevo producto con código formado por caracteres (letras).	<b>codigoProducto = A150</b> categoría = Termo modelo = Termo clasico 1.4 lts tamaño = 11 x 36 cm. color = Verde nombre = Termo original Stanley fechaIngreso = 20/15/2021 EstadoActual = false descripción = Termo de de acero precioUnitario = 13900 cantidadActual = 66 disponible = true	Debería arrojar error en el ingreso del código de producto ya que éste ha sido declarado como un dato del tipo entero.	No permite el ingreso de una cadena de caracteres alfanuméricos.	Ok	-	-

## Pruebas unitarias

Las pruebas unitarias fueron realizadas en el código fuente por lo que para su revisión se debe recurrir al mismo que se encuentra cargado en el repositorio git.

## CONCLUSIÓN:

Para finalizar con el desarrollo del trabajo práctico integrador y cierre del cursado de la cátedra, es fundamental visualizarse en el comienzo de la materia. Durante éste, se contaba con un nivel de conocimiento escaso sobre el tema “**programación**”.

Con el transcurso del cursado, se fueron alcanzando nuevos **conceptos y/o herramientas** para llevarla a cabo.

Es así que se eligió un **lenguaje de programación** para lograr enfocarse en el estudio del mismo. Contando con la libertad de elegir cualquiera de ellos, siempre y cuando haga uso del **paradigma orientado a objetos**.

La obtención de los nuevos conceptos fue gracias al trabajo realizado durante el primer cuatrimestre con las **entregas** de los distintos **trabajos prácticos** que fueron propuestos por la cátedra.

Sin embargo, cabe recalcar que no ha sido sencillo el aprendizaje del lenguaje, pero se fue logrando con el **constante estudio y perseverancia** puesta.

Este proceso permitió comenzar con el **trabajo práctico integrador final** de la materia. Para el **comienzo** de éste se contaba con más información que nos ayudó a la hora de su implementación. Sin embargo, también se presentaron distintas situaciones que llevaron a **pensar** más tiempo la solución de diseño de un determinado **requerimiento**.

Con el trabajo efectuado, considero que he logrado alcanzar la **solución** a cada uno de los requerimientos propuestos, tanto en la **primera iteración** como así también en la **segunda**. Sin embargo, como **autocrítica** podría remarcar que la **decisión de diseño** que adopté y puse en práctica no fue la mejor posible, ya que ésta presenta algunas irregularidades como por ejemplo la **falta de flexibilidad**, haciendo que el diseño no se adapte tanto a la realidad en la que podría implementarse o las **malas costumbres de programación** que se fueron tomando, generando que sea más compleja la lectura del código fuente.

Finalmente, como se estudió en la materia, toda decisión de diseño viene acompañada de alguna dificultad, ya que **más próximos a la realidad** estamos, más compleja será su implementación debido a que su **pensamiento** es más **completo**. Sin embargo, durante el desarrollo de este trabajo se intentó conseguir un “**equilibrio**” entre, los **conocimientos teóricos y prácticos**



aprendidos de diseño, y el **tiempo** que lleva la **implementación** de la realidad en la que podría desarrollarse.