

- *Maria Alejandra Mantilla Coral: A00395792*
- *Pablo Fernando Pineda Patiño: A00395831*

Formato de escenarios y casos de uso

Configuración de los Escenarios

Nombre	Clase	Escenario
setupStage1	InventoryTest	Un objeto de la clase Inventory con un objeto de la clase Product: <ul style="list-style-type: none"> - name="Product D", description="Description D", category=BEAUTY_AND_PERSONAL_CARE, price= 25.0, quantity= 4
setupStage2	InventoryTest	dos objetos de la clase Inventory con un objeto de la clase Product: <ul style="list-style-type: none"> - name="Product A", description="Description A", category=BOOKS, price=22.0, quantity=2 - name="Product B", description="Description B", category=BEAUTY_AND_PERSONAL_CARE, price=56.0, quantity=6
setupStage3	InventoryTest	Lista con los siguientes objetos de clase Order: <ul style="list-style-type: none"> - name="Alejandra", productsList=products1, total=30.0 el atributo productList de ... tiene los siguientes productos: <ul style="list-style-type: none"> - name="Product 1", description="The description 1", category=STATIONERY, price=10.0, quantity=5 - name="Pablo", productsList=products2, total=30.0 el atributo productList tiene los siguientes productos: <ul style="list-style-type: none"> - name="Product 2", description="The description 2", category=ELECTRONICS, price=15.0, quantity=7

Diseño de Casos de Prueba

Objetivo de la Prueba: Verificar que el inventario se actualiza correctamente al agregar o eliminar elementos y que la búsqueda de un producto arroja resultados correctos. También verifica que los métodos testLoadProducts y testLoadOrder permitan guardar y cargar productos y órdenes desde un archivo.				
Clase	Método	Escenario	Valores de Entrada	Resultado esperado
Inventory	addProduct	setupStage1	name="Product E", description="Description E", category= BOOKS, price= 50.0, quantity= 1	El inventario ahora tiene 2 productos, el primero con nombre "producto D" y el segundo "producto E"
Inventory	removeProduct	setupStage1	name="Product D"	El inventario ahora no tiene ningún producto
Inventory	searchProduct	setupStage2	keyWord="Product A"	se encontró un solo producto con ese keyWord
Inventory	searchProduct	setupStage2	keyWord="Description B"	se encontró un solo producto con ese keyWord
Inventory	searchProduct	setupStage2	keyWord="Category C"	no se encontró ningún producto con ese

				keyWord
Inventory	testLoadProducts	setupStage1	Cada atributo de cada uno de los objetos de clase Product registrados anteriormente en el inventory	Los productos registrados en el inventory son iguales a los que están guardados en el archivo json.
Inventory	testLoadOrder	setupStage3	Cada atributo de cada uno de los objetos de clase Order registrados anteriormente	Las órdenes registradas en la lista de órdenes son iguales a las que están guardadas en el archivo json.

Nombre	Clase	Escenario
setupStage1	OrderSearcherTest	<p>Objeto de clase OrderSearcher con una lista de objetos de clase Order la cual contiene:</p> <ul style="list-style-type: none"> - name="customer1", products=array vacío, total=2000 - name="customer2", products=array vacío, total=4000
setupStage2	OrderSearcherTest	<p>Se crean dos listas de productos(products1 y products2) cada uno respectivamente con un objeto de clase Product:</p> <ul style="list-style-type: none"> - name="product1", description="description1", category=SPORTS, price=10.0, quantity=15 - name="product2", description="description2", category=STATIONERY, price=5.0, quantity=12 <p>Ahora se crea una lista de ordenes(orders) la cual tiene los siguientes objetos de clase Order:</p> <ul style="list-style-type: none"> - name="customer1", products=products1, total=20.5 - name="customer2", products=products2, total=50.7

Objetivo de la Prueba: Verificar que la búsqueda mediante nombre del cliente y nombre del producto arrojen resultados correctos.

Clase	Método	Escenario	Valores de Entrada	Resultado esperado
OrderSearcher	SearchByCustomerName	setupStage1	name="customer1"	se encontró una sola orden con ese nombre de cliente
OrderSearcher	SearchByProduct	setupStage2	name="product1"	se encontró una sola orden la cual contiene ese producto con las siguientes características: name="product1", description="description1", category=SPORTS, price=10.0, quantity=15

Nombre	Clase	Escenario
setupStage1	OrderTest	<p>Un objeto de clase Order</p> <ul style="list-style-type: none"> - name="John Doe", productList=products, total=30.0 <p>el atributo productList(en este caso llamada products) del objeto de clase Order contiene una lista de objetos de clase Product:</p> <ul style="list-style-type: none"> - name="Product A", description="Description A", category=ELECTRONICS, price=10.0, quantity=7 - name=Product B", description="Description B", category=SPORTS, price=20.0, quantity=5

Objetivo de la Prueba: Comprobar que el constructor de la clase Order inicializa correctamente los atributos de la instancia a partir de los parámetros de entrada, y que los getters y setters de la clase permiten acceder y modificar adecuadamente los valores de los atributos de la instancia.

Clase	Método	Escenario	Valores de Entrada	Resultado esperado
Order	orderConstructor	setupStage1	name= "John Doe" productList=products total=30.0	El objeto Order se creó correctamente con los atributos anteriormente vistos en el setupStage1
Order	orderSettersAnd Getters	setupStage1	newId=1 newCustomerName="Jane Smith" productList=newProducts newProducts= -name="Product C", description="Description C", category=BOOKS, price=15.0, quantity=2 newTotal=15.0	El objeto Order se cambió correctamente con sus nuevos atributos y ahora es: id=1, name="Jane Smith", productList=newProducts, total=15.0

Nombre	Clase	Escenario
setupStage1	ProductSearcherTest	<p>Objeto de clase Inventory con los siguientes objetos de clase Product:</p> <ul style="list-style-type: none"> - name="product1", description="description1", category=ELECTRONICS, price=100, quantity=5 - name="product2", description="description2", category=BOOKS, price=50, quantity=3; - name="product3", description="description3", category=ELECTRONICS, price=200, quantity=2

Objetivo de la Prueba: Verificar que los métodos de la clase permiten buscar productos por diferentes criterios, como el nombre o una palabra clave, la categoría, y el precio.

Clase	Método	Escenario	Valores de Entrada	Resultado esperado
ProductSearcher	testSearchByCategory	setupStage1	category=ELECTRONICS	se encontraron 2 objetos de clase Product con esta categoría, el nombre de estos objetos son respectivamente: product1 y product3
ProductSearcher	testSearchByKeyword	setupStage1	"description"	Se encontraron 3 coincidencias de búsqueda en las descripciones de los productos.
ProductSearcher	testSearchByPrice	setupStage1	minprice=75 maxprice=150	Se encontró un solo objeto de clase Product en este rango de precios, el nombre del objeto es product1

Nombre	Clase	Escenario
setupStage1	ProductTest	un objeto de tipo Product con los siguientes atributos: name="Test Product", descrip="A product for testing purposes", category=ELECTRONICS, price=999.99, quantity=10

Objetivo de la Prueba: Verificar que los métodos de la clase funcionan correctamente y que los atributos de la clase se asignan y validan correctamente.

Clase	Método	Escenario	Valores de Entrada	Resultado esperado
Product	testGetName	setupStage1	name="Test Product"	El String es igual al atributo name del producto ya registrado, esto indica que se pudo haber guardado correctamente.
Product	testGetDescription	setupStage1	descrip="A product for testing purposes"	El String es igual al atributo description del producto ya registrado, esto indica que se pudo haber guardado correctamente.
Product	testGetCategory	setupStage1	category=ELECTRONICS	La categoría(enum) es igual al atributo category del producto ya registrado, esto indica que se pudo haber guardado correctamente.
Product	testGetPrice	setupStage1	price=999.99	El double es igual al atributo price del producto ya registrado, esto indica que se pudo haber guardado correctamente.
Product	testGetQuantity	setupStage1	quantity=10	El int es igual al atributo quantity del producto ya registrado, esto indica que se pudo haber guardado correctamente.
Product	testSetName	setupStage1	newName="new Name"	El nombre del objeto de clase Product ya registrado ahora es new Name
Product	testSetDescription	setupStage1	newDesc="new Description"	La descripción del objeto de clase Product ya registrado ahora es new Description
Product	testSetCategory	setupStage1	newCategory=CLOTHING_ACCESSORIES	La categoría del objeto de clase Product ya registrado ahora es CLOTHING_ACCESSORIES
Product	testSetPrice	setupStage1	newPrice=1499.99	El precio del objeto de clase Product ya registrado ahora es 1499.99
Product	testGetQuantity	setupStage1	newQuantity=20	La cantidad del objeto de clase Product ya registrado ahora es 20