```
----------------------------------
|       Salable Product        |
----------------------------------
| - name: String              |
| - description: String        |
| - price: double             |
| - quantity: int             |
----------------------------------
| + getName(): String          |
| + getDescription(): String    |
| + getPrice(): double          |
| + getQuantity(): int          |
| + setName(name: String): void   |
| + setDescription(desc: String): void |
| + setPrice(price: double): void |
| + setQuantity(quantity: int): void |
```
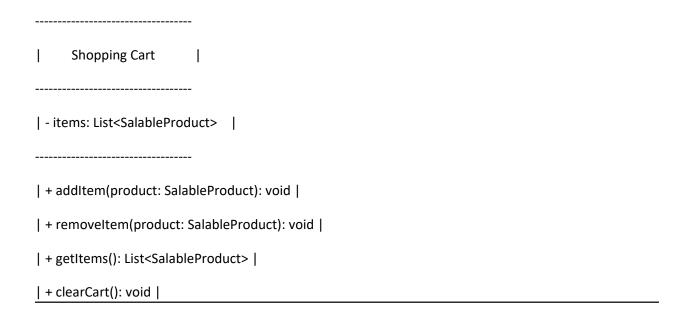
Based on the requirements, the Salable Product should have attributes such as Name, Description, Price, and Quantity.

```
---------------------------------
|        Store Front          |
---------------------------------
| - inventoryManager: InventoryManager |
| - shoppingCart: ShoppingCart   |
---------------------------------
| + StoreFront(inventoryManager: InventoryManager) |
| + initializeStore(): void      |
| + purchaseProduct(product: SalableProduct): void |
| + cancelPurchase(product: SalableProduct): void |
```

The Store Front should support initializing the state of the Store, purchasing a Salable Product, and canceling the purchase of a Salable Product.

The Store Front class has a composition relationship with the Inventory Manager and Shopping Cart classes. It has methods to initialize the store, purchase a Salable Product, and cancel a purchase.

```
---------------------------------
|      Inventory Manager      |
---------------------------------
| - products: List<SalableProduct>|
---------------------------------
| + addProduct(product: SalableProduct): void |
| + removeProduct(product: SalableProduct): void |
| + getProduct(name: String): SalableProduct |
| + getAllProducts(): List<SalableProduct> |
```

The Inventory Manager is responsible for managing the inventory of Salable Products

The Inventory Manager class has a list of Salable Products and provides methods to add, remove, and retrieve products from the inventory.

```
----------------------------------

|        Shopping Cart        |

----------------------------------

| - items: List<SalableProduct>    |

----------------------------------

| + addItem(product: SalableProduct): void |

| + removeItem(product: SalableProduct): void |

| + getItems(): List<SalableProduct> |

| + clearCart(): void |
```

The Shopping Cart is responsible for managing the selected products for purchase.

The Shopping Cart class has a list of selected Salable Products and provides methods to add, remove, retrieve items, and clear the cart.