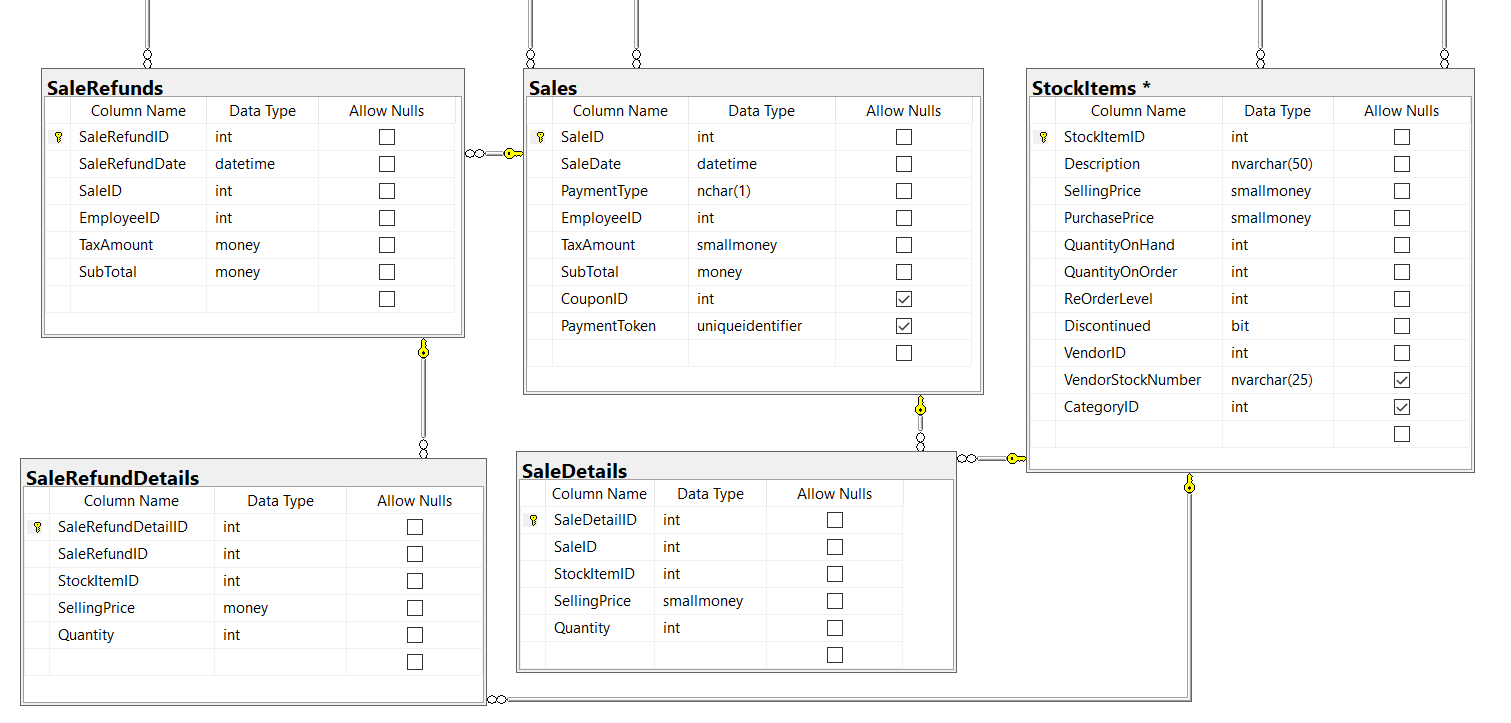
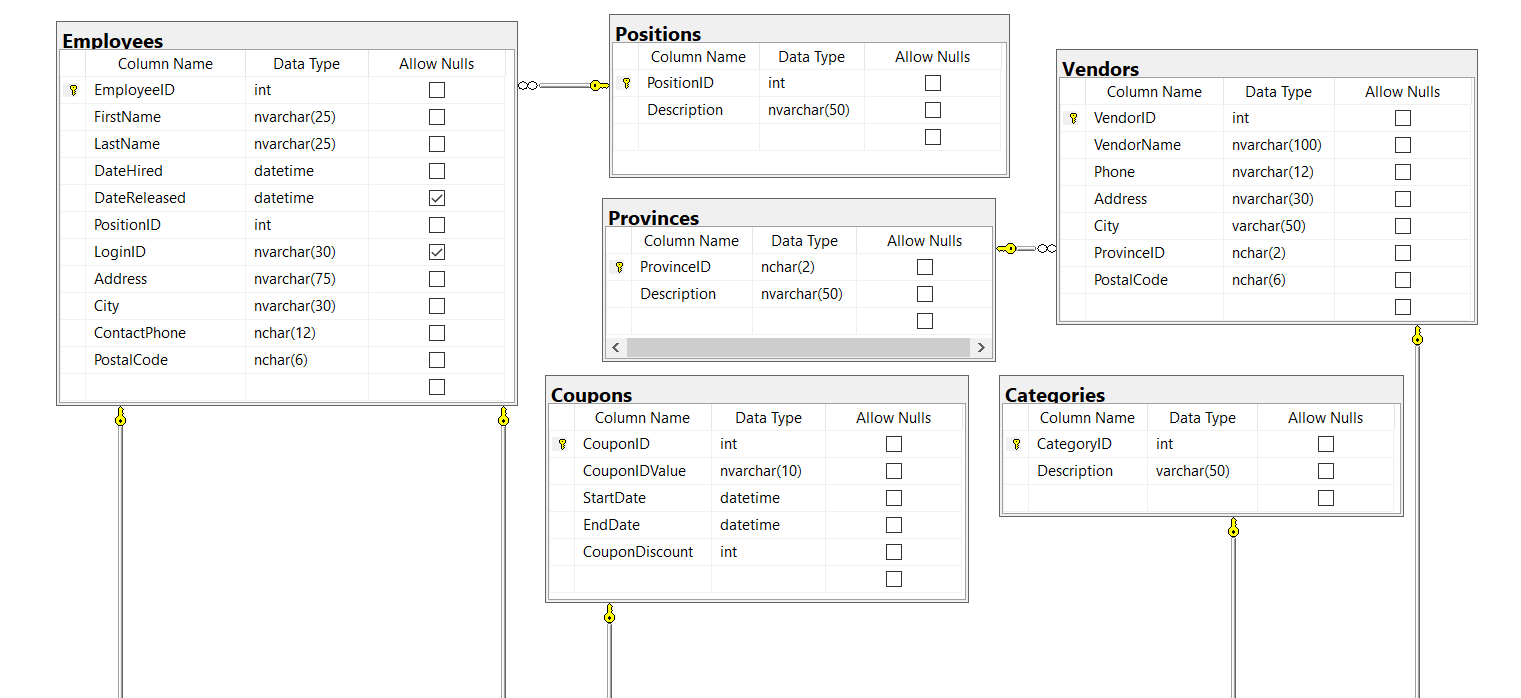
Sales Subsystem



# ERD for UX process

# Possible UX Interface

Place your screen design(s) here

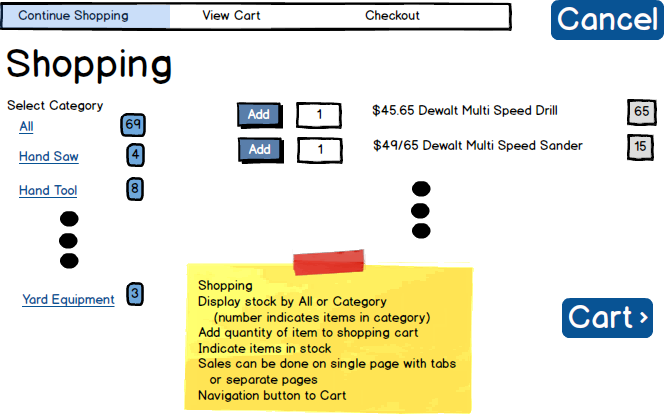
**Overview**: This is a 2 pages project which consist of “Sales Order” page and “Sales Return” page.

Figure 1 This is the “Sales Order” web page. The view is under “Continue Shopping” tab.

Table 1 This task will basically just cancel the transaction.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CancelBtn (4) | | Cancel\_OnClick   * Confirmation message about data loss * Cancel Stock items in gridview * Refresh StockItemsGV |
| **BLL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
|  | |  |

Table 2 This task will go to "View Cart" tab. Can be done by clicking “Cart” button or “View Cart” tab.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CartLinkBtn (5) OR  ViewCart (6) | | Cart\_OnClick   * Redirect to the “View Cart” tab |
| **BLL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
|  | |  |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CategoriesGV (1) | | Page\_Load / wired via ODS   * All and the items in category will display on page load. * Bind data to Categories |
| **BLL Class(es) and Method (s)** | | |
| CategoriesController | List<CategoriesList> List\_CategoriesForStockItems( string cat, int catCount)   * Retrieve a list of categories for stock items * Return category description (Categories.Description) and the count of category ID (Categories.StockItems.CategoryID) | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| Categories (R), StockItems (R) | | CategoriesList |

Table 3 This task will display the categories on page load. User can click on each category.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CategoriesGV (1)  StockItemsGV (2) | | OnClick / wired via ODS   * Move selection value which identifies category id (hidden field) * Bind data to StockItems |
| **BLL Class(es) and Method (s)** | | |
| StockItemsController | List<StockItemsList> List\_StockItems( int categoryId)   * Retrieve a list of stock items from arguments int * Return selling price (StockItems.SellingPrice), item description (StockItems.Description), and quantity on hand (StockItems.QuantityOnHand) | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| StockItems (R) | | StockItemsList |

Table 4 This task will display StockItemGV when user clicks on the categories under CategoriesGV.

Table 5 This task will add whatever quantity user entered. Item details will be displayed on "View Cart" tab. Confirmation message will appear when user entered a quantity more than what is on-hand.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| AddBtn(3)  ItemQuantityTextBox(7) | | OnClick   * Collect stock item information (StockItemID) * Validate the quantity * Send to BLL to process the items and quantity |
| **BLL Class(es) and Method (s)** | | |
| StockItemsController | void Add\_ItemsToSales(string item, string sale)   * TRX (all errors to be returned) * Create SaleDetailID | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| StockItems (C), SaleDetails (C), Sales (C) | |  |

Table 6 ViewModel definition of CategoriesList

|  |  |  |
| --- | --- | --- |
| **CategoriesList** | | |
| Description | string | Categories |
| CategoryCount | int | StockItems |

Table 7 ViewModel definition of StockItemsList

|  |  |  |
| --- | --- | --- |
| **StockItemsList** | | |
| SellingPrice | decimal | StockItems |
| Description | string | StockItems |
| QuantityOnHand | int | StockItems |

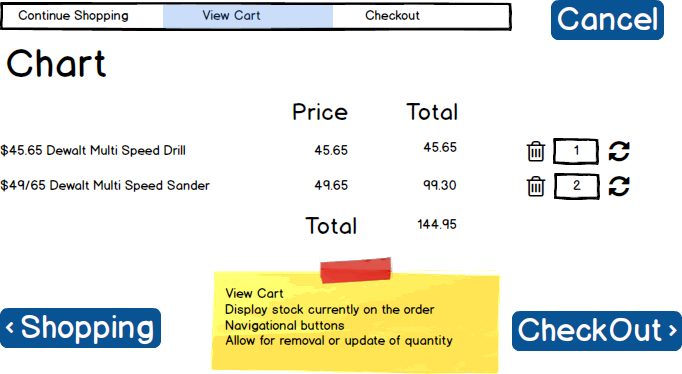


Figure 2 This is the “Sales Order” web page. The view is under “View Cart” tab

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CancelBtn (1) | | Cancel\_OnClick   * Confirmation message about data loss * Cancel the transaction * Refresh OrderGV |
| **BLL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
|  | |  |

Table 8 This task will cancel the transaction.

Table 9 This task will go to "Checkout" tab. Can be done by clicking “Checkout” button or “Checkout” tab.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CheckoutLinkBtn (2) OR  Checkout (3) | | Checkout\_OnClick   * Redirect to the “Checkout” tab |
| **BLL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
|  | |  |

Table 10 This task will go back to "Continue Shopping" tab. Can be done by clicking "Shopping" button or "Continue Shopping" tab.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| ShopLinkBtn (4) OR  Shop (5) | | Shop\_OnClick   * Redirect to the “Continue Shopping” tab |
| **BLL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
|  | |  |

Table 11 This task will display the order(s) in a gridview (GV). The girdview will display right away when user decided to go to “View Cart” tab.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| OrderGV (6) | | wired via ODS   * Move selection value which identifies order id (hidden field) * Bind data to Orders |
| **BLL Class(es) and Method (s)** | | |
| OrdersController | List<OrdersList> List\_OrderItems( int orderId)   * Retrieve list of order(s) * Return description (StockItems.Description), price (SaleDetails.SellingPrice), subtotal (Sales.SubTotal), and total (sum of all the subtotal) | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| StockItems (R), SaleDetails (R), Sales (R) | | OrdersList |

Table 12 This task will delete the entire instance of order.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| DeleteBtn (7), OrderGV (6) | | OnClick   * Warning message for removing the order. * Send to BLL for removing the item(s) in the order. * Refresh OrderGV |
| **BLL Class(es) and Method (s)** | | |
| OrdersController | void Remove\_ItemsToSales (string item, int orderId)   * TRX (all errors to be returned) * Remove orders | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| StockItems (R), SaleDetails (U/D), Sales (U/D) | |  |

Table 13 This task will update the subtotal and total price when user update the OrderQuantityTextBox

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| RefreshBtn (9)  OrderQuantityTextBox (8) | | OnClick   * Get the orderId and do the calculation * Validation * calculate when less than QuantityOnHand * warning message when more than QuantityOnHand * Send to BLL for processing the calculation |
| **BLL Class(es) and Method (s)** | | |
| OrdersController | void Update\_ItemsToSales ( string item, int orderId)   * TRX (all errors to be returned) * Do the calculation for subtotal and total | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| Sales (U) | |  |

Table 14 ViewModel definition of OrdersList

|  |  |  |
| --- | --- | --- |
| **OrdersList** | | |
| Description | string | StockItems |
| Price | decimal | SaleDetails (SellingPrice property) |
| SubTotal | decimal | Sales |
| Total (sum of all SubTotal) | decimal | Sales |

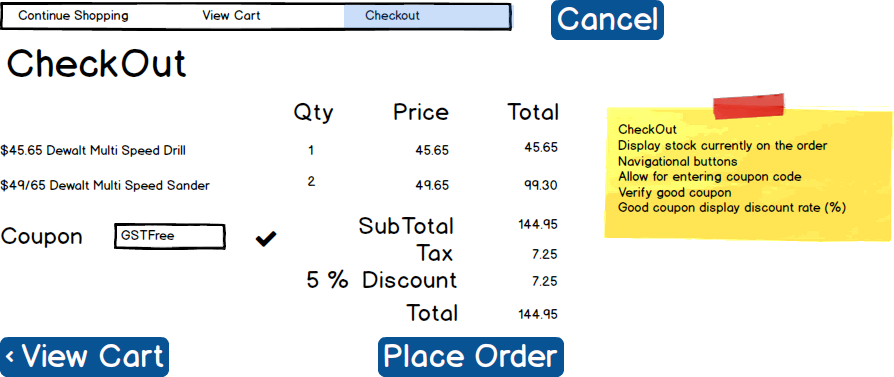


Figure 3 This is the “Sales Order” web page. The view is under “Checkout” tab.

Table 15 This task will cancel the transaction.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CancelBtn (1) | | Cancel\_OnClick   * Confirmation message about data loss * Cancel the transaction * Refresh CheckoutGV |
| **BLL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
|  | |  |

Table 16 This task will let user go back to "Continue Shopping" tab.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| Shop (2) | | Shop\_OnClick   * Redirect to the “Continue Shopping” tab |
| **BLL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
|  | |  |

Table 17 This task will let user go back to "View Cart" tab

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CartLinkBtn (4) OR  ViewCart (3) | | Cart\_OnClick   * Redirect to the “View Cart” tab |
| **BLL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
|  | |  |

Table 18 This task will display the final order(s) in a gridview (GV). The gridview will display right away when user decided to go to “Checkout” tab.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CheckoutGV (5) | | wired via ODS   * Move final order(s) |
| **BLL Class(es) and Method (s)** | | |
| OrdersController | List<CheckoutList> List\_CheckoutItems(int orderId, int quantity)   * Retrieve list of order(s) from int argument * Return description (StockItems.Description), quantity (SaleDetails.Quantity), price (SaleDetails.SellingPrice), subtotal of each orderId (quantity x price), subtotal of all orderId (sum of all orderId), tax (Sales.TaxAmount), discount (Coupons.CouponDiscount), total (includes tax and discount) | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| StockItems (R), SaleDetails (R), Sales (R) | | CheckoutList |

Table 19 This task is to display coupon validity (if available).

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CouponTextBox(6) | | OnClick   * Display message if invalid coupon * Accept coupon if valid |
| **BLL Class(es) and Method (s)** | | |
| OrderCouponsController | void Check\_CouponDetails (int couponId)   * Check validity of coupon base on EndDate | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| Coupons (R) | |  |

Table 20 This task will place the order.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| PlaceOrderBtn(7) | | OnClick   * Validation message to place the order. * Send to BLL to add saleId for this order |
| **BLL Class(es) and Method (s)** | | |
| OrderSalesController | void Add\_AddToSale (int saleId)   * Generate saleId as well as Sale Number * Sale number will increment by 1 everytime order is place | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| Sales (C) | |  |

Table 21 ViewModel definition of CheckoutList

|  |  |  |
| --- | --- | --- |
| **CheckoutList** | | |
| Description | string | StockItems |
| Quantity | int | SaleDetails |
| Price | decimal | SaleDetails |
| OrderSubTotal | decimal | Sales (quantity x price) |
| AllOrderSubTotal | decimal | Sales (sum of all order) |
| TaxAmount | decimal | Sales |
| Discount | decimal | Coupons |
| Total | decimal | Sales |

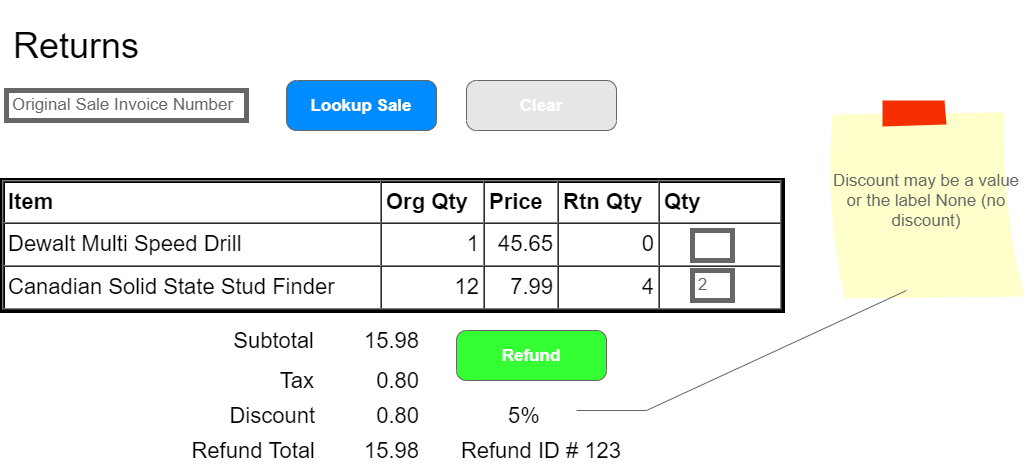


Figure 4 This is the “Sales Return” web page.

Table 22 This task will lookup SaleRefundID and will display the information on SaleReturnGV

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| SaleReturnTextBox (1), SaleBtn (2)  SaleReturnGV (4) | | OnClick / wired via ODS   * Move selection value to SaleRefundId (hidden field) * Bind data to SaleReturnGV |
| **BLL Class(es) and Method (s)** | | |
| OrdersController | List<RefundList> List\_RefundItems (int saleId)   * Retrieve list of refund items * Return description (StockItems.Decription). order quantity (SaleDetails.Quantity), Price (SaleDetails.SellingPrice), and return quantity (SaleRefundDetails.Quantity) * sales subtotal (Sales.SubTotal), tax (Sales.TaxAmount), discount (Coupons.CouponDiscount), and refund | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| StockItems (R), SaleDetails (R), SaleRefundDetails (R) | | RefundList |

Table 23 This task will clear lookup result.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| ClearBtn (3) | | Clear\_OnClick   * Confirmation message about data loss * Clear SaleReturnTextBox and SaleReturnGV |
| **BLL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
|  | |  |

Table 24 This task will display refund details when user enter the quantity and click refund button.

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| ReturnQuantityTextBox (5)  RefundBtn (6), RefundGV (7) | | OnClick / wired via ODS   * Validation message for refund confirmation. |
| **BLL Class(es) and Method (s)** | | |
| OrdersRefundController | Void Add\_SaleRefund(int saleRefund, int couponId, int saleId)   * Generate SaleRefundID * Calculate subtotal and refund total base on quantity enetred | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModel** |
| Sales (U), Coupons (U), SaleRefunds (C) | |  |

Table 25 ViewModel definition of RefundList

|  |  |  |
| --- | --- | --- |
| **RefundList** | | |
| Description | string | StockItems |
| OrderQuantity | int | SaleDetails |
| Price | int | SaleDetails |
| ReturnQuantity | int | SaleRefundDetails |