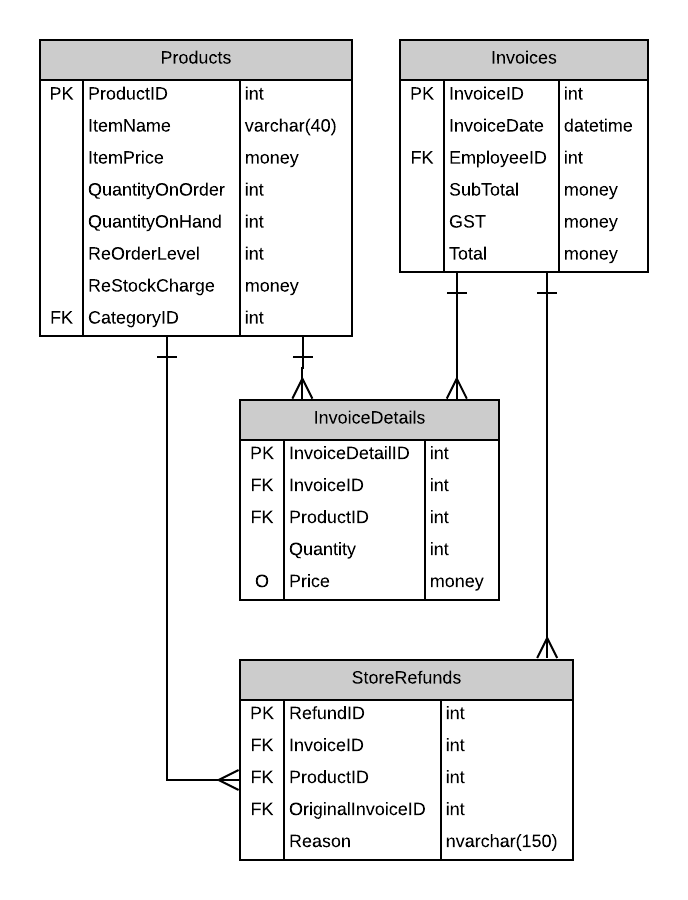
Sales Sub-System

# ERD for Sales Sub-System UX process



# Possible UX Interface

## Instore Sales

# 

## Refunds/Returns

# 

# Event and Wiring Summations

### In-Store Sales

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CategoryDDL | | Wired via ODS |
| **BBL Class(es) and Method (s)** | | |
| CategoryController(R) | List<SelectionList> List CategoryList ()   * retrieve a list of Descriptions and their Category ID for DDL | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModels** |
| Category(R) | | SelectionList |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| ProductDDL | | OnCommand/Wired via ODS   * Collect the categoryid and validate * Issue call to get Product info * Data bind to display items |
| **BBL Class(es) and Method (s)** | | |
| ProductController(R) | List<SelectionList> List findbyCategoryID (int categoryid)   * retrieve a list of products and their Product ID for DDL related to the categoryid supplied | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModels** |
| Category(R) | | SelectionList |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| Add Button,  InvoiceDetailGridView, QtyArg  ProductDDL | | OnClick/Wired via ODS   * get product item information (productid) and validate * Quantity textbox value to QtyArg * send to BLL to add to Product List * Databind to display in products in GridView |
| **BBL Class(es) and Method (s)** | | |
| InvoiceDetailController  InvoiceController | Void Add\_DetailToInvoice(int productid, string QtyArg)  TRX   * Create invoice if needed, set itemnumber to 1, * Invoice exists, find current highest itemnumber to increase by 1, * Verify product is not already added to invoice. If so, add the QtyArg to the Quantity field * Add to the InvoiceDetail | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModels** |
| InvoiceDetail(C), Invoice(C optional) | | InvoiceItemList |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| Refresh Button  InvoiceDetailGridView | | OnClick   * Validate quantity and productid * Collect and package values * Send to BBL to update values * Refresh the ItemDetails |
| **BBL Class(es) and Method (s)** | | |
| InvoiceDetailController  InvoiceController | void Update\_InvoiceDetailQty (int invoicedetailid, int qtyarg, List<InvoiceDetailList> invoicelist)  TRX   * Validate the data requirements * Ensure quantity values are valid * Update ItemDetails (qty, amount)   Update invoice (int invoiceid, List<InvoiceItemList> items)   * Update (Subtotal, total, tax) | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModels** |
| InvoiceDetail(U) | | InvoiceItemList |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| Delete Button | | OnClick   * Collect the invoice detail id and validate * Send to BLL to remove item from invoice detail * Refresh the invoice detail list |
| **BBL Class(es) and Method (s)** | | |
| InvoiceDetailController  Invoice Controller | void Delete\_ItemFromInvoice (int invoicedetailid, List<int> itemnumber)  TRX   * Find invoice item * Remove item * Renumber the items remaining in the list   Update invoice (int invoiceid, List<InvoiceItemList> items)   * Update (Subtotal, total, tax) | |
| **SQL Table(s): (C, R, U, D)** | | **ViewModels** |
| Invoice (R, U), OrderDetails (U/D) | |  |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| Clear cart Button | | On Click   * Clear form |
| **BBL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModels** |
|  | |  |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| Payment Button | | OnClick   * Validate Invoicenumber, invoicedetailid and quantities * Collect and package data * Send to BLL |
| **BBL Class(es) and Method (s)** | | |
| InvoiceController,  InvoiceDetailController | void Add\_InvoicePayment (int invoice number, List<InvoiceItemList> invoiceitems  TRX   * Validate all data requirements * Verify all items have valid quantities * OrderDetail Update for each item (quantity)   Update invoice (int invoiceid, List<InvoiceItemList> items)   * Update fields (Subtotal, total, tax) | |
| **SQL Table(s): (C, R, U, D)** | | **ViewModels** |
| Invoice (C, U), InvoiceDetail (R, U) | | InvoiceItemsList |

### Refunds

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| Lookup Invoice, InvoiceNumber,  RefundListGridView | | On Click   * Verify invoice number is entered * Call for form information * Display invoiceList items (Hidden invoiceid on each row) |
| **BBL Class(es) and Method (s)** | | |
| StoreRefundController  InvoiceDetailController  InvoiceController  ProductController | RefundInvoice Get\_Invoice (int invoiceid, List<InvoiceItemList> items)   * Check if StoreRefunds exist with OriginalInvoiceID same as invoiceid * If exists, find the attached invoice   Get refundItems attached to that invoice  (productid, qty, price, amount, restockcharge, refundreason)   * If not exists, create new invoice   Get List of RefundItems (productname, qty, Price, Amount, restockcharge, reason) | |
| **SQL Table(s): (C, R, U, D)** | | **ViewModels** |
| StoreRefunds (C opt, U), InvoiceDetail (R), Invoice(C) | | InvoiceItemsList, StoreRefundList, InvoiceList |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| Refund Button  RefundListGridView | | OnCommand   * Validate required fields (restockcharge, reason, invoiceitemid(hidden)) * Collect and package values * Send to BBL to update values * Refresh and display the RefundItems * Display RefundInvoiceNumber (new invoiceID) |
| **BBL Class(es) and Method (s)** | | |
| InvoiceDetailController  InvoiceController  ProductController | void Apply\_Refund (int invoicedetailid, int restock, string reason, List<InvoiceItemList> refunditems)  TRX   * Validate the data requirements * Verify refunditem has not been refunded before (restockcharge and/or reason values are null) * If does not exist, update restock and reason * Update ItemDetails (restockcharge, reason) * Remove refunditems not updated (restockcharge and reason fields still null)   Update invoice (int invoiceid, List<InvoiceItemList> items)   * Update (Subtotal, total, tax)   Update\_Product (int productid, int qty)   * Update QuantityOnHand by adding qty | |
| **SQL Table(s): (C, R, U, D)** | | **ViewModels** |
| InvoiceDetail (C, U), Product(U), Invoice (C, U opt) | | InvoiceItemList, ProductList, InvoiceList |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| Clear Button | | On Click   * Clear form |
| **BBL Class(es) and Method (s)** | | |
|  |  | |
| **SQL Table(s): (C,R,U,D)** | | **ViewModels** |
|  | |  |

# ViewModels:

## SelectionList

* int IDValue
* string DisplayText

## InvoiceItemList

* int InvoiceDetailID
* int ProductID
* int Quantity
* decimal? Price

## StoreRefundList

* int RefundID
* int InvoiceID
* int ProductID
* int OriginalInvoiceID
* string Reason

## ProductList

* int ProductID
* string ItemName
* decimal? ItemPrice
* int QuantityOnHand
* decimal? ReStockCharge
* int CategoryID

## InvoiceList

* int InvoiceID
* localDate InvoiceDate
* int EmployeeID
* decimal? SubTotal
* decimal? GST
* decimal? Total