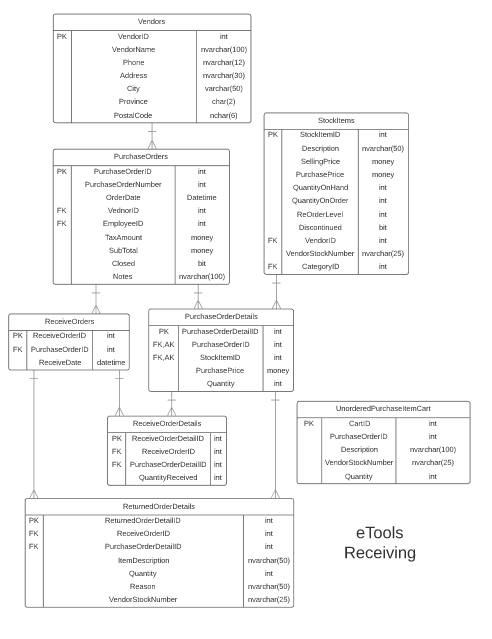
Receiving

# ERD for UX process

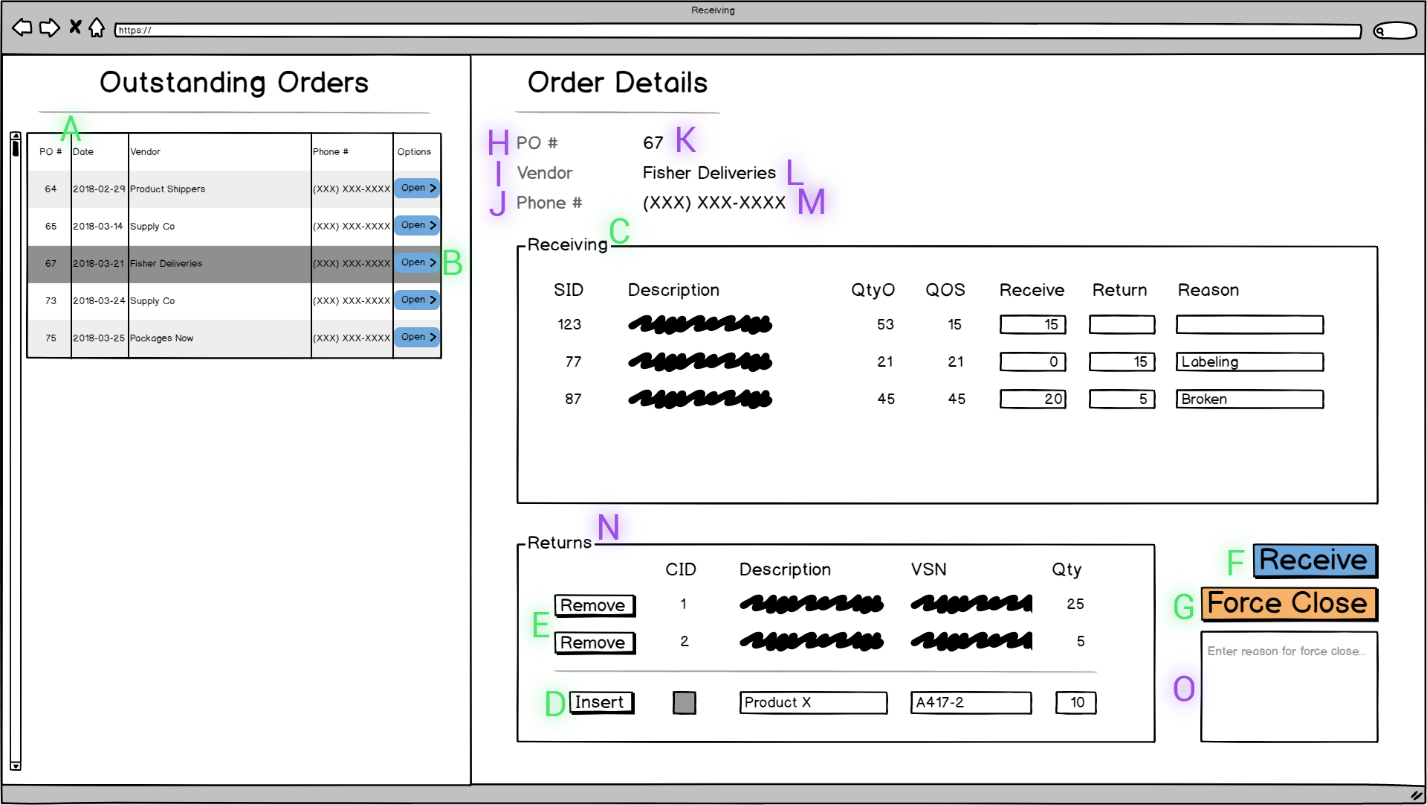


# Possible UX Interface

*User View*



*Controls Marked*



*Control Notes*

A Control

*Name:* OrderGrid

*Type:* GridView

*Entity:* OpenOrder (P)

*Columns:*

* PurchaseOrderID (is hidden)
* PurchaseOrderNumber (shown as: PO #)
* OrderDate (shown as: Date)
* VendorName (shown as: Vendor)
* Phone (Shown as: Phone #)

B Control

*Name:* OrderButton

*Type:* Button

C Control

*Name:* ReceivingList

*Type:* ListView

*Entity:* OpenOrderDetails (P)

*Columns:*

* PurchaseOrderID (is hidden)
* StockItemID (shown as: SID, read only)
* Description (read only)
* QuantityOrdered (shown as: QtyO, read only)
* QuantityOutstanding (shown as: QOS, read only)
* QuantityReceived (shown as: Receive, write enabled)
* QuantityReturned (shown as: Return, write enabled)
* ReturnReason (shown as: Reason, write enabled)

D Control

*Name:* InsertButton

*Type:* Button

E Control

*Name:* RemoveButton

*Type:* Button

F Control

*Name:* ReceiveButton

*Type:* Button

G Control

*Name:* ForceCloseButton

*Type:* Button

H Control

*Name:* PONumberLabel

*Type:* Label

I Control

*Name:* VendorNameLabel

*Type:* Label

J Control

*Name:* PhoneLabel

*Type:* Label

K Control

*Name:* PONumberText

*Type:* TextView

L Control

*Name:* VendorNameText

*Type:* TextView

M Control

*Name:* PhoneText

*Type:* TextView

N Control

*Name:* ReturnList

*Type:* ListView

*Entity:* UnorderedPurchaseItemCart (P)

*Columns:*

* CartID (shown as: CID, read only)
* Description (write enabled)
* VendorStockNumber(shown as: VSN, write enabled)
* Quantity (shown as: Qty, write enabled)

O Control

*Name:* ForceCloseText

*Type:* TextView

# BLL Designs

A Control

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| OrderGrid | | None / Wired via ODS |
| **BLL Class(es) and Method (s)** | | |
| OpenOrderController | List<OpenOrder> OpenOrder\_List()  *Retrieve a list of outstanding purchase orders for display in a grid view* | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| PurchaseOrders (R) > PONumber, OrderDate, POID  Vendors (R) > VendorName, Phone | | OpenOrder (D) |

B Control

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| OrderButton  PONumber, VendorName, Phone | | OnClick   * Call BLL * Bind returned data to controls in B & C |
| **BLL Class(es) and Method (s)** | | |
| OpenOrderController | OpenOrder OpenOrder\_FindByID(int purchaseOrderID)  *Retrieve an instance of an outstanding purchase order for display via 3 labels and the purchase order details* | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| PurchaseOrders (R) > PONumber, OrderDate  Vendors (R) > VendorName, Phone | | OpenOrder (D)  PurchaseOrder (E) |

C Control

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| ReceivingList  QuantityReceived, QuantityReturned, ReturnReason | | OnClick (OrderButton)   * Call BLL * Bind returned data to a ListView (ReceivingList) |
| **BLL Class(es) and Method (s)** | | |
| PurchaseOrderDetailsController | List<OpenOrderDetails> OpenOrderDetails\_List(int purchaseOrderID)  *Retrieve a list of outstanding order details for display in a list view, does so when the above OrderButton event occurs* | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| PurchaseOrderDetails (R) > StockItemID, Quantity, POID  StockItems (R) > Description  ReceiveOrderDetails (R) > QuantityReceived  *Note: QuanityOutstanding derived from Quantity - Received* | | OpenOrderDetails (P)  PurchaseOrderDetails (E)  ReceiveOrderDetails (E) |

D Control

|  |  |
| --- | --- |
| InsertButton  ReturnList | OnClick   * Validate data (Description, VSN, Qty) * UPIC instance created and held **temporarily** in a list (ReturnList) |
| **BLL Class(es) and Method (s)** | |
| N/A | N/A |
| **SQL Table(s): (C,R,U,D)** | **Entities/DTOs/POCOs** |
| N/A | UnorderedPurchaseItemCart **(UPIC)** (P) |

E Control

|  |  |
| --- | --- |
| RemoveButton  ReturnList | OnClick   * UPIC instance removed from **temporary** list (ReturnList) |
| **BLL Class(es) and Method (s)** | |
| N/A | N/A |
| **SQL Table(s): (C,R,U,D)** | **Entities/DTOs/POCOs** |
| N/A | UnorderedPurchaseItemCart **(UPIC)** (P) |

F Control

|  |  |  |
| --- | --- | --- |
| ReceiveButton  ReceivingList, ReturnList, OrderGrid | | OnClick   * Bulk process ReceivingList by sending data to BLL method * Bulk process ReturnList by sending data to BLL method * Refresh ReceivingList * Empty ReturnList * Refresh OrderGrid |
| **BLL Class(es) and Method (s)** | | |
| PurchaseOrderDetailsController | void PurchaseOrderDetails\_Receive(  int poID,  List<OpenOrderDetails> OODs,  List<UnorderedPurchaseItemCart> UPIs)  *TRX:*   * *Create new ReceiveOrder record when at least one item is received, returned, or unordered* * *Create new ReceiveOrderDetails record for an item being received, called for each detail record* * *StockItem QuantityOnHand increased by the received quantity (each detail row)* * *StockItem QuantityOnOrder decreased by the received quantity (each detail row)* * *Create new ReturnedOrderDetails record for returned quantities or if there are any unordered items in the cart* * *On PurchaseOrder record set closed to true (only if all purchase order items are fully received). Done when each OpenOrderDetails QuantityOrdered field is checked against the sum of all the ReceiveOrderDetails records QuantityReceived field.* | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| PurchaseOrders (U *If PO is closed*) > Closed  StockItems (U) > QuantityOnHand, QuantityOnOrder  ReceiveOrders (C)  ReceiveOrderDetails (C)  ReturnedOrderDetails (C) | | UnorderedPurchaseItemCart **(UPIC)** (P)  OpenOrder (D)  OpenOrderDetails(P)  PurchaseOrder (E)  ReceiveOrder (E)  ReceiveOrderDetails (E)  ReturnedOrderDetails (E)  StockItem (E) |

G Control

|  |  |
| --- | --- |
| ForceCloseButton  ReceivingList, ReturnList, OrderGrid | OnClick   * Update PO as closed * Clear ReceivingList * Clear ReturnList * Refresh OrderGrid |
| **BLL Class(es) and Method (s)** | |
| PurchaseOrderController | void PurchaseOrder\_ForceClose(  int poID,  List<OpenOrderDetails> OODs)  *TRX:*   * *On PurchaseOrder record set closed to true* * *StockItem QuantityOnOrder updated by subtracting outstanding amount for each detail row* |
| **SQL Table(s): (C,R,U,D)** | **Entities/DTOs/POCOs** |
| PurchaseOrders (U) > Closed  StockItems (U) > QuantityOnOrder | PurchaseOrder (E)  StockItem (E)  OpenOrderDetails (P) |

Comments

Nice job on creating your screens and noting controls. The only concern I have is with the UnorderedPurchaseItemCart Insert and delete.