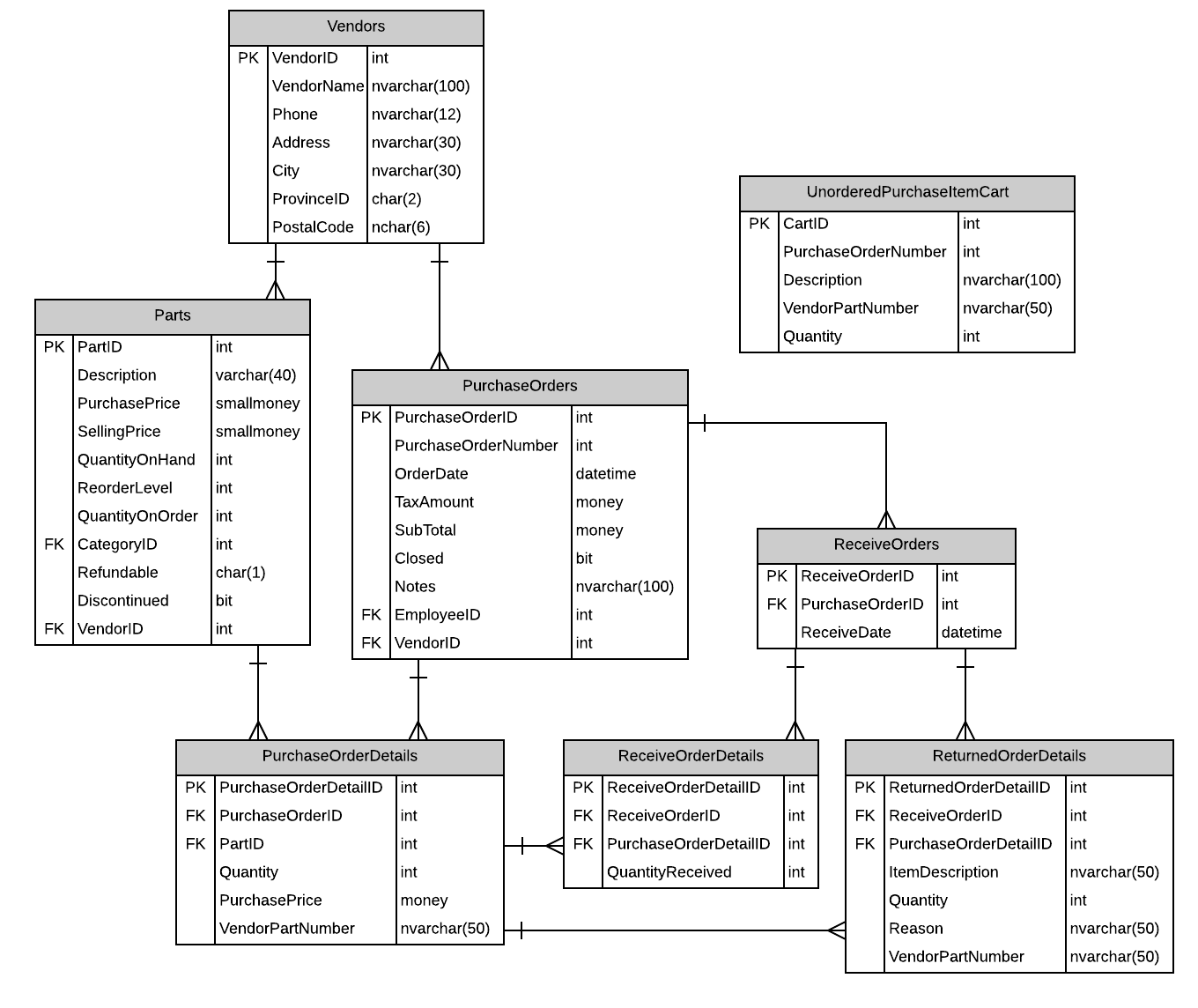
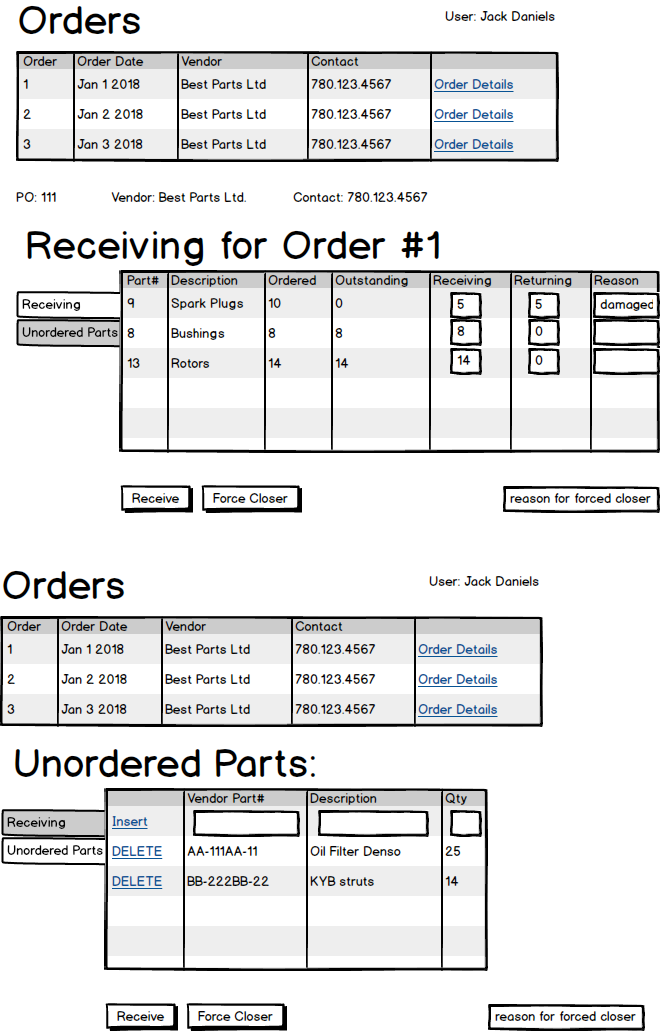
SubSystem title

# ERD for UX process

Place your ERD diagram here.



# Possible UX Interface



# Event and Wiring Summations

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| **OutsandingOrdersLIST (ListView)**(PO Number, Order Date, Vendor Name, Vendor Phone, PurchaseOrderID(hidden)) | | Wired via ObjectDataSource |
| **BBL Class(es) and Method (s)** | | |
| PurchaseOrderController | List<OutstandingPOList>OutstandingPO\_GET()   * Retrieves a list of Outstanding Purchase Orders. * Returns (PO Number, Order Date, Vendor Name, Vendor Phone, PurchaseOrderID(hidden)) | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| PurchaseOrders(R), Vendors(R) | | OutstandingPOList(P) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| **ViewOrder(link), PO(label), Contact(label), Vendor(label)**  **PODetailsList(ListView)**(PurchaseOrderID, PartID, PartDescription, Quantity On Order, Quantity Outstanding) | | OnClick/ wired via ObjectDataSource   * Validate data * Send poID to BLL(PurchaseOrderDetailsController) * Bind return data to |
| **BBL Class(es) and Method (s)** | | |
| PurchaseOrderDetailsController | List<PurchaseOrderDetails >PODetails\_GET (int poID)  Gets information for the specific purchase order including (PurchaseOrderID, PartID, PartDescription, Quantity On Order, Quantity Outstanding) | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| PurchaseOrderDetails(R), Parts(R), | | PurchaseOrderDetails(P) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| **Receive(button) Receiving(txt), Returning(txt), Reason(txt)** | | OnClick   * Validates Data * Gets receive(int), Return(int), Reason(str) from PODetailsList * Send to BLL for update * Clear fields |
| **BBL Class(es) and Method (s)** | | |
| Receiving Controller | 1. Void UpdateReceivedOrders (int poID,List<ReceiveOrder>OrderDetails) 2. IF ORDER IS FULL   Create new QuantityReceived in ReceivedOrderDetails  IF PARTIAL  Compare purchaseOrderID to existing purchaseOrderDetails and to ReceivedOrderDetails and update QuantityReceived   1. Update Parts Table (part quantity based on how many where received) 2. Returned items must be updated based on poID and return quantity 3. Close Order | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| PurchaseOrderId(R), ReceiveOrders (C) ReceiveOrderDetail (C) Part(U) | | ReceiveOrders(P) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| **Force Close Button** | | OnClick   * Validate Data(text field) * Send to BLL |
| **BBL Class(es) and Method (s)** | | |
| PurchaseOrdersController | Void Update\_ClosePurchaseOrder(int poID, str notes)   * Find matching poID * Update status of passed poID and notes fields | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| PurchaseOrders(U) | | PurchaseOrderList(P) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| **UnorderedVendorPartsLIST (ListView)**(VendorPart#, Description, Quantity) | | wired via ObjectDataSource |
| **BBL Class(es) and Method (s)** | | |
| UnorderedPurchaseItemCartContorller | List <UnorderedPurchaseItemCart> UnorderedPurItemCart\_GET   * Retrieves a list of Unordered Vendor Parts. * Returns Vendor Part #, Description, Qty. | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| UnorderedPurchaseItemCart(R) | | UnorderedPurchaseItemCart(E) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| **Insert(link)** | | OnClick / wired via ObjectDataSource   * Collect unorder part data * Validate required Data Fields * Send to BLL * Referesh **UnorderedVendorPartsLIST (ListView)** |
| **BBL Class(es) and Method (s)** | | |
| UnorderedPurchaseItemCartContorller | Void Insert\_UnorderPart (List< UnorderedPurchaseItemCart >AddUnorderedParts)(int poID, str desc, str vendorPartNumber, int qnt)   * Insert new VendorPartNumber | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| UnorderedPurchaseItemCart(C) | | UnorderedPurchaseItemCart(E) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| **Remove(link)** | | OnClick   * Find VendorPartNumber * Send to BLL to remove * Referesh **UnorderedVendorPartsLIST (ListView)** |
| **BBL Class(es) and Method (s)** | | |
| UnorderedPurchaseItemCartContorller | Void Delete\_UnorderPart (str vendorPartNumber)   * Removes vendorPartNumber | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| UnorderedPurchaseItemCart(D) | |  |

# You seem to have a handle on the subsystem. Remember all work in bll is done in a single transaction.