

**Note:** This document serves as a high-level overview and may not cover all rules or key features. Its purpose is to offer additional clarity for you and your team. If you have any questions, please see your instructor.

---

## Purchase Order

---

### Access & Authentication:

- Employees must log onto the system.
- Only users within the **Purchasing Role** can access the subsystem.
- Purchases are initiated by a Department Head.
- The form should display the Employee's full name.

### Purchase Orders (PO):

- Each vendor can have at most one "currently open" purchase order.
- An open PO is one where the **OrderDate** is NULL.
- Purchases are made to specific vendors.
- Purchase processing involves:

- Requesting a new suggested PO list.
- Retrieving an existing, unplaced vendor order.

## Suggested Purchase Order:

- Generated when there's no active order for the vendor.
- The system compares inventory reorder levels to on-hand quantities and unreceived PO quantities.
- Managers can add, adjust, or remove items from the suggested order.
- Clicking the **Update** button activates the order for the vendor.

## Current Active Order:

- An unplaced order.
- Managers can modify the order: add items, adjust quantities/prices, or remove items.
- Identified by a lack of an **OrderDate** .

## Placed Order:

- The order is finalized when the manager is satisfied.
- Once placed, the **OrderDate** is set, and the order cannot be modified.
- The StockItems's QuantityOnOrder will be updated at this time.

## Form Display & Interaction:

- Display a dropdown for vendor selection, with **[Select a Vendor]** as the default.
- Vendor-related actions:
  - Generate a new suggested order.
  - Retrieve and display the active order.
  - Items qualify for ordering when  $\text{ReorderLevel} - (\text{QuantityOnHand} + \text{QuantityOnOrder}) > 0$ .
  - The current order remains open until the order is placed.
  - Vendor details should be displayed.

## Command Buttons:

- **Remove Item** : Deletes an item from the PO display.
- **Refresh Item** : Updates item details on the PO display.
- **Add to Order** : Adds an item to the PO. Each item appears only once.

## Process Buttons:

- **Update Order** : Activates a suggested order.
- **Place Order** : Finalizes and places the order, updating relevant quantities.
- **Delete Order** : Removes an open order (placed orders cannot be deleted).
- **Clear** : Resets the order form.

# Images (Samples)

---

# Vendor



Select a Vendor from the following list:

HandTools Wholesale || 11206-106 ST. Edmonton AB T5R6H7



Select a Vendor...

HandTools Wholesale || 11206-106 ST. Edmonton AB T5R6H7

Hardware International || 172 DOWNERS GROVE Calgary AB T3W2D4

THOMAS Brown & Sons || 11206 EMPIRE BUILDING Vancouver BC V7U9K7

Nuts, Bolts and Fasteners || 172 Pilot Cres. Camrose AB T8G2D3

# Purchase Order

Welcome Hess Agonor!

## Vendor:

**PO:** #360  
**Vendor:** Hardware International  
**Phone Number:** 403.632.5698  
172 DOWNERS GROVE Calgary  
AB T3W2D4

Update

Place

Delete

Clear

Cancel

	ID	Description	QOH	ROL	QOO	QTO	Price	Original Price (\$)	
X	5581	Snow Joe 11 in 10 Amp Electric Snow Shovel	15	50	0	<input type="text" value="35"/>	<input type="text" value="129.65"/>	129.65	<div>ReCalc</div>
X	5582	Toro Electric Power Shovel Snowblower	23	65	0	<input type="text" value="42"/>	<input type="text" value="115.45"/>	115.45	<div>ReCalc</div>
X	5583	Toro 1500 Electric PowerCurve Snowblower	6	25	0	<input type="text" value="19"/>	<input type="text" value="206.78"/>	206.78	<div>ReCalc</div>
X	5587	Echo 25.4 cc Handheld Power Blower	16	25	0	<input type="text" value="9"/>	<input type="text" value="154.56"/>	154.56	<div>ReCalc</div>

**Subtotal:** \$14,706.51

**GST:** \$735.33

**Total:** \$15,441.84

Refresh Cost

## Stock Item(s):

ID	Description	QOH	ROL	QOO	Buffer	Price (\$)	
4567	65 oz Hammer	0	150	0	-150	11.74	<a href="#">Add</a>
5555	Canadian Solid State Stud Finder	66	50	0	16	5.65	<a href="#">Add</a>
5579	Botanica Compact Shovel Round Point	65	50	0	15	13.56	<a href="#">Add</a>
5580	Botanica D grip Round Point 35 in	2	65	0	-63	10.89	<a href="#">Add</a>
5584	Yukon Sleigh Shovel	125	75	0	50	27.56	<a href="#">Add</a>
5585	Echo 21.2 CC Tiler/Cultivator	5	25	0	-20	265.54	<a href="#">Add</a>

## View Models

View models and methods are used as examples, and you are welcome to modify them as needed in your coding example. Please include comments to explain any modifications or choices you make.

```
public class VendorView
{
    public int VendorID { get; set; }
```

csharp 

```
public string VendorName { get; set; }
public string Phone { get; set; }
public string Address { get; set; }
public string City { get; set; }
public string Province { get; set; }
public string PostalCode { get; set; }
}
```

```
public class EmployeeView
{
    public int EmployeeID { get; set; }
    public string FullName { get; set; }
}
```

csharp



```
public class PurchaseOrderView (Web Page View)
{
    public VendorView Vendor { get; set; }
    public int PurchaseOrderID { get; set; }
    public List<ItemView> Items { get; set; }
    public decimal SubTotal { get; set; }
    public decimal GST { get; set; }
}
```

csharp



```
// This view can be used as the vendor inventory with a PurchaseOrderDetailID == 0 as we are
not using it.
// The PurchaseOrderDetailID will be set when you are either "Update" or "Place" using the
save method in your project library.
```

csharp





```
// Nov 8
// Added PurchaseOrderDetailID
// Updated ID to StockItemID for better clarification.
public class ItemView
{
    public int PurchaseOrderDetailID { get; set; } // this is the
    public int StockItemID { get; set; }
    public string Description { get; set; }
    public int QOH { get; set; } // Quantity on Hand
    public int ROL { get; set; } // Reorder Level
    public int QOO { get; set; } // Quantity on Order
    public int QTO { get; set; } // Quantity to Order
    public decimal Price { get; set; }
}
```

```
public class PurchaseOrderEditView
{
    public int PurchaseOrderID { get; set; }
    public int VendorID { get; set; }
    public int EmployeeID { get; set; }
    public List<ItemDetailView> ItemDetails { get; set; }
    public decimal SubTotal { get; set; }
    public decimal GST { get; set; }
}
```

csharp 

```
public class ItemDetailView
{
    public int StockItemID { get; set; }
    public int QTO { get; set; }
}
```

csharp 

```
    public decimal Price { get; set; }  
}
```

// Nov 8 // Updated return from FetchInventoryBy from Item to ItemView

## Methods

```
public List<VendorView> GetVendors()  
public EmployeeView GetEmployee(int employeeID)  
public PurchaseOrderView GetVendorPurchaseOrder(int vendorID)  
public List<ItemView> FetchInventoryBy( int vendorID)  
public List<ItemView> FetchInventoryBy(PurchaseOrderView PurchaseOrderInfo, int vendorID)  
  
public void or bool AddItemToPurchaseOrderLine(ItemView item)  
public void or bool RemovePurchaseOrderLine(int purchaseOrderLineID)  
public PurchaseOrderView UpdatePurchaseOrder(PurchaseOrderEditView order, bool placeOrder)  
public bool DeletePurchaseOrder(int purchaseOrderID)
```

csharp

