* User can view the catalog of products even when they are not logged on.
* A user is allowed to have only **one** shopping cart (but as many items in the cart as they want). Duplicate items in the cart is not allowed (changing of item quantities must be done in the first phase of the Checkout process).
* Discontinued products should not be shown and cannot be added to the cart.
* If a user is logged in, show the "Add" button and quantity textbox next to the product; otherwise show a message that they must log in to add items to the cart
* Do not allow employees to purchase items; if the purchaser is an employee, then show a message informing them that they must not use their employee account to shop online.
* User can only add products if logged in. When clicking Add, they must supply the quantity (which must be a whole number greater than zero)
* Quantity next to Add defaults to 1
* Added products go immediately to the user's shopping cart table items
  + Method signature for SalesController.AddToCart()
* public void AddToCart(IPrincipal user, int productId, int quantity)
* Users stay on the shopping page after adding an item, but a message is shown confirming that the item has been added to the cart
  + [*Optional*] The **View Cart** link should also include a number of the distinct items in the cart.

Checkout - three phases: View Cart, Purchase Info, Place Order (each in their own pages/tab)

* View Cart
  + When listing the cart items, place an "Out of Stock" warning next to the item description/quantity along with the LastUpdatedDate so they can see when that item was added to the cart; An "out of Stock" warning appears if ther quantity on hand for the item is zero.(don’t know how to test it)
  + User can change quantities and/or remove items from their cart
  + Add a Continue Shopping button that navigates them to the shopping page.
  + Add a Proceed button that will navigate to the next phase, Purchase Info.
* Purchase Info
  + This will be a "placeholder" page, as we do not yet have any customer or payment tables in the database. As such, use form controls that are disabled.
  + Confirm the customer's information (for shipping, etc.): First and last names, shipping address, billing address, etc.
  + Add a Return to Cart button that will navigate to the previous phase, View Cart. This button must be enabled.
  + Add a Proceed button that will navigate to the next phase, Place Order. This button must be enabled.
* Place Order
  + Display Customer order details summary information
  + Box to enter/seelct a coupon & apply to the pricing of the order
  + Payment types are 'C', or 'D', for 'Credit' and 'Debit', respectively
  + Subtotal, GST, Total
  + Coupon value is a rate; ie 5 means 5% discount
  + Place Order button
    - Ensure that the product is not discontinued
    - If items are all in stock, the link button is for ordering all items (**Place Order**)
    - If items are not all in stock, the link button is for **Order In-Stock Items Only**
      * Subtotal, GST, Total based on in-stock items only
      * Show an "Out of Stock" warning and list all items that would be EXCLUDED from the order so they can see what items would be omitted
      * EXCLUDED item will have the SaleDetail Backordered amount set to the order quantity
    - Generate a new GUID as the payment token. Normally, this would be a value received from a Payment Processor, but we are not using one in this project.
    - Shipped date is filled for items shipped(today)
  + Placing the order is done as a comlete transaction on the server (a single context.SaveChanges()). In this transaction, you must do the following:
    - Create a sale and sale items based on all the information in the user's shopping cart.
    - Reduce the stock on hand based on the quantities of the items purchased **only if** the remaining quantity on hand is greater than zero (otherwise, the sale detail will show the item as backordered and the shipped date will be null). As appropriate, set the shipped date to the current date; we are assuming items are "set aside" for shipping on the same day.
    - Remove the user's cart and cart items.