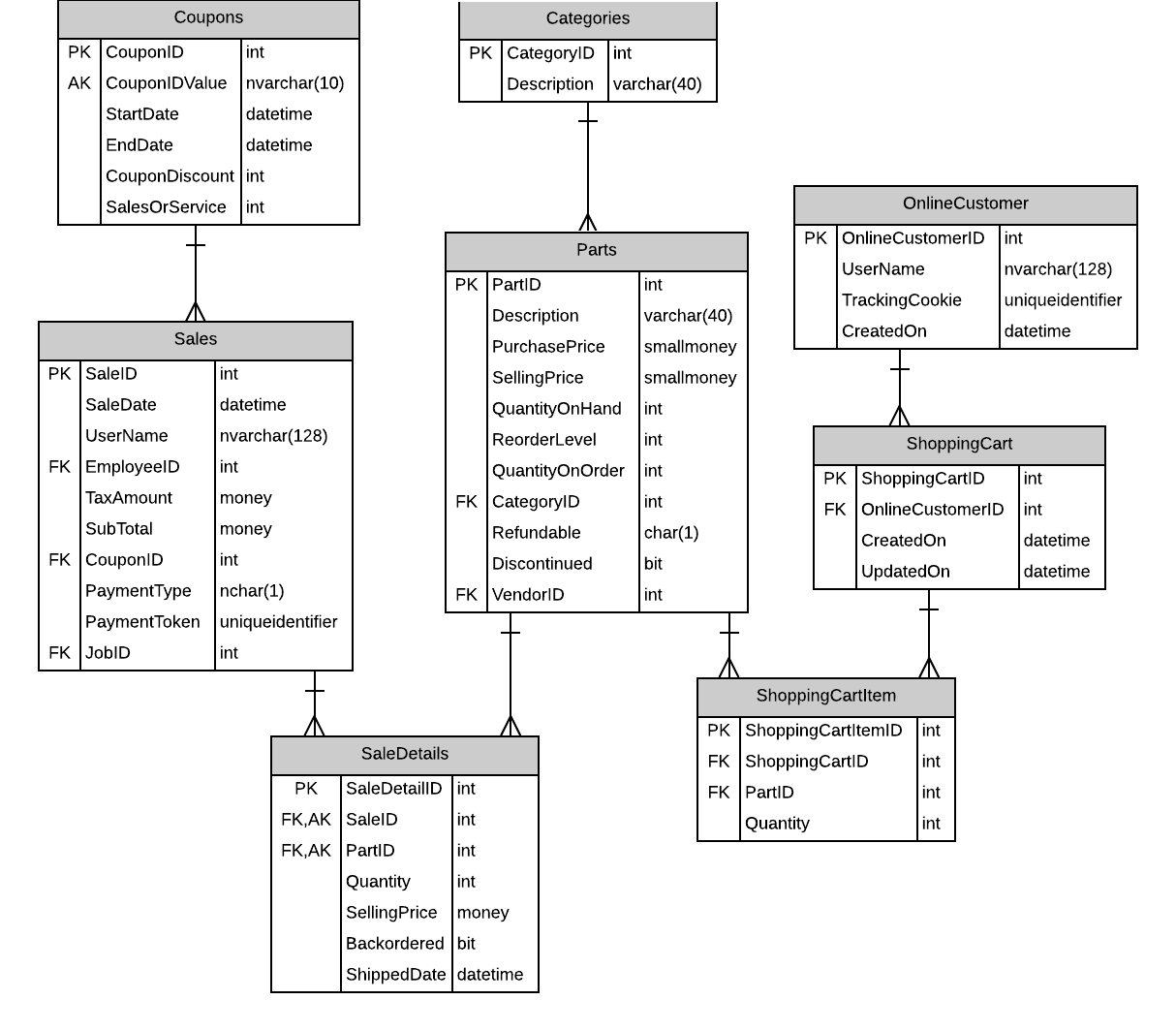
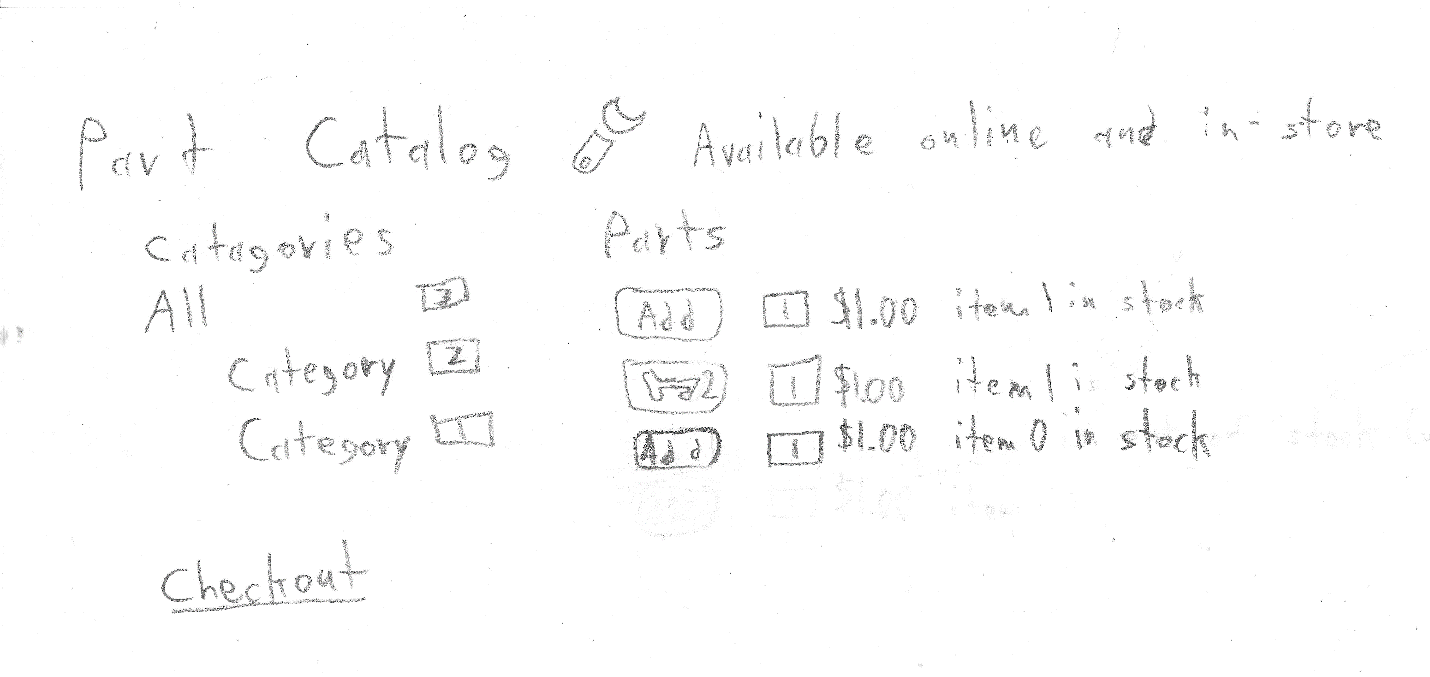
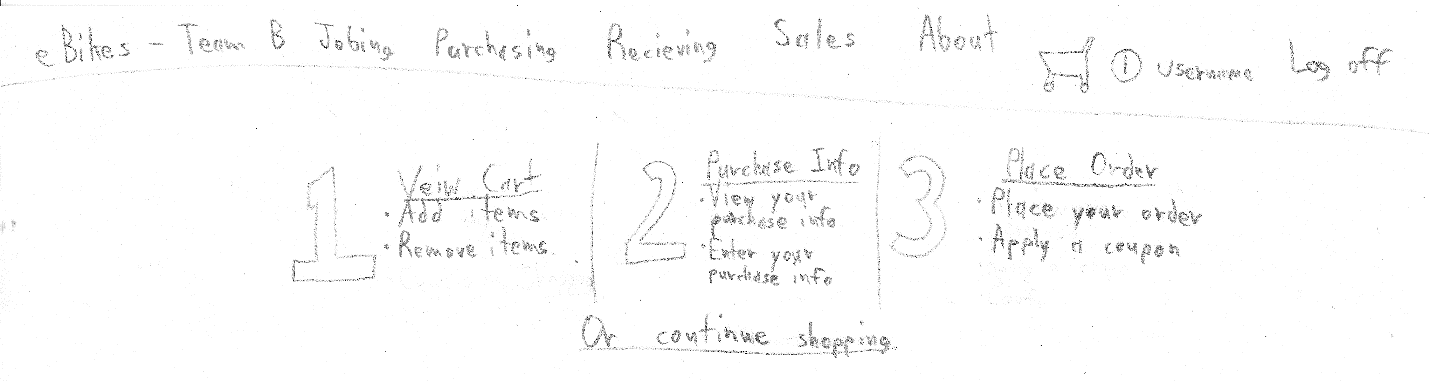
Sales

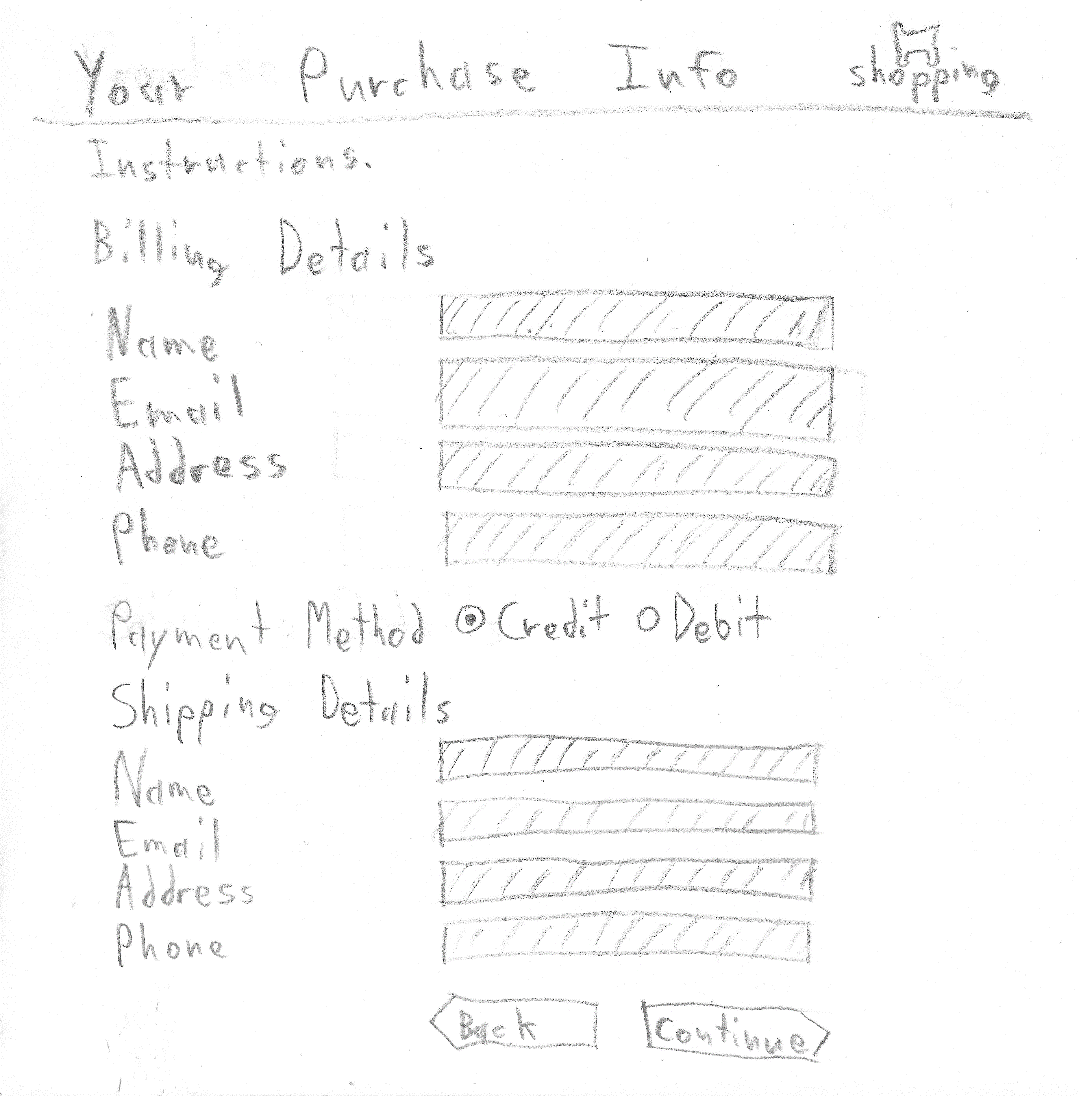
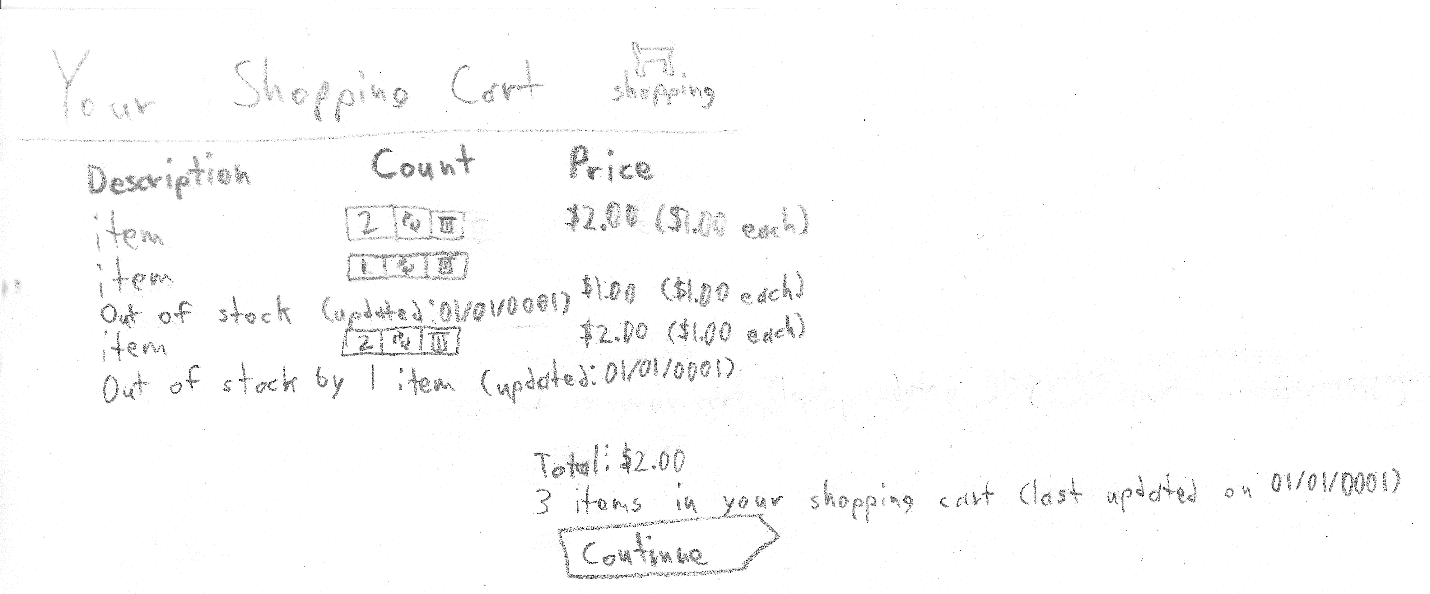
# ERD for UX process

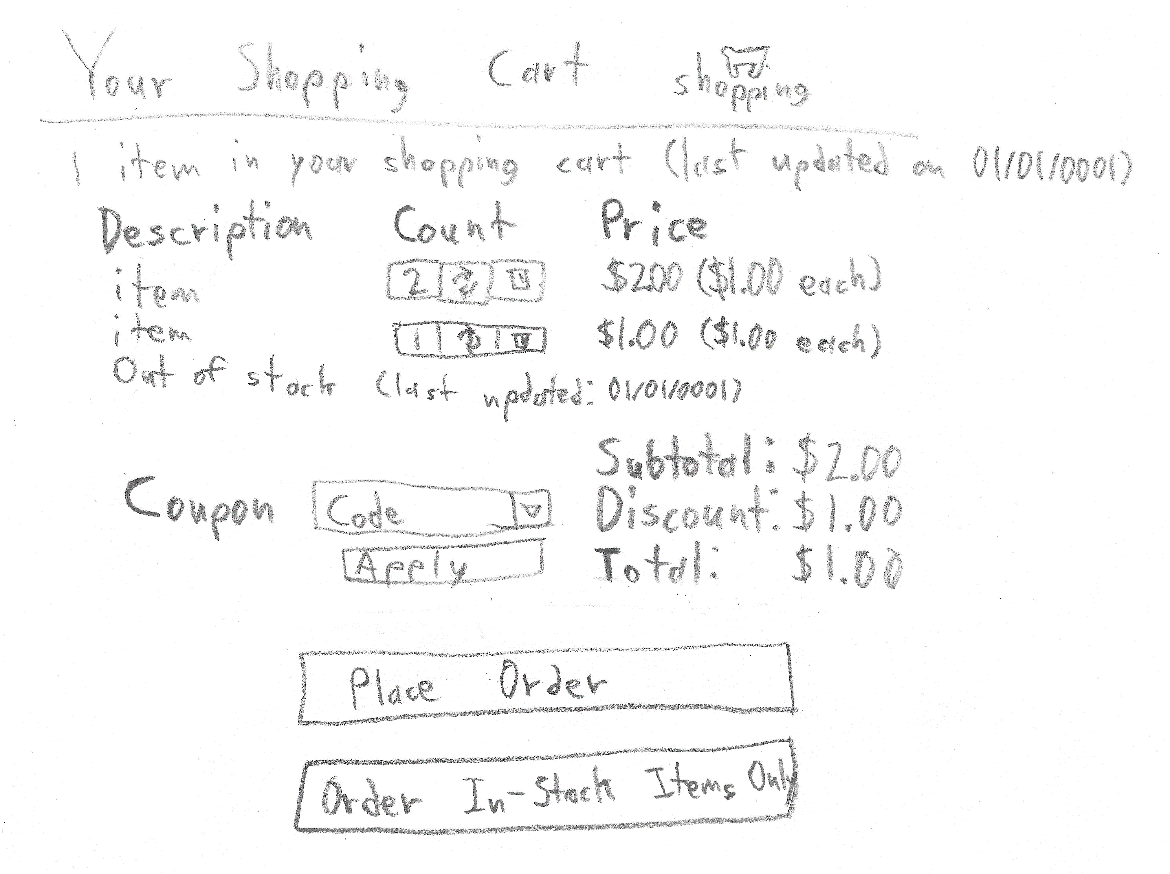


# Possible UX Interface









# Event and Wiring Summations

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CategoriesList | | None / wired via ObjectDataSource |
| **BBL Class(es) and Method (s)** | | |
| CategoriesController | List<OnlineCategories> GetCategories()  Return a list of all available categories. | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| Categories (R), Parts (R) | | OnlineCategories (P), Category (E), Part (E) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CategoryIdValue, PartsListView | | None / wired via ObjectDataSource |
| **BBL Class(es) and Method (s)** | | |
| PartsController | List<OnlinePart> GetParts(IPrincipal user, int categoryId)  Returns a list of all available parts in a category (including how much the user added to their cart). If the categoryId is 0, all parts will be returned. | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| Parts (R) | | OnlinePart (P), Part (E), ShoppingCartItem (E), ShoppingCart (E), OnlineCustomer (E) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| PartsListView | | PartsListView\_OnItemCommand |
| **BBL Class(es) and Method (s)** | | |
| SalesController | Void AddToCart(IPrincipal user, int productId, int quantity)  Adds an item to the user’s cart (creating their cart if not present) if,   1. The user is logged in, 2. And the user is not logged in as an employee.   If any of these conditions do not apply, an error will be thrown.  If the user does not have an associated OnlineCustomer, or ShoppingCart, they will be created automatically. | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| ShoppingCart (C,R,U), ShoppingCartItem (C) | | ShoppingCart (E), ShoppingCartItem (E), OnlineCustomer (E) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| PartsListView, PartsReviewListView | | None / wired via ObjectDataSource |
| **BBL Class(es) and Method (s)** | | |
| ShoppingCartController | OnlineShoppingCart GetShoppingCart(IPrincipal user)  Gets the user’s shopping cart (or null if the user does not have one). | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| ShoppingCart (R) | | OnlineShoppingCart (D), ShoppingCart (E), OnlineCustomer (E) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| PartsListView | | PartsListView\_OnItemCommand |
| **BBL Class(es) and Method (s)** | | |
| ShoppingCartItemController | Void ChangeShoppingCartItemQuantity(int shoppingCartItemId, int quantity)  Changes the quantity of shopping cart items. If the quantity is being set to 0, the item will be removed. If the shopping cart does not exist, an error will be thrown. | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| ShoppingCartItem (U,D) | | ShoppingCartItem (E) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CouponDropdownList | | None / wired via ObjectDataSource |
| **BBL Class(es) and Method (s)** | | |
| CouponsController | List<ListValue> GetCoupons()  Gets a list of all available coupons. | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| Coupons (R) | | ListValue (P) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CouponDropdownList, ApplyCouponButton | | ApplyCouponButton\_OnClicked |
| **BBL Class(es) and Method (s)** | | |
| UserManager | Void SetCoupon(IPrincipal user, int couponId)  Set’s the user’s current coupon. | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
|  | |  |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CouponDropdownList | | Page\_OnLoad |
| **BBL Class(es) and Method (s)** | | |
| UserManager | Int GetCoupon(IPrincipal user)  Gets the user’s current coupon. | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
|  | |  |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| PlaceOrderButton | | PlaceOrderButton\_OnClicked |
| **BBL Class(es) and Method (s)** | | |
| Sales | Void PlaceOnlineOrder(int shoppingCartId, OrderDetails details)  Places an order using the shopping card and the provided details. If the shopping cart does not exist, or the details are invalid, an error will be thrown. | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| ShoppingCart (R,D), ShoppingCartItem (R,D), SaleDetails (C), Sales (C), Coupons (R) | | Sale (E), SaleDetail (E), ShoppingCart (E), ShoppingCartItem (E), Coupon (E), OrderDetails (P) |

|  |  |  |
| --- | --- | --- |
| **Controls** | | **Events** |
| CartItemsCount | | CartItemsCount\_OnLoad |
| **BBL Class(es) and Method (s)** | | |
| ShoppingCartItem | Int GetShoppingCartItemCount(IPrincipal user, int partId)  Gets the count of items in the user’s shopping cart by partId. If partId is 0, all parts will be summed. | |
| **SQL Table(s): (C,R,U,D)** | | **Entities/DTOs/POCOs** |
| ShoppingCart, (R) ShoppingCartItem (R) | | ShoppingCart (E), ShoppingCartItem (E) |

What about the remove button the the View Cart??? Otherwise you have a good handle on this subsystem.