**Notes on the Final E-Bazaar Project**

Completed Scenarios These show you implementation strategies at all layers of the application, including reading and writing data and rule validation.

1. Save New Address
2. Retrieve customer’s saved addresses

Left to Be Done

*Browse and Select Use Case*

* Product Details: In bsPCB getProduct, BrowseSelectController.getProduct
* Shopping Cart: Delete and edit functions not working (start with bsPCB)
* BrowseSelectController: read cart item, product name is stubbed
* Quantity Rules: RulesQuantity in business, BrowseSelectController runRulesOnQuantity
* Save Cart: - bsPCB, BrowseSelectController CustSubsystem saveShoppingCart DbClassShoppingCart
* Retrieve Saved Cart – from bsPCB through DbClassShoppingCart – many things are missing

*Checkout Use Case*

* Submit order – many steps missing, starting with Checkout PCB.

*Manage Products Use Case  
\** In JSF, there is UI support but all main functions need to be implemented.

*View Order History Use Case  
\** In JSF, there is no UI for this yet. In Swing, default data must be replaced. Many of the steps are missing to complete functionality

*Products Subsystem*

* Check DbClassProduct and ProductsSubsystemFacade for stubs

*Customer Subsystem*

* Reading saved cart, default billing, default payment in initialize function – need to be implemented
* Check DbClasses and CustomerSubsystemFacade for stubs

*Shopping Cart Subsystem*

* Check DbClassShoppingCart and ShoppingCartSubsystemFacade for stubs

*Order Subsystem*

* DbClassOrder and OrderSubsystemFacade